

TpaCAD

Версия 2.4.28

Редактор программ



Tecnologie e Prodotti per l'Automazione

Настоящая документация является собственностью компании TPA S.r.l.
Запрещается копирование без предварительного согласования.
Компания сохраняет за собой право в любой момент вносить
изменения в текст.

Оглавление

1 Введение 1

1.1 Что это такое 1
1.2 Подключение и режимы функционирования 2
1.3 Право доступа к системе 7
1.4 Многоязыковая поддержка 8
1.5 Совместимость формата 9
1.6 Новинки 9
1.7 Системные требования 11
Выполнение TpaCAD на виртуальных станках 11
1.8 Контроль сигнальных сообщений при запуске TpaCAD 12
1.9 Справка 13

2 Версии и обновления 14

2.1 Версии 2.4.0 (22 мая 2020 г.) - 2.4.28 (03 февраля 2026 г.) 14
2.2 Версии 2.3.1 (08 марта 2019 г.) - 2.3.20 (02 декабря 2020 г.) 22
2.3 Версии 2.2.0 (04 декабря 2017 г.) - 2.2.15 (28 марта 2019 г.) 25
2.4 Версии 2.0.0 (08 апреля 2016 г.) - 2.1.18 (26 сентября 2018 г.) 27
2.5 Версии 1.4.0 (10 апреля 2015 г.) - 1.4.9 (30 марта 2016 г.) 31
2.6 Версии 1.3.0 (10 февраля 2014 г.) - 1.3.11 (31 марта 2015 г.) 34
2.7 Версии 1.1.0 (06 декабря 2012 г.) - 1.2.4 (10 октября 2013 г.) 37

3 Графический интерфейс 41

3.1 Порядок отображения 41
3.2 Список горячих клавиш и клавиш мыши 46

4 Работа с программами 49

4.1 Создать программу 49
Создать программу по модели 49
4.2 Открыть и импортировать программу 49
Импорт программы из внешнего формата 50
Импорт с созданием нескольких файлов 50
Импорт программы в формате TpaEdi32 51
Импорт программы в формате EdiCad 51
Открытие программы-детали, созданной во внешней среде 52

Запуск TpaCAD из Управления ресурсами 52
Сигналы при открытии программы 52
Формат регистрации программы-детали 53
4.3 Открыть программу из меню Файл 53
4.4 Буксировка (drag & drop) 53
4.5 Распечатать программу 54
4.6 Сохранить программу 54
Имена, которые нельзя использовать 55
4.7 Оптимизировать программу 55
4.8 Напечатать ярлык программы 55
4.9 Экспортировать программу 56
4.10 Преобразовать архив программ 56
4.11 Оптимизировать архив программ 57
4.12 Показать предварительный просмотр для оптимизации программы 57
4.13 Установка 58
4.14 Рабочая среда 58
4.15 Несколько копий TpaCAD 58
4.16 Таблица инструментов 59
4.17 Информация о внешних компонентах, соединенных с TpaCAD 60
5 Порядок задания графического представления 61
5.1 Индивидуализировать виды 61
5.2 Индивидуализировать вид при введении коррекции на инструмент 64
5.3 Регулировать вид 65
5.4 Трехмерное изображение 66
5.5 Специальные виды и Фильтры вида 67
5.6 Информация о профиле 68
6 Деталь 69
6.1 Графическое представление Общего вида 69
6.2 Геометрия детали 69
6.3 Задания 74
Зона "Задания" 74
Размеры, Режимы выполнения и Свойства 74
LxHxS 74
Режим выполнения 75
Свойства 76
Переменные "O" 76
Переменные "V" 78
Переменные "R" 78
Направляемое редактирование 79
Восстановление переменных "r" из существующей программы 81
Специальные уставки 81

Дополнительная информация 83
Моделирование 83
Фиктивные грани 83
Информация, относящаяся к фиктивным граням на локальной строке состояния 85
Закругленные фиктивные грани, поверхности и моделирование 85
Пример 1 86
Пример 2 88
Пример 3 89
Пример 4 90
Раздел привязок 90
Оптимизации 91
Последовательности 91
6.4 Продвинутые задания 92
Исключения 92
Слои 93
Специальные фильтры 94
7 Грань 97
7.1 Графическое представление Вида грани 97
7.2 Порядок открытия 99
7.3 Зона текста ASCII 100
8 Грань-деталь 103
8.1 Что это 103
8.2 Порядок открытия 103
8.3 Зона задания операций 103
8.4 Зона Текст ASCII 104
8.5 Поле F 104
8.6 Представление 104
8.7 Последовательности исполнения 105
9 Операции 106
9.1 Типы операций 106
Простые и сложные операции 106
Точка применения 107
Технология 108
Ориентированные геометрии 110
Технологический приоритет 113
Графическое представление 113
9.2 Профиль 113
Операции профиля 113
Построение профиля 114
Точка применения 115
Программирование углов 115
Касательные и секущие 115
Разметка 116
Задание технологии 117
Многократная подготовка 119
Открыть и закрыть профиль 120
Зацепить профили 123

Простые профили 123
Коррекция на инструмент 124
Изменение коррекции 128
Изменение стороны коррекции 130
Отображение 130
Выполнение профиля для насечки с острыми ребрами 131
Компенсация диаметра коррекции в случае конического инструмента 133
Задание профилей в режиме "грань-деталь" 133
9.3 Логические команды 134
Структуры IF ... ELSEIF ... ELSE ... ENDIF 134
Команда "Exit" 136
Команда "Error" 136
Команда "Warning" 136
Переменные J 137
Глобальные функции 138
9.4 Подпрограмма 139
Подпрограмма 139
Задать переменные подпрограммы 141
Позиционировать подпрограмму 143
Запрограммированная точка применения 145
Зацепление точки 145
Конечная точка применения 146
Применить операции к соответствующей грани 146
Вложенные вызовы (автоматические) 146
Выбор вложенных граней 147
Размещение вложенных вызовов 147
Решение переменных <j> в автоматических вложенных вызовах 148
Прямые вызовы 149
Запрограммированные вложенные вызовы 149
Решение переменных <j> в запрограммированных вложенных вызовах 151
Применить геометрические трансформации 151
Повторения при выполнении подпрограммы 152
Повторения со свободным распределением 152
Повторения с распределением в матричной форме 153
Показать развертку подпрограммы 154
Встроить вызовы подпрограмм 155
Направляемое редактирование и функции с использованием дополнительных средств 155
9.5 Запрограммированные инструменты 156
Продвинутое использование Запрограммированных инструментов 160
9.6 Автоматические грани 161
9.7 Ввод геометрических элементов из меню "Начертить" 163
9.8 Ввод Закладок 169
9.9 Изменение и Ввод 170
Выбрать точку вставки в программном списке 170
Выбор 171
Общие команды выбора 172
Изменить текущую операцию 172
Общие команды Изменения в программе грани 173
Изменение свойств 173
Общие команды изменения 174
Найти 176

Заменить 177
Заменить переменную 179
Решить 181

10 Инструменты 183

10.1 Введение 183

10.2 Общие 183

Центрирование и выравнивание 183
Перенос 184
Вращение 185
Правка (меню "Графическое представление") 187
Симметрично отобразить 187
Разбить 188
Продвинутые положения 189
Повторения 190
Свободные повторения 190
Прямоугольная серия 190
Круговая серия 191
Повторения на профиле 191

10.3 Инструменты профиля 192

Изменить отрезок профиля 192
Заменить ребро на дугу 194
Изменить линию в разметку 194
Применить вход к профилю 195
Применить выход к профилю 196
Замкнуть профиль 196
Инвертировать профиль 197
Масштабировать профиль 197
Растянуть профиль 198
Разделить профиль 199
Отсоединить каждый отрезок профиля 200
Продлить 200
Скруглить профиль 201
Выполнить фаску на профиле 202
Минимизировать профиль 203
Фрагментировать профиль 204
Линеаризировать по Z 206
Объединение профилей 206
Переносом 206
Соединительным отрезком 207
Соединить соседние профили 207
Сместить подготовку на замкнутый профиль 208
Применить подготовку к профилю 208
Применить многократную подготовку 209

10.4 Построения 209

Исправленный профиль 209
Применить точки крепления к профилю 210
Применить подачу глубины 213
Применить повторение профиля 215
Повторить профиль 216
Разрезать профили 217
Построение профиля 218
Разделить на точках пересечения 219
Создание текстов 220
Создать сплайн из полилинии 224

Выемка зоны 228
Поворот профилей в картезианской плоскости 232
10.5 Построения раскроя профилей 233
Раскрой 233
Решение раскроя 242
10.6 Продвинутые инструменты в программе грани 244
Создать фиктивную грань из геометрии 244
Создать поверхность, исходя из геометрии 245
Создать моделирование, исходя из геометрии 245
Создать шрифт, исходя из геометрии 245
10.7 Инструменты утилиты 248
Проставление размеров 248
Размеры 249
10.8 Общие инструменты программы 249
Применить технологию 250
Преобразовать [мм]-[дюймы] 250
Подтвердить профили 250
Применить сокращение к профилям 252
Применить фрагментацию к профилям 252
Применить соединение к профилям 252
10.9 Общие трансформации программы 253
Повернуть деталь 253
Зеркальное отображение детали 254
Перевернуть деталь 255
11 Параметрическое программирование 257
11.1 Введение 257
11.2 Переменные и параметры числового типа 257
11.3 Функции 258
11.4 Переменные и параметры типа строки 258
11.5 Числовые форматы специального применения 260
11.6 Термы выражения 261
Операторы 261
Арифметические 261
Логические 261
Скобки, разделители 262
Переменные аргументы 262
Общие аргументы 262
Режим выполнения 264
Задания среды 265
Переменные для детали 266
Ссылки на переменные детали 268
Задания, связанные с применением подпрограммы или макропрограммы 269
Задание клиентских разделов 274
Глобальные переменные 274
Вспомогательные функции 274
Математические функции 275
Тригонометрические функции 278
Начала тригонометрии 278
Функции 279
Функции, действующие на строки 280

Логические функции 283
Технологические функции 285
Технологические параметры, заданные с символическим формализмом 285
Функции доступа к общей группе установки 286
Функции доступа к ветви станка для конфигурации групп головки 286
Функции доступа к инструментам 287
Функции прямого доступа к матрицам установки 292
Универсальные функции геометрической библиотеки 295
Функции расчета углов 295
Функции расчета расстояний 298
Функции определения точек на геометрических элементах 299
Функции вращения точки 305
Функции симметрии 306
Функции вращения угла 307
Функции коррекции геометрического элемента со сдвигом 308
Функции преобразования координат и считывания информации граней 309
Алгебраические функции 315
Функции доступа к информации запрограммированной операции 315
Клиентские функции 319

12 Сообщения об ошибке 320

12.1 Общие ошибки 320
1 - Ошибка процедуры 320
2 - Ошибка распределения памяти 320
5 - Ошибка доступа к файлу 321
6 - Ошибка доступа к Локальным примечаниям 321
7 - Ошибка доступа к временному файлу Undo 321
13 - Уровень системы не позволяет выполнить требуемую операцию 321
18 - Текущая операция неправильна 321
36 - Достигнуто максимальное число операций для грани 322
38 - Невозможно вставить указанную операцию в текущей грани 322
39 - Инструмент не может использовать основную операцию 322
41 - Ошибки в присвоении свойства операции 322
42 - Модификации или замены не были внесены 322
49 - Инструмент применяется только к профилям 323
281 - Чтение файла: неожиданный конец файла 323
282 - Чтение файла: закрытие раздела не найдено 323
283 - Чтение файла: неправильный идентификатор грани 323
284 - Чтение файла: неправильный идентификатор операции 323
286 - Чтение файла: ошибка декодировки файла 324
287 - Чтение файла: программа не совместима с назначениями среды 324

12.2 Специфические ошибки при применении инструментов 324
50 - Инструмент не обработал трансформации 324
51 - Инструмент применяется к простому профилю 324
53 - Минимизировать профиль: угол сокращения превышает значение 90.0° 324
54 - Фрагментировать профиль: максимальная длина отрезков равна нулю 324
55 - Применить точки крепления профиля: неправильное число точек крепления [минимум: 2; максимум: 255] 325
56 - Применить точки крепления профиля: неправильная длина точек крепления или чрезмерная компенсация на инструмент 325
59 - Применить точки крепления профиля: остаточная толщина точек крепления неправильна или не задана 325

60 - Применить точки крепления профиля: невозможно распределить точки крепления по профилю (уменьшить число точек крепления) 325
61 - Инверсия профиля: получены сложные неинвертируемые коды 325
62 - Применить инструмент: сложный код конца профиля не завершается с отрезком профиля 325
63 - Сместить подготовку на профиль: указанное положение совпадает с текущей подготовкой 326
64 - Инструмент применяется к замкнутому профилю 326
67 - Скругление или фаска профиля: заданный радиус равен нулю 326
68 - Резать профиль: указанное положение уже совпадает с подготовкой 326
69 - Разрезать профиль: указанное положение уже завершает профиль 326
70 - Начало/ Конец профиля: не задана исходная операция 326
71 - Применить инструмент: невозможно зацепиться перед профилем 327
72 - Начало профиля: не задано перемещение для точки начала профиля 327
73 - Конец профиля: не задано перемещение для точки конца профиля 327
75 - Соединение профилей: второй профиль определен неправильно 327
78 - Соединение профилей: профили разъединены 327
79 - Масштабировать профиль: получаются сложные неизменяемые коды 327
80 - Масштабировать профиль: коэффициент увеличения или уменьшения не задан или равен 1.0 328
82 - Инструмент запрашивает слишком много повторений 328
85 - Применить инструмент: профиль задает дуги в плоскости, отличной от xy 328
86 - Конец профиля: невозможно зацепить конец после профиля 328
88 - Применить инструмент: невозможно задать подготовку из-за отсутствия исходного кода 328
92 - Инструмент не запросил движения по осям 329
93 - Инструмент запросил нулевой поворот 329
94 - Инструмент не запросил повторных применений 329
95 - Создать текст: надпись была обрезана до максимально допустимой длины для построения кривой 329
96 - Создать текст: неправильное коническое сечение развертки 329
98 - Создать текст: высота символов недостаточна (минимум = eps * 100) 329
99 - Создать текст: неправильная дуга 330
294 - Выемка зоны: профиль не замкнут 330
295 - Выемка зоны: профиль не соответствует заданному инструменту 330
296 - Выемка зоны: радиус инструмента задан на ноль [минимум: 10*эпсилон] 330
297 - Выемка зоны: перекрытие превышает радиус инструмента 330
298 - Выемка зоны: отклонение по глубине идет от Z=0.0 330
299 - Выемка зоны: неправильная координата Z воздуха 330
300 - Выемка зоны: слишком много профилей для оценки (>300) 331

12.3 Ошибки параметрического программирования 331

101 - Параметрическое программирование: слишком длинная строка 331
102 - Параметрическое программирование: неправильный синтаксис 331
103 - Параметрическое программирование: переменная <r>, вызванная по имени, не найдена 332
105 - Параметрическое программирование: численное значение превышает допустимый диапазон (-3.4E+30; 3.4E+30) 332
106 - Параметрическое программирование: решение параметра в виде слишком длинной строки (макс.: 260 символов) 332
109 - Параметрическое программирование: неправильный контекст для использования аргументов подпрограммы 332
111 - Параметрическое программирование: неправильный контекст для использования переменных <$> 332
112 - Параметрическое программирование: неправильный контекст для использования переменных <r> 332

113 - Параметрическое программирование: неправильный контекст для использования переменных <v> 333
114 - Параметрическое программирование: неправильный контекст для использования переменных <o> 333
115 - Параметрическое программирование: неправильный контекст для использования переменных <j> 333
116 - Параметрическое программирование: неправильный контекст для использования имени операции 333
117 - Параметрическое программирование: неправильный индекс переменной <r> 333
118 - Параметрическое программирование: неправильный индекс переменной <j> 333
119 - Параметрическое программирование: неправильный индекс переменной <$> 334
120 - Параметрическое программирование: неправильный индекс переменной <v> 334
121 - Параметрическое программирование: неправильный индекс переменной <o> 334
122 - Параметрическое программирование: функция имеет слишком много операндов (максимум: 30) 334
123 - Параметрическое программирование: функция без операндов 334
124 - Параметрическое программирование: функция с ошибочным числом операндов 334
125 - Параметрическое программирование: деление на ноль 334
126 - Параметрическое программирование: значение тригонометрической функции (sin, cos) не попадает в интервал от -1 до +1 335
127 - Параметрическое программирование: квадратный корень из отрицательного числа 335
128 - Параметрическое программирование: возведение в степень с неправильным показателем [минимум = 0; максимум = 10] 335
129 - Параметрическое программирование: неправильная функция геометрической библиотеки 335
130 - Параметрическое программирование: функция с опущенным обязательным аргументом 335
132 - Параметрическое программирование: неправильный угол для расчета тангенса 335
134 - Параметрическое программирование: слишком много встроенных вызовов клиентских функций (максимум: 5) 335
135 - Параметрическое программирование: неправильное использование клиентской функции 335
136 - Параметрическое программирование: неправильное использование аргументов arg# res# var# 336
137 - Параметрическое программирование: неправильный индекс или имя аргумента arg# 336
138 - Параметрическое программирование: неправильный индекс или имя аргумента res# 336
139 - Параметрическое программирование: ошибка из-за вызова клиентской функции 336
140 - Параметрическое программирование: ошибка в использовании функций, отведенных для клиентских функций 336
141 - Параметрическое программирование: неправильный индекс аргумента var# 336

12.4 Ошибки при обработке изменяемых геометрий 336

22 - Невозможно удалить грань, если для нее заданы операции 336
144 - Изменяемые геометрии: исходная грань неверна или не задана 336
145 - Изменяемые геометрии: не все вершины грани различны 337
146 - Изменяемые геометрии: вершины грани находятся на одной линии 337
147 - Изменяемые геометрии: неправильная полярная геометрия грани 337
148 - Изменяемые геометрии: неправильная плоскость вращения 338
149 - Изменяемые геометрии: невозможно задать третью точку грани 338
150 - Изменяемые геометрии: неправильная глубина точки 339
165 - Изменяемые геометрии: неправильный радиус закругления грани 340
166 - Изменяемые геометрии: ошибка в решении геометрии поверхности 340

167 - Изменяемые геометрии: максимальное число элементов на поверхности 340

12.5 Ошибки на этапе компиляции программы грани 341

151 - Неправильный код операции <имя рабочего кода> 341
152 - Параметр <имя параметра>: неправильное значение 341
153 - Параметр <имя параметра>: задать формат "$nn" 341
155 - Свойство <имя поля>: недопустимое значение 341
156 - Поле <имя поля>: значение не соответствует заданному минимуму 342
157 - Поле <имя поля>: значение превышает заданный максимум 342
158 - Моделирование: код или последовательность кодов неправильна 342
161 - Автоматических граней слишком много или они недоступны 342
162 - Поле F: недопустимое значение 342
190 - Операции за пределами применения (ось <имя оси>) 343

12.6 Ошибки в операциях профиля 343

192 - Расчетный радиус бесконечен 343
193 - Нулевой радиус 343
194 - Неправильная дуга 343
195 - Неправильная секущая 343
196 - Неправильная касательная в начале 343
197 - Неправильная касательная в конце 343
198 - Расчетная точка вне отрезков 344
199 - Несуществующая точка пересечения 344
200 - Неправильная дуга (точки не различны) 344
201 - Неправильная дуга (точки на одной прямой) 344
202 - Овал: неправильный радиус 344
203 - Овал превратился в круг 344
204 - Овал: ось/оси нулевые или неправильные 344
205 - Эллипс/Овал: начальная точка вне габаритов конуса 344
206 - Прямоугольник: неправильная ось/оси или радиус 344
207 - Многоугольник: неправильное число сторон 344

12.7 Ошибки в подпрограмме или макропрограмме 345

208 - Запрограммированный инструмент: совпадение не найдено 345
209 - Неправильное применение зашифрованной программы 345
210 - Неправильное имя подпрограммы 345
211 - Подпрограмма не существует 345
212 - Указанный файл имеет неправильный формат для подпрограммы 345
213 - Неправильный номер грани 345
214 - Исходный технологический элемент не применен 346
216 - Не удалось прочитать подпрограмму 346
217 - Не задано имя подпрограммы 346
218 - Создание кривой не применимо 346
219 - Выемка не применима 346
220 - Вращение не применимо 346
221 - Инверсия не применима 346
222 - Зеркальное отражение по x не применимо 346
223 - Зеркальное отражение по y не применимо 346
224 - Растягивание не применимо 347
225 - Запрограммированный инструмент: одна или несколько операций были исключены 347
226 - Чрезмерное число встроенных вызовов подпрограммы (максимум: 5) 347
227 - Клиентская ошибка номер <код клиентской ошибки> 347
228 - Невозможно задать шрифт (неправильное имя) 347
229 - Невозможно задать устройство для создания шрифта 347

12.8 Ошибки логических условий 348

230 - Число закрывающих ELSE или ENDIF больше числа открывающих IF 348
231 - Число закрывающих ENDIF меньше числа открывающих IF 348
232 - Неправильный код после открытого IF 348
233 - Число закрывающих ENDFOR больше числа открывающих FOR 348
234 - Число закрывающих ENDFOR меньше числа открывающих FOR 349
235 - Число команд FOR больше максимально допустимого (максимум: 500) 349
236 - Число итерраций при выполнении циклов FOR превышает максимально допустимое (максимум: 100000) 349
237 - Команда ENDIF используется для завершения цикла FOR 349
238 - Команда ENDFOR используется для завершения цикла IF 349

12.9 Ошибки при задании глобальных функций 349

239 - Команда ELSEIF используется в цикле IF после ELSE 349
240 - Имя клиентской функции не задано 350
241 - Недопустимое имя клиентской функции 350
242 - Ошибка исполнения клиентской функции: не заданы возвраты 350

12.10 Ошибки подготовки мультипрофилей 350

245 - Расчет мультипрофилей превышает максимальное число операций, которые можно задать для грани 350

12.11 Ошибки технологического задания профилей и точечных операций 350

250 - Невозможно применить подготовку к разомкнутому профилю из-за отсутствия исходного кода 350
251 - Невозможно применить технологическую точку из-за отсутствия исходного кода 350
252 - Невозможно задать открытые профили 351
253 - Была выполнена замена технологии 351
254 - Не удалось выполнить замену технологии 351

12.12 Ошибки задания начального/конечного отрезков профиля 351

271 – Вход/Выход профиля: невозможно построить трехмерную дугу 351
272 - Вход/Выход профиля: запрограммированная геометрия несовместима с запросом коррекции на инструмент 351
273 – Начало / Конец профиля: невозможно построить отрезок перекрытия, если профиль не замкнут 351

12.13 Ошибки при применении коррекции на инструмент 351

261 - Коррекция на инструмент: чрезмерная коррекция на радиус дуги 351
262 - Коррекция на инструмент: чрезмерная коррекция на отрезок 352
263 - Коррекция на инструмент: применяется сокращение к профилю 352
265 - Коррекция на инструмент: ошибка из-за коррекции в плоскости, отличной от xy, с вариантом пересечения отрезков 352
266 - Коррекция на инструмент: ошибка из-за коррекции в плоскости, отличной от xy 352
267 - Коррекция на инструмент: инверсия поправки должна устранить проблему пересечения или возобновить отрезок после прерывания 353
268 - Коррекция на инструмент: была запрошена приостановка коррекции без дальнейшего возобновления 353
269 - Коррекция на инструмент: приостановка и дальнейшее возобновление коррекции не помогает выполнить скругление 353
270 - Коррекция на инструмент: приостановка и дальнейшее возобновление коррекции должно обеспечить проверку геометрической непрерывности отрезков 353

12.14 Ошибки фрагментации и линеаризации дуг в плоскостях, отличных от xy 353

255 - При линеаризации трехмерных дуг превышено максимальное число линий 353
256 - Линеаризация трехмерных дуг невозможна из-за отсутствия исходного линейного кода 353

13 Индивидуализация TpaCAD 354

13.1 Среда 354
Запуск 354
Действие 354
Редактирование операции 355
Сохранение 357
Импорт формата 358

13.2 Цвета 359
Графическое представление 359
Слой 361
Конструкция 361
Поле О 362
Раскрой 362

13.3 Виды 362
Индивидуализировать виды 362
Индивидуализировать графическое представление 365
Сетка и Рисунки 369
Мышь 371

13.4 Технология 371
Коды по умолчанию 372
Технология по умолчанию 372

13.5 Персонализировать файл "прототип" 373

14 Создание клиентских операций 374

15 Программы преобразования 376

15.1 Из формата DXF в формат TpaCAD 376

15.2 Из формата TpaCAD в формат DXF 376
Параметры 376
Операции и Слои 376
Запрограммированные операции 377
Точечные операции (рабочий код в интервале: 1-1000) 377
Операции подготовки (рабочий код в интервале: 1-1000) 377
Операции профиля линейного типа 378
Операция профиля типа дуги (плоскость xy) 378
Операция профиля типа дуги (плоскость не xy) 378

15.3 Из формата ISO в формат TpaCAD 378
Настройки 380

15.4 Из формата TpaCAD в формат ISO 381
Настройки 381
Синтаксис и примеры 383
Фрезерование 383
Опрация сверления 385

15.5 Из формата TpaCAD в формат Edicad 385
Режим перекодировки 386
Общая информация о детали 386
Запрограммированные операции 386
Точечные 386
Клиентские логики 387
Подготовка 387

Профиль 388

15.6 Программа TpaCAD 390

Строки заголовка 390
Раздел программы грани 391
Раздел операции, заданной в программе грани 391
Операция: Отверстие 391
Операция: Подготовка фрезы 393
Операция: Линия 393
Операция: Дуга на плоскости для грани 394

1 Введение

1.1 Что это такое



TpaCAD версия 2.4.28

TpaCAD представляет собой систему CAD/CAM, реализованную в графической среде, позволяющую создавать, менять и импортировать рабочие программы и разрабатывать персонализированные макро- и подпрограммы и предназначенную для программирования станков с числовым программным управлением для обработки дерева, металла, мрамора и пластмасс.

Рабочая зона содержит меню и многофункциональные панели управления (Лента), организованные таким образом, чтобы создать среду для простого создания рабочей программы.
Рабочая программа организована в виде списков операций, организованных по грани применения.
Базовая геометрия, на которой определяется программа, - параллелепипед, заданный тремя размерами (длина, высота и толщина) и шестью гранями применения. К этой базовой геометрии можно добавить вообще составные и ориентированные плоскости (переменные геометрии: поские, кривые и поверхности) и ассоциировать с каждой из них список операций.

Рабочая программа представлена как в графическом, так и в текстовом формате с немедленным взаимодействием между этими двумя представлениями.
Представление в графическом формате может выполняться в двух- или трехмерном изображении или в виде развертки детали на плоскости (каркасное представление), где двухмерное графическое изображение или каркасное представление позволяют отображать операции на плоскости каждой отдельной грани, а трехмерное графическое изображение дает общий вид операций на детали.
Представление может поворачиваться (по трем плоскостям вращения, независимым между собой), расширяться, уменьшаться (с зумом на нескольких уровнях) или центроваться на основе потребностей.

Графическое представление позволяет выполнять интерактивный выбор операции или группы операций с возможностью подключения многочисленных фильтров отображения. Используются различные вспомогательные средства:

- курсор в виде креста
- линейка
- сетка с постоянным шагом или с отдельными элементами с возможностью индивидуализации элементов сетки (решетка, вершины и т.д.)

Представление в **текстовом** формате позволяет иметь структурное отображение программы грани. Он включает все запрограммированные блоки, включая те, которые не имеют заданного графического представления:

- блоки логических ограничений (IF.. ELSEIF.. ELSE.. ENDIF)
- блоки, соответствующие запрограммированным погрешностям
- задания локальных переменных
- комментируемые блоки

В частности, текст программы представлен с отступами для выделения структуры заданной вставленными логическими условиями.
Текстовый формат - это представление программы в ASCII, допускающий:

- отдельный и множественный выбор
- в случае сложных операций (подпрограмм или макропрограмм) - отображение отдельных операций, отвечающих ее развитию
- в случае сложных операций, применимых к нескольким граням (вложенные вызовы), - отображение перечней операций, соответствующих созданию каждой грани

Операции могут вводиться выбором из графической палитры с вводом геометрических элементов и применением инструментов CAD, например, написание текстов и выемка замкнутых зон.
Их изменение может осуществляться:

- воздействуя напрямую на отдельную операцию
- применяя изменения, общие для группы операций
- применяя геометрические трансформации к группе операций (перенос, симметрии, повторения)
- применяя инструменты манипулирования профилями (масштаб, инверсия, прерывания, соединения, коррекция на инструмент)

Для целенаправленного манипулирования рабочей программой предусматриваются многочисленные **инструменты**:

- общие инструменты: перенос, вращение, симметрии, повторения последовательно на установленных маршрутах, разбивка подпрограмм или макрокоманды
- инструменты профиля: инверсия, масштаб, применение технологии, соединения профилей, изменение вершин, применение открытия или закрытия, фрагментация и минимизация, прерывание, расширение профилей
- инструменты CAD: создание текстов, выемка зоны, резка профилей, создание кривых сплайна из полилиний, раскрой форм, использование операций, применяющих геометрические трансформации.

Особым аспектом задания операций является **параметрическое программирование**, позволяющее использовать:

- переменные детали
- математические, геометрические, статистические, логические функции и функции работы со строками
- технологические функции

Параметрическое программирование может использоваться при задании программных переменных, переменных геометрий и параметров операций.
Важно подчеркнуть возможность задания клиентских функций. Это функции, которые обрабатывают логику расчета, определенную исходя из потребностей клиента и которые затем могут использоваться на любом уровне программирования.

Функции и переменные аргументы, предусмотренные в параметрическом программировании, обеспечивают возможность полного управления контекстом, в котором работают TpaCAD и каждая отдельная рабочая программа:

- технология
- задания, связанные с конфигурацией
- режим выполнения
- задание геометрических характеристик детали

Особенно эффективна универсальная функция геометрической библиотеки, предоставляющая немедленное решение проблемам геометрического характера, в т. ч. повышенной сложности.
Большое число функций и переменных аргументов, предусмотренных в параметрическом программировании, позволило реализовать **контекстуальную помощь** на этапе программирования:

- можно выбрать функцию (или переменный аргумент) из упорядоченного списка
- можно запросить отображение помощи, связанной с синтаксисом вызова функции

Общие задания детали допускают повышенную степень ограничений **исполнительного состава** рабочей программы. Таким образом, всего одна зарегистрированная программа может генерировать неограниченное число различных рабочих схем на станках с различными геометрическими характеристиками (размерами, заданием рабочих граней), режимами выполнения (нормальный, в зеркальном отражении), исключениями, повторными заданиями программных переменных (сдвигов, переменных цикла).

Функциональность Раскроя панелей и / или форм предлагает систему раскроя, интегрированную со средой программирования отдельных частей:

- с динамическим открытием списков раскроя
- с автоматическим распознаванием программ, созданных с помощью CAD Tpa
- с прямым импортом из других форматов (g-код, DXF)
- с прямым созданием отчетов, производственных файлов, этикетов.

1.2 Подключение и режимы функционирования

Программа имеет три рабочих режима:

- **Essential** (Существенный)
- **Standard** (Стандартный)
- **Professional** (Профессиональный)

Режим **Essential** соответствует минимальному уровню функционирования. Этот режим не имеет прямого соответствия в предыдущих версиях продукта (т. е. TpaEdi32) и предназначен для предоставления среды, специально предназначенной для редактора.

Standard режим соответствует промежуточному уровню функционирования, который раньше определялся как базовое функционирование.

Режим **Professional** ответствует продвинутому уровню функционирования. По сравнению с базовым уровнем функционирования в нем имеются следующие дополнительные команды и функции:

- создание текстов
- функция задания и применения клиентских шрифтов
- выемки зоны
- создание кривых сплайн
- инструменты резки и построения профилей
- утилита проставления размеров
- задание фиктивных граней с указанием исходной грани
- функция, дополненная запрограммированными вложенными вызовами
- инструмент создания фиктивных граней из запрограммированной геометрии
- задание автоматических граней
- распространение параметрического программирования с добавлением клиентских функций и кодов глобальных функций
- введение операций, применяющих геометрические трансформации (коды STOOL)
- расширение функции коррекции на инструмент (приостановка и смена стороны при коррекции профиля)
- полнота программирования Критериев отображения на этапе Печати
- функция задания Закладок
- функция привязки между элементами, заданными на различных гранях, подключенная в интерактивных процедурах
- автоматическое преобразование в другие форматы файла на этапе архивирования программы
- возможность добавления клиентской типологии файлов при считывании программы.

Команды, активизированные только в режиме **Professional,** обозначаются в данном руководстве символом



К режиму Профессиональный могут затем добавляться определенные функции:
- задание неплоских рабочих граней (закругленных граней)
- задание моделирования детали для экструзии
- задание составных рабочих граней (поверхностей).

Работа TpaCAD защищается аппаратным ключом USB исходя из запросов производителя станка. Аппаратный ключ можно перемещать с одного компьютера на другой, что позволяет работать в режиме "Professional", "Standard" или "Essential" на различных инсталляциях TpaCAD, естественно, не одновременно. Наличие ключа проверяется при каждом запросе исполнения особых команд.

ВНИМАНИЕ: аппаратный ключ может программироваться таким образом, чтобы позволить использование пакета инсталляции TpaCAD, но с ограничением использования только в среде исполнения. В этом случае запуск TpaCAD не удается (сообщение на английском языке указывает, что ключ исключает запуск программы). Это функция: **Вне сети**.
Ключ, запрограммированный **Вне сети,** может распознавать один из трех указанных выше режимов: Professional, Standard или Essential.

Если ключ подключения не активируется, подключается **Демонстрационный** режим, имеющий неограниченный срок действия. При подключении демонстрационного режима специальное окно предупреждает, что выполненная инсталляция не позволяет запускать полную функцию.

Если это окно отображается и при наличии правильно запрограммированного аппаратного ключа, это означает неправильное функционирование:
- аппаратный ключ не считывается правильно или вставлен в несоответствующий порт.

В этом случае выполнить необходимые проверки.

Демонстрационный режим ограничивает некоторые функции приложения, в частности:
- всегда остается активированным минимальный уровень доступа;
- невозможно сохранять программы;
- невозможно оптимизировать программы;
- исключается создание, правка и удаление операций пользователя;
- исключается создание, правка и удаление клиентских шрифтов;
- исключена возможность изменения установки.

По умолчанию **Демонстрационный** режим работает в режиме Professional, подключая другие особые функции, в т. ч. для продвинутого применения:
- фиктивные грани, в т. ч. в определении закругленных граней и поверхностей
- моделирование
- использование клиентского шрифта
- использование базовых клиентских функций.

Можно также подключить режим Standard или режим Essential для оценки различий между различными режимами. В меню имеется команда **ДЕМОНСТРАЦИОННЫЙ режим** , который подключен только в Демонстрационном режиме и при закрытой программе.

В таблице приводится сравнение этих трех рабочих режимов:

	Professional	Standard	Essential
База данных операций			
• Полное использование подписанных изготовителем баз данных	**v**		
• Использование нефирменных клиентских баз данных	**v**	**v**	
• Изменение нефирменной клиентской базы данных	**v**	**v**	
• Управление базой данных клиентских операций	**v**	**v**	**v**
Управление Конфигурацией			
• Изменение конфигурации TpaCAD	**v**	**v**	**v**
• Определение и Изменение клиентских функций, персонализированных изготовителем	**v**		
• Выбор установки	**v**	**v**	
• Определение глобальных переменных	**v**	**v**	
• Управление Продвинутой конфигурацией	**v**		
Конфигурация детали			
• Геометрия детали в абсолютной системе	**v**	**v**	
• Задание фиктивных граней в локальных системах	**v**		
• Присвоение параметров фиктивным граням	**v**	**v**	
• Задание закругленных граней (фиктивных, автоматических)	**v** Θ		
• Задание поверхностей	**v** Θ		
• Моделирование для экструзии	**v** Θ		
• Последовательность операций	**v**	V	
• Максимальная персонализация персонализируемых разделов	**v**	**v**	[примечание 1]
• Возможность блокировки изменения разделов программы	**v**	V	
• Управление гранью-деталью	**v**	**v**	**v**
Открытие / Создание программы			
• Открыть программы в другом формате (импорт: DXF, ISO,..)	**v**	**v**	**v**
• Персонализировать отдельный импорт программы	**v**	**v**	
• Открыть базовые макропрограммы	**v**	**v**	
• Создание клиентских макропрограмм	**v**	**v**	
• Создание защищенных программ или подпрограмм	**v**	**v**	**v**

[примечание 1] функция Essential исключает управление клиентскими разделами, называемыми: Раздел дополнительной информации, Раздел привязок

	Professional	Standard	Essential
• Добавление клиентского расширения для открытия программ	**v**		
• Максимальный размер программ (число программируемых строк)	**v**	**v**	[примечание 1]
• Команда преобразования Архива программ	**v**	**v**	**v**

[примечание 1] функция Essential ограничивает число программируемых операций 100000 (на грань)

Сохранение программы

• Регистрация программы в другом формате (DXF, ISO,..)	**v**		
• Регистрация клиентских макропрограмм	**v**	**v**	

Печать программы

• Наличие программируемых опций	**v**	**v**	
• Максимальный контроль программируемых опций	**v**		

Общие утилиты программы

• Полнота общих команд изменения программы (меню Редактирование)	**v**	**v**	[примечание 1]
• Максимальная индивидуализация фильтров отображения и изменения	**v**	**v**	[примечание 2]
• Логические исключения	**v**	**v**	
• Создание фиктивных граней из программируемых геометрий	**v**		
• Создание поверхностей из программируемых геометрий	**v** Θ		
• Создание моделирования из программируемых геометрий	**v** Θ		
• Создание клиентских шрифтов из программируемых геометрий	**v**		
• Операции измерения на детали	**v**	**v**	**v**
• Создание проставлений размеров	**v**		
• Предварительный просмотр для оптимизации	**v**	**v**	
• Общие трансформации программы	**v**		

[примечание 1] (меню: Редактирование, группа: Задать) функция Essential исключает команду Задать Специальные фильтры

[примечание 2] (меню: Редактирование, группа: Изменить) функция Essential исключает команду Решить

Коды Операций

• Коды применения кривых ISO	**v**	**v**	
• Коды выемки замкнутых зон	**v**		
• Коды генерирования текстов с использованием системного шрифта	**v**		

	Professional	Standard	Essential
• Коды генерирования текстов с использованием клиентского шрифта	**v**		
• Коды программирования инструментов (STOOL)	**v**		
• Дополнительные коды задания логических циклов (глобальные функции)	**v**		
• Код программирования наклонных плоскостей (автоматические грани)	**v**		
Индивидуализация операций			
• Подключение "стандартных" свойств (C, L, B, O, M, K)	**v**	**v**	**v**
• Подключение "дополнительных" свойств (N, K1, K2, V)	**v**	**v**	
• Возможность блокировки особых значений свойств	**v**	V	
• Параметрическое программирование свойств	**v**	**v**	
• Функции, добавленные в коррекцию на инструмент (приостановка и смена стороны)	**v**		
• Функция выполнения острых краев в профилях	**v**		
• Развертка запрограммированных вложенных вызовов (в грани-детали)	**v**	**v**	
• Развертка запрограммированных вложенных вызовов (во всех гранях)	**v**		
• Возможность вызова макропрограммы из общих кодов SUB	**v**	**v**	
Функция чертежа			
• Изменение текущей операции интерактивной процедурой	**v**	**v**	
• Меню Чертеж (точки, дуги, круги, эллипсы, полилинии)	**v**	**v**	**v**
• Использование закладок	**v**		
• Привязка между гранями	**v**		
• Привязка к элементам моделирования	**v** Θ		
Инструменты и построения			
• Полнота общих инструментов	**v**	**v**	[примечание 1]
• Полнота общих инструментов изменения профилей	**v**	**v**	[примечание 2]
• Применение сдвига к профилям	**v**	**v**	
• Применение точек крепления к профилям	**v**	**v**	
• Применение подачи глубины к профилям	**v**	**v**	
• Поворот профилей на плоскостях координат	**v**	**v**	
• Инструменты резки и селективного построения профилей	**v**		
• Выемка зоны	**v**		
• Генерирование текстов с использованием системного шрифта	**v**		

	Professional	Standard	Essential
• Генерирование текстов с использованием клиентского шрифта	**v**		
• Создание кривых сплайн	**v**		
• Инструменты "Раскрой профилей"	**v** Θ		
[примечание 1] (меню: Инструменты, группа: Общие) функция Essential исключает команды: Общая симметрия, Повторение на профиле, Разбить			
[примечание 2] (меню: Инструменты, группа: Изменить профили) функция Essential исключает команды: Растянуть профиль, Продлить			
Параметрическое программирование			
• Использование имени операций	**v**	**v**	
• Использование клиентских функций	**v**		
Графические опции			
• Управление сеткой по точкам (клиентская сетка)	**v**	**v**	

Условные обозначения:
v реализованный пункт
Θ требует подключения дополнительного аппаратного устройства

1.3 Право доступа к системе

Помимо рабочего режима, определенного ключом подключения, TpaCAD предусматривает различные режимы доступа к системе.

- **Пользователь** - это уровень с максимальными ограничениями доступа. Нельзя изменить ни одну защищенную настройку, открывать или изменять программы макрокоманд, открывать или изменять файлы, задающие операции, используемые в программе TpaCAD. При запуске TpaCAD запускается этот режим доступа.
- **Сервис** используется для задания уровня доступа или изменения программы. Этот уровень позволяет также изменять ограниченную часть конфигурации TpaCAD. Он выше уровня пользователя.
- **Изготовитель** - уровень, который используется для конфигурации TpaCAD и служит для записи программ типа макро. На этом уровне могут вноситься любые возможные изменения.

Доступ к каждому режиму защищается соответствующим паролем. Уровень Пользователь - самый низкий уровень доступа, а уровень Изготовитель - самый высокий.
Для доступа к требуемому уровню выполнить следующие операции:

1. при открытом как минимум одном приложении TpaCAD нажать сочетание клавиш **[CTRL+* (звездочка)]**. Открывается окно для ввода пароля, соответствующего уровню. В качестве альтернативы в правой части строки приложений Windows ® имеется значок . При щелчке правой кнопкой мыши на этом значке можно отобразить меню, в котором имеется пункт **Изменить уровень пароля**.
2. ввести пароль необходимого уровня и нажать кнопку **[OK]**.

Если введенный пароль неправилен, отображается сообщение об ошибке "Внимание! Неправильный пароль!!!".
После ввода пароля вы сразу попадаете на соответствующий уровень доступа.
Уровень доступа, выбранный описанным выше образом, является общим для среды TPA, инсталлированной и работающей на компьютере.

ВНИМАНИЕ: в TpaCAD может подключаться локальный режим для изменения уровня доступа при подключении функции **Автономный режим** (из конфигурации). В частности, этот второй режим:

- является дополнением к уже описанному режиму и не заменяет его. То есть: процедура, подключаемая сочетанием клавиш **[CTRL+* (звездочка)]** остается функциональной и в TpaCAD;
- имеется только если активна функция **Автономный режим** (из конфигурации) и если изготовитель станка подключил локальную учетную запись в среде TpaCAD;
- изготовитель должен выдать клиенту подключенную учетную запись (пароль);
- позволяет подключать учетную запись на уровне **Изготовитель** локально в TpaCAD. Это означает, что выбранный здесь уровень доступа не работает в установленной и работающей на компьютере среде TPA.

Команда **Уровень пароля** выбирается из меню . При вводе правильного пароля оператор может изменить его, персонализируя привилегированный доступ к своей инсталляции системы CAD.

ВНИМАНИЕ: Изготовитель машины может активировать локальную учетную запись в среде TpaCAD только при доступе к этой команде с уровня **Изготовитель,** признанного для среды TPA, активированной при помощи:
- сочетание клавиш **[CTRL+* (звездочка)]**; или
- команды на строке приложений Windows или
- той же командой **Уровень пароля,** выбранной из меню TpaCAD.

1.4 Многоязыковая поддержка

TpaCAD поддерживает отображение текста на различных языках. В качестве основы поставляется полная поддержка на 10 языках:

- Итальянский
- Английский
- Французский
- Немецкий
- Испанский
- Чешский
- Русский
- Голландский
- Польский
- Китайский

Для этих языков предусматривается переведенная справка в контекстуальном печатаемом варианте.

Для другой группы языков предусматривается перевод только сообщений интерфейса:

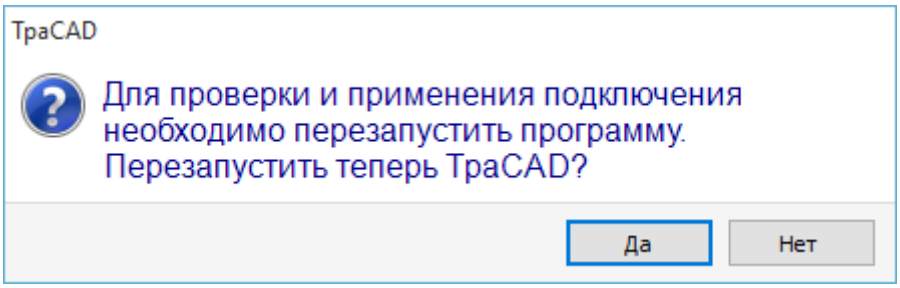
- Иврит
- Венгерский
- Японский
- Латышский
- Словенский
- Болгарский
- Румынский
- Португальский
- Шведский
- Турецкий

При выпуске каждой новой версии не гарантируется полнота и соответствие обновления перевода сообщений и руководств. Сообщения, еще не переведенные на определенный язык, в любом случае, представляются на английском языке.

Можно также расширить поддержку на другие языки исходя из необходимости. Некоторые ограничения обуславливаются языками, например, восточными, для которых требуется использовать определенный набор символов или которые не пишутся слева->направо, что характерно для западных языков.

Переключение языка
Переключение языка может осуществляться на любом уровне доступа в систему. Для изменения выбранного языка необходимо использовать сочетание клавиш **[CTRL** + **/]** или щелкнуть значок в "**строке приложений**" Windows ®.
В открывающемся окне выбрать нужный язык и щелкнуть кнопку **[OK]**.

ВНИМАНИЕ: если TpaCAD используется не в среде TPA, изменение языка доступно локально в приложении при подключении функции **Автономный режим** (из конфигурации). В частности:
- в строке состояния TpaCAD добавлен и подключен перечень выбора доступных языков. Даже в неподключенном состоянии ячейка в строке состояния указывает обозначение активного языка. При запросе смены языка при закрытой программе можно подтвердить прямой перезапуск TpaCAD, как предлагается в следующем окне:



в противном случае язык меняется не сразу, а при следующем запуске TpaCAD.

1.5 Совместимость формата

Формат программ TpaCAD несовместим с предыдущими версиями CAD TPA (TpaEdi32, Edicad). TpaCAD, тем не менее, может читать программы предыдущих версий.
Для создания программ при помощи TpaCAD в версии, совместимой с TpaEdi32, необходимо использовать команду Сохранения в нужном формате.

1.6 Новинки

Ниже приводится перечень самых важных новинок, внесенных в TpaCAD по сравнению с предыдущей версией программы, TpaEdi32.
Полностью измененный графический интерфейс - это, несомненно, самое очевидное отличие, однако отличия носят не только графический характер. Рассмотрим его основные аспекты:

- использование Ленты, заменившей классическое меню, обусловило полный пересмотр группы команд. Вкладок Ленты меньше по сравнению с предыдущими основными пунктами меню
- некоторые средства управления могут перемещаться в пределах графической зоны приложения. Например, зону данных операции можно переместить вправо или влево;
- зона, зарезервированная для отображения команд и ошибок, теперь имеет структуру группы обновляющихся автоматически вкладок, которая содержит также таблицу программных переменных <j> и окно отладки, что в TpaEdi32 реагируют на диалоговые окна, выбираемые в меню
- зона, соответствующая комплексным заданиям программы (размеры, переменные и т.д.), остается структурированной в постоянно отображаемую группу вкладок. При этом можно видеть размеры или переменные или задание клиентского раздела, в т. ч. на этапе задания операций на грани детали
- зона, соответствующая тексту ASCII программы, теперь может расширяться и могут меняться ее размеры: можно “закрывать” текст ASCII и снова открывать его при необходимости, например, для задания ветвей логических условий
- полностью обновлена зона графического представления детали. Эффекты прозрачности и сияния, помимо возможности задания фонового изображения детали и/или текущей грани, которое адаптируется к соответствующей зоне, делают отображение более соответствующим для трехмерного представления детали и материала, из которого изготовлена деталь (дерево, стекло и т. д.)
- зона задания отдельной программной строки (операция) дополнена командной строкой, которая позволяет прокручивать список операций, быстро перемещать программную строку, вставлять копию текущей операции, изменять параметры той же области задания
- в интерактивных зонах реализованы специальные команды, подключаемые из контекстуальных меню (отображаются при щелчке правой кнопкой мыши на соответствующей зоне), составленных на основе рабочего контекста. Этот аспект придает однородность функционированию в различных зонах и позволяет также управлять большим числом персонализаций в самих меню управления, например, зона графического представления радикально меняет локальное меню с подключением интерактивных процедур
- строка выбора операций полностью обновлена и имеет широкие возможности настройки.

Выполнено множество функциональных изменений:

- программа в исходном формате TpaCAD или совместимом с ним со сконфигурированным одним из модулей импорта, может напрямую открываться методом буксировки в графическую зону TpaCAD
- окно открытия программ содержит все варианты выбора, возможные на этапе загрузки детали:
  - подключение модуля импорта и соответствующие дополнительные задания
  - открытие программы в исходном формате в виде копии
- список последних программ увеличен по количеству и оснащен локальным меню
- число конфигурируемых модулей импорта и экспорта увеличено до 8

- графическое представление детали, дополнительно к традиционным трех- (3D) и двухмерным (2D) изображениям (вид на плоскости XY грани), позволяет видеть развертку на базовой плоскости граней параллелепипеда (каркасное представление)
- графическое представление отображает операции, которые в целом заданы на детали, в т. ч. на виде грани: различные цвета позволяют отличить текущий перечень операций от других
- графическое представление действует также на уровне прямого изменения вида грани при прямом щелчке в зоне габарита грани
- графическое представление профиля может выбрать напрямую в меню между вариантами отображения различных размеров (горизонтальные, вертикальные, функции обрезки). Например: сплошным или линейным отрезком, удовлетворяя потребность, например, *видеть* как зону с выполненной выемкой (сплошным отрезком), так и габарит отдельного профиля
- при задании операций добавлена возможность интегрирования прямых заданий с интерактивным считыванием координат (координаты точки применения или центра вращения или вспомогательной точки)
- разрешение вложенных вызовов теперь интегрировано в управление главным вызовом с соответствующим удалением вложенных строк в тексте программы и ускорением понимания этой функции. Расширение, требуемое на главном вызове, позволяет напрямую определять операции, заданные для добавленных вызовов
- была значительно усовершенствована функция прямого задания Кодов клиента применения подпрограмм, позволяя конечному пользователю TpaCAD создавать собственную базу данных операций, которая может охватить значительную часть его потребностей персонализации
- выбор операций сохраняется также при изменении активированного вида, позволяя применять сложные программные инструменты (задание технологии; сокращение, фрагментация, линеаризация и соединение профилей), также с учетом активированных выборов, помимо случаев применения активных фильтров вида или изменения
- команды поиска и замены элементов операций распространяются на всю программу и добавляется возможность рассмотрения соответствия и/или замены также свойств. Замена параметрических форм теперь может выполнять как селективные замены переменных или переменных аргументов, так и общие замены подстрок.
- команды комплексного задания свойств позволяют также напрямую задавать параметрическое задание
- новые команды позволяют вводить непосредственно в программный список уже определенные структуры логических циклов: IF..ENDIF, IF..ELSEIF..ELSE..ENDIF с возможностью прямой связи логической структуры с группой операций
- меню справки параметрического программирования всегда можно вызывать нажатием правой кнопки мыши; оно предусматривает возможность доступа ко всем программным переменным (помимо 'r', также к 'o', 'v'), а также напрямую к информации по технологии
- выбор инструментов унифицирован в плане способов задания необходимой информации: непосредственно из этого окна можно интегрировать прямые задания другими, полученными в интерактивном режиме. Это, помимо упрощения, позволило повысить мощность многочисленных инструментов.
- при подключении интерактивных процедур сохраняется активизированный вид (двух-, трехмерный или каркасное представление), уделяется также особое внимание справке при выполнении процедуры за счет предоставления кратких, но функциональных подсказок непосредственно в графической зоне и пояснительных сообщений в зоне команд
- режимы привязки, подключаемые при выполнении интерактивных процедур, дают возможность учесть:
  - глубину элементов привязки
  - элементы, заданные на различных гранях,
  - соответствующим образом добавленные графические ориентиры (маркеры, закладки), использование которых позволяет распространить привязку не только на все запрограммированные на детали объекты, но и на дополнительные соответствующим образом определенные объекты
  - привязка к сетке точками или горизонтальными / вертикальными линиями

  Привязка между гранями позволяет, например:
  - позиционировать отверстие в грани 3 (передней) "на" отверстие в грани 1 (верхней)
  - измерять расстояние между элементами, запрограммированными на различных гранях
- задание последовательностей управляет, помимо выбора зоны, также определением и графическим выбором операции. Кроме того, этот список содержит операции конструкции
- более многочисленные и подробные конфигурации позволяют предусмотреть окна задания фильтров так, чтобы обеспечить полное удовлетворение конкретных потребностей каждого отдельного применения
- задание локальных параметров среде TpaCAD позволяет более эффективно использовать как в автономном режиме, так и интегрировано в среду TPA
- специальные тесты и разработки направлены на обработку "больших программ" (например, количество запрограммированных строк свыше 250000).

Рассмотрим изменения, касающиеся непосредственно программы:
- на этапе редактора отображаются некоторые стандартные свойства файла и дополнительные персонализированные свойства программы с возможностью конфигурации состояния изменения каждого раздела конкретной программы. Кроме того, можно принудительно задать предпочтительную единицу измерения программы. Добавление управления порядковым номером сохранения позволяет оптимизировать создание и поддержание оптимизированных файлов.

- режимы выполнения записаны в память программы с возможностью задания умолчания особого выполнения
- максимальное число переменных типа <o> и <v> увеличено до 16 (с 8)
- программирование фиктивных граней добавляет возможность назначать криволинейные плоскости и/или поверхности
- максимальное число дополнительных параметров, определенных при задании фиктивной грани, увеличено до 5 (с 3)
- можно задать моделирование детали, выбрав один из трех различных режимов (специальный раздел, назначение в переменной геометрии или в запрограммированной обработке)
- свойствам операций добавлены два новых поля, называемые K1 и K2, максимальное задаваемое значение для которых составляет 255. Для новых полей можно задавать фильтры отображения и блокировки изменения, а также исключения программы
- при открытии программы, написанной в формате Edicad или TpaEdi32, свойства K, K1 и K2 могут восстановить задания, заданные ранее в параметрах операции: это улучшает структурирование операций с оптимальным использованием ставших теперь доступными свойств
- управление структурами IF...ELSE...ENDIF добавляет команду ELSEIF
- новый рабочий код (EXIT) позволяет усовершенствовать контроль за выполнением цикла IF..ELSE..ENDIF, позволяя программировать прямой выход из логического цикла или из потока программы
- программирование в режиме "грань-деталь" дополнено, например, возможностью вызова применения операции на автоматической грани с указанием имени грани
- многочисленные новые коды типа STOOL (операции запрограммированных инструментов) значительно улучшают интеграцию собственных возможностей инструментов с возможностями операций. Можно изменить или создать профили для:
    - поправки на инструмент
    - применения скруглений или точек крепления
    - фрагментации и линеаризации
    - линеаризации или подачи в развертку глубины
    - соединение профилей

  Функционирование на основе новых кодов аналогично функционированию соответствующих продвинутых инструментов профиля с тем преимуществом, что созданные профили адаптируются к изменениям исходных профилей, кроме того, можно работать в общей сложности на нескольких профилях.
- функция создания кривых сплайн добавляет управление *Кардинальным сплайном* и *Трассировками*
- добавлена функция задания и применения клиентских шрифтов.

1.7 Системные требования

Перед инсталляцией TpaCAD проверить, что компьютер отвечает минимальным требованиям по аппаратному и программному обеспечению.
Требуется операционная система Microsoft® Windows® 7 Enterprise, Ultimate, Professional или Home Premium; Microsoft® Windows Vista® Enterprise, Business, Ultimate или Home Premium (SP1 или следующая версия); или Microsoft® Windows® XP Professional или Home edition (SP2 или следующая версия), Microsoft® Windows® 8, Microsoft® Windows® 10.
Требуется минимальная память ОЗУ 4 Гб.
Для инсталляции требуется 500 Мб свободного пространства на жестком диске.
Минимальные требования для платы видео и монитора: 1024x768 32-бит цветная с ОЗУ 256 Мб.
Для оптимизации характеристик TpaCAD рекомендуется проверять и обновлять драйверы графических плат компьютера после инсталляции.

TpaCAD использует программное обеспечение OpenGL и требует версию не ниже 1.2: если на компьютере не установлена как минимум эта версия, приложение представляет предупредительное сообщение и закрывается.

Выполнение TpaCAD на виртуальных станках

Не гарантируется выполнение TpaCAD на виртуальных станках. Выполнение TpaCAD на таких станках, но не с родной средой Windows, может обуславливать снижение качества графического отображения и общее замедление работы.

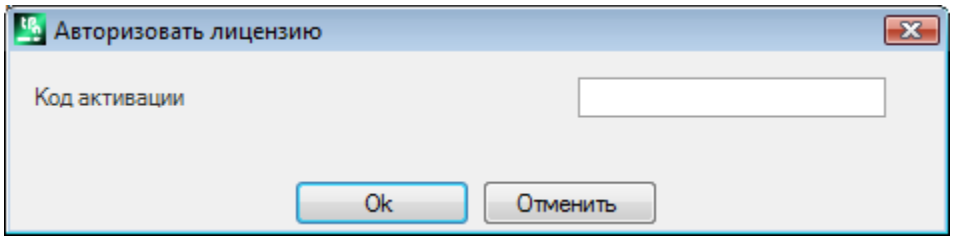
Проблемы характеристик зависят от типа виртуального станка и присвоенной конфигурации.

1.8 Контроль сигнальных сообщений при запуске TpaCAD

При запуске TpaCAD могут диагностироваться некоторые ситуации, не допускающие ее функционирования. Для всех этих ситуаций отображается сообщение на английском языке и программа завершается. Рассмотрим возможные случаи:

- **"Critical error!!":** сигнализирует ситуацию серьезной ошибки, произошедшей на этапе инициализации. Эта ошибка может происходить в результате неправильной инсталляции пакета TpaCAD или вследствие повреждения файлов, необходимых для функционирования приложения, как файлов кодов, так и файлов данных. Рекомендуется выполнить полную инсталляцию.
- "**Error in configuration loading:..**": сигнализирует ситуацию ошибки, произошедшей на этапе загрузки файла конфигурации TpaCAD. Сообщение об ошибке может означать конкретный сигнал, например, отсутствие файла или ошибку доступа или ошибку формата в самом файле. В зависимости от типа ошибки TpaCAD может предложить метод устранения ошибки, например, задание конфигурации по умолчанию или восстановление конфигурации из резервной копии. Эта ошибка может указывать на удаление или повреждение файла конфигурации.
- **"OpenGl: unsupported version (min: 1.2) !!":** инсталлированная версия программного обеспечения OpenGL несовместима с приложением. Необходимо выполнить обновление по меньшей мере до версии 1.2. Этот сигнал типичен для работы на виртуальных машинах, которые часто используют очень сильно устаревшие версии программы OpenGL.
- **"It's not checked the license to use TpaCAD !!":** была распознана лицензия для использования пакета TpaCAD "Офлайн".
- **"Too many instances !!":** уже запущено 4 (четыре) копий приложения.

При запуске TpaCAD после персонализированной инсталляции пакета может быть представлено окно:



Это окно указывает, что при выполненной инсталляции была скопирована клиентская база данных операций, для подключении которой требуется определенная процедура. База данных является фирменной базой производителя станка и требуется также код активации, задаваемый при запуске программы. Код активации должен сообщаться производителем и задаваться при каждой инсталляции: например, при инсталляции пакета на два различных компьютера в обоих случаях должна выполняться операция активации.
Операцию активации можно выполнить и в более поздние сроки, однако это может ограничить возможности использования пакета.

- закрыть окно после задания требуемого кода для полной активации лицензии на применение. В этом случае специальное сообщение предупреждает, что программа TpaCAD закрывается: снова запустить TpaCAD для проверки того, что операция завершилась правильно. Если при запуске окно запроса разрешения появляется снова, обратиться к изготовителю станка;
- закрыть окно без задания кода для откладывания этой операции: при последующем запуске TpaCAD это окно будет представлено снова.

Если активация лицензии выполнена неправильно, база данных клиентских операций все равно загружается, однако она будет работать неполностью.

При запуске TpaCAD могут возникать ситуации с частичным или полным нарушением некоторых функций. В этом случае специальное окно отображает список выявленных отклонений. Этот список можно также вызывать из меню **->Проверить сигнальные сообщения.**

Сигнальные сообщения могут касаться следующих случаев:

- не удалось подключить управление аппаратным ключом USB: TpaCAD функционирует только в демонстрационном режиме
- подключение демонстрационного режима после нераспознавания аппаратного ключа USB с соответствующими ограничениями в работе приложения

- не удалось использовать управление уровнями доступа с соответствующим функционированием только на уровне оператора
- не удалось использовать управление сообщениями с соответствующим отображением сообщений полностью или частично на итальянском языке.
- необходимость активации лицензии на использования клиентской базы данных операций.

окно сигналов можно вызвать, даже если не были поданы сигналы, для просмотра подключений определенных параметров комплекта TpaCAD, активных на аппаратном ключе.

1.9 Справка

Инсталляция TpaCAD предусматривает несколько руководств, доступных в виде справки онлайн, напрямую вызываемых из программы (справка) и представленных в печатаемом формате в папке СПРАВКА.

Руководства всегда доступны на языках, указанных в инсталляции.
При инсталляции обновлений или новых версий программы руководства могут обновляться.

2 Версии и обновления

2.1 Версии 2.4.0 (22 мая 2020 г.) - 2.4.28 (03 февраля 2026 г.)

Версия 2.4.28 (03 февраля 2026 г.)

Новинки:
- Раскрой: в Конфигурацию раскроя добавлен флаг, позволяющий выбрать, следует ли создавать папки с этикетками для отходов и бракованных изделий в качестве подпапок или на том же уровне, что и папка LABEL.
- Раскрой: добавлена возможность управления зигзагообразным вводом также в предварительных разрезах при выполнении с помощью CUTRECT.

Версия 2.4.27 (12 декабря 2025 г.)

Новинки:
- Раскрой: в мастере создания этикеток для отходов и перерабатываемых тарных знаков отображается только полезная информация.

Исправления:
- Раскрой: в мастере ярлыка этикеток восстановительных плит и отходов можно было настроить только общую информацию, такую как дата и единица измерения.
- Раскрой: если в базе данных восстановительных плит присутствовала плита материала, не определенная в конфигурации раскроя, TpaCAD выдавал серьезную ошибку.
- Раскрой: на этикетке общее графическое изображение плиты отображалось некорректно, если был включен соответствующий флаг RightToLeft.
- Раскрой: страница справки, вызываемая мастером ярлыка восстановительных плит и отходов, не открывалась.
- Раскрой: на этикетке Раскроя, общая графика листа не отображалась, если был включен флаг переворачивания вложенных элементов (*nesting flip*), даже если управление метками на нижней грани не было включено.

Версия 2.4.26 (20 октября 2025 г.)

Новинки:
- Раскрой: добавлено создание файлов формата SVG для графического представления решения по раскрою
- Раскрой: на вкладке «Индивидуализировать» при изменении значения в поле «Все поля» изменяются значения внешних полей, а не внутренних
- TpaCAD: в конфигурацию TpaCAD добавлен флаг, позволяющий применять инструменты, использованные при импорте, также к файлам TCN, распознанным как созданные внешними программами.

Исправления:
- Раскрой: в некоторых случаях папка для хранения этикеток отходов не создавалась. Выводилась ошибка «Общая ошибка в GDI+».
- Раскрой: расширенная резка отходов была доступна только в том случае, если также была включена вставка этикетки.
- Раскрой: если значение DPI для печати этикеток, заданное в конфигурации Раскроя, отличалось от 300, края могли быть расположены неправильно.
- Раскрой: исправлена ошибка при сортировке деталей по спирали в направлении от центра к периферии.
- TpaCAD: если путь к подпрограмме содержал пробелы, он всегда считался абсолютным. Это могло ограничить переносимость программ из одной пользовательской программы в другую.
- TpaCAD: исправлена ошибка, из-за которой вызов макроса в подпрограмме, который, в свою очередь, вызывал защищенный макрос, приводил к появлению предупреждения «225 одно или несколько заданий были исключены».
- TpaCAD: исправлена ошибка, возникавшая при включенном управлении последовательностью и создании заданий с помощью процесса STOOL: в матрице деталей присваивались одинаковые порядковые номера, что делало задания неразличимыми. Максимальное количество назначаемых уровней последовательности — 5.

Версия 2.4.25 (14 апреля 2025 г.)

Новинки:

- Раскрой: изменения для улучшения размещения деталей с разными диаметрами инструмента на плите
- Обработка Lamello с помощью 3-осевого фрезерного инструмента: добавлен флаг для управления античипированием

Исправления:
- Раскрой: исправлена проблема с сортировочными плитами с включенным приоритетом при наличии восстановительных плит
- Раскрой: исправлена ошибка, из-за которой профили раскроя перекрывались, если флажок **Автоматическое позиционирование ярлыка** был отключен, а флажок **Перекрытие профилей раскроя** был включен.

Версия 2.4.24 (29 января 2025 г.)

Новинки:
- Изменить диапазон значений, которые можно задать в конфигурации параметра **Угловое положение минимума** для функции **Насечка с острыми вершинами**
- Улучшены проверки распознавания дуги, уподобленной окружности, при выполнении коррекции радиуса фрезы

Версия 2.4.23a (25 ноября 2024 г.)

Исправления:
- Обработка Lamello фрезерным инструментом: данные профиля не были переданы во время оптимизации

Версия 2.4.23 (23 октября 2024 г.)

Новинки:
- Раскрой:

1. добавлена возможность автоматического позиционирования ярлыка детали, когда невозможно разместить его в центре детали.
2. добавление обрезков отходов на плите, даже когда из ней создается восстановительная плита
3. дополнительная обработка ярлыков-Штрих-код восстановительных плит на решении раскроя
4. добавлена информация в файл отчета в формате xml о восстановительных плитах, положении ярлыка, типе детали и отходах
5. добавлена опция в Конфигурации раскроя, которая позволяет размещать ярлык на детали, даже если размер ярлыка больше размера детали.
6. добавлена новая усовершенствованная операция резки лома с возможностью размещения ярлыков в местах разреза.
7. добавлен Мастер ярлыка для создания специальных ярлыков для восстановительных плит и областей, образовавшихся в результате резки обрезков.
8. добавлена возможность скрыть отображение переменных V и переменных O
9. добавление обработки Lamello фрезерным инструментом на 3-координатном станке

Исправления:
- Раскрой: были исправлены ситуации с перекрывающимися частями и размещения вне панели.

Версия 2.4.22a (05 июня 2024 г.)

Новинки:
- Операции **ПРЯМОУГОЛЬНИК** и **ВЫЕМКА ПРЯМОУГОЛЬНИКА (ПО ГРЕЧЕСКОМУ ОРНАМЕНТУ)**: обновлена база данных операций и макрос Rect.tmcr с добавлением параметра **Допуск по диаметру**. Этот параметр доступен и управляется только в том случае, если активна технология замены по износу.

Исправления:
- Раскрой: исправлена ошибка при обработке с лишними деталями, которая в некоторых случаях приводила к закрытию ТраCAD.

Версия 2.4.22 (21 мая 2024 г.)

Новинки:

- Добавлена опция **Сохраняет состояние отображения активной грани при выходе из программы**, которая позволяет сохранять состояние отображения активного грани при закрытии TpaCAD
- В панель кнопок в *Области назначения операции* добавлена команда **Редактировать подпрограмму**, позволяющая открыть подпрограмму в другом экземпляре TpaCAD
- В инструменте **Применить точки крепления к профилю** и **STOOL: ПРИМЕНИТЬ ТОЧКИ КРЕПЛЕНИЯ** минимальное количество креплений или разрывов равно 1
- Раскрой: добавлено управление восстановительными плитами
- Раскрой: изменена работа с графикой пользовательских изображений, которые теперь адаптируются к контейнеру и растяжка не выполняется
- Раскрой: дополнительные детали обрабатываются, даже если количество доступных деталей равно 0.

Исправления:
- Раскрой: исправлена ошибка, при которой пользовательские изображения не загружались в ярлык, если путь к файлу был длиннее 100 символов
- Раскрой: в некоторых случаях отходы не добавлялись в конце всех операций
- Раскрой: исправлено наложение деталей и расположение за пределами панели.

Версия 2.4.21 (11 марта 2024 г.)

Новинки:
- Раскрой: при запуске функции раскроя вставляются чеки и предупреждения в глобальную технологию: коррекция на инструмент не должна устанавливаться
- Раскрой: управление столкновением частей было добавлено в окно создания ручных кластеров
- Раскрой: добавлена вставка изображенй на ярлыке из **Мастера ярлыка**, из CSV-файла , из таблицы программ Раскроя
- Раскрой: увеличено количество строк в таблице программы с 300 до 500
- Изменены операции **Lamello**: добавлены технологические параметры (станок, группа, электрический шпиндель, инструмент и тип инструмента) в крепежное отверстие

Исправления:
- Инструмент раскроя и решение раскроя: если в конфигурации TpaCAD была включена опция **Использовать запятую как десятичный разделитель**, полученные операции из процедуры раскроя не были созданы
- Раскрой: функция копирования и вставки числовых значений в таблицу **Деталей** и таблицу **Плит** не работала
- Исправлена проблема, когда в некоторых особых случаях TpaCAD создавал серьезную ошибку, если служба печатной катушки Windows была отключена
- Исправлена проблема в описании программы: если между одной строкой комментария и другой оставалась пустая строка, то при следующем открытии удалялись строки, присутствующие после пустой строки
- Решение различных проблем: коррекция радиуса фрезы, геометрическая библиотека

Версия 2.4.20 (15 ноября 2023 г.)

Новинки:
- Раскрой: добавлена опция **Оптимизировать все в True Shape** в Конфигурации раскроя (дополнительную информацию см. в руководстве TpacadNt), что позволяет оптимизировать области отходов, образующиеся при раскрое ручных кластеров
- Раскрой: опция **Формы: оценить внешние геометрии** также применяется в случае Раскрой True Shape
- Раскрой: в графической окне настройки ручного кластера добавлена команда привязки между прямоугольниками, панелями и частями прямоугольной формы
- Раскрой: в графической окне настройки ручного кластера добавлены две кнопки включения/выключения отображения габарита профилей линейным отрезком и ссылки на размещение детали

Исправления:
- STOOL выемки: в некоторых особых случаях начальная точка траектории выемки рассчитывалась неправильно
- SRECT: когда было установлено значение длины прямоугольника меньше диаметра инструмента выемки, TpaCAD выдавал серьезную ошибку
- Раскрой: при нажатии и удержании клавиши в поле колонны **Имя** списка частей, TpaCAD выдавал серьезную ошибку
- Раскрой: когда единица измерения на ярлыке был в дюйм, ярлык рассчитывалсья неправильно
- Инструменты раскроя: когда в конфигурации TpaCAD на вкладке **Экспорт** для настроенного экспортера были включены опции **Активировать в "Сохранить"**, **Активировать в "Экспорт"**, **Активировать в "Раскрой"** и был активирован инструмент **Решение раскроя** при сохранении решения, TpaCAD выдавал серьезную ошибку

- Инструменты Раскроя: при наличии ручного кластера в списке частей значение поля **Номер решения** в окне решения плит было заполнено неправильно, а кнопка **Возобновить** была включена, хотя все его части были типа прямоугольников или панелей
- Раскрой: при наличии ручного кластера в списке частей неправильное значение поля **Детали** в окне решений плит было отображено
- Раскрой: когда флаг **Отобразить зоны** был включен, цвет заливки части был белым в некоторых случаях, даже если цвет части

Версия 2.4.19a (06 сентября 2023 г.)

Исправления:
- Исправлена ошибка GDI+ при сохранении меток, содержащих QrCode. Проблема появилась в версии 2.4.19.

Версия 2.4.19 (31 июля 2023 г.)

Новинки:
- Добавлено управление универсальным инструментом
- Раскрой: ярлык не вставляется в деталь, если ее размер больше размера детали
- Модификация к обработке Lamello: добавлен параметр **Угол поворота** обработки на плоскости
- Модификация к обработке Lamello с фрезой: добавлены параметры **Длина канавки** и **Угол поворота** обработки на плоскости
- Раскрой: добавлена опция, определяющая, как интерпретируется значение поля **Приоритет** в деталях, кластерах и плитах
- Раскрой: улучшение графических оконных команд создания кластера: добавлена возможность перемещать и вращать деталь прямо с клавиатуры

Исправления:
- Раскрой: параметры Длины [DL] и Высоты [DH] при операции ШТРИХ-КОДА не содержат значений, заданных в Мастере ярлыка
- Раскрой: исправлены случаи, когда ручные кластеры, размещались с перекрытием.
- Раскрой: первая часть списка не удалялась при использовании команды удаления одной строки.
- Раскрой: поле с активным цветом не отображалось в окне выбора цвета или окне просмотра плит.
- ТраCAD: исправлена проблема в процедуре выемки при включенном флаге **Выемка наружу**. Усовершенствование процедуры выемки привело к восстановлению многих областей, которые ранее не опорожнялись, но тот факт, что эти области всегда должны выполняться после первичной выемки, не принимался во внимание.
- При многократной setup замена изношенного инструмента, если он включен, выполняется только при первой setup.
- ТраCAD: исправлен случай серьезной ошибки при закрытии окна Конфигурации ТраCAD. Проблема была обнаружена на компьютерах с графическими чипсетами Intel Graphics UDH 730 или Intel Iris xe и решена путем обновления драйверов чипсетов.
- Напечатать ярлык раскроя и программы: в некоторых случаях ярлык печатался меньше требуемого размера.

Версия 2.4.18 (18 мая 2023 г.)

Новинки:
- Добавлена обработка Lamello фрезерным инструментом
- В метки добавлена возможность вставки информации без дополнительных псевдонимов в QR-код
- Раскрой: добавлено управление ручным кластером

Исправления:
- ТраCAD: исправлена проблема в работе ТраCAD из командной строки, из-за которой даже если флаг был отключен **Запросить подтверждение на выполнение извещения команд**, всегда отображалось окно подтверждения

Версия 2.4.17 (24 февраля 2023 г.)

Новинки:
- ТраCAD: добавлена команда в режиме запуска ТраCAD из командной строки (/W), которая сохраняет файл без показа окна подтверждения сохранения. Новая команда — idFileSaveSilentTo|«полный путь». Пример: +idFileSaveSilentTo|"c:\albatros\product\programBase.tcn"

Версия 2.4.16 (17 февраля 2023 г.)

Новинки:
- Раскрой: введено управление метками на нижней грани, когда включен режим Nesting-Flip
- STOOL Выемка прямоугольника (по греческому орнаменту): добавлен параметр, который назначает значение покрытия в процентах от радиуса
- Меню **Предпочтительные инструменты**: на панель быстрого доступа добавлено меню со списком предпочтительных инструментов

Исправления:
- Редактирование операций: в некоторых случаях была отключена клавиша подтверждения вставки операции
- Процедура выемки: исправлены некоторые предельные расчетные ситуации, при которых выемка не производилась
- Процедура выемки: улучшена процедура управления остаточными зонами
- Функция раскроя: данные центрального положения (XC;YC) на габаритных прямоугольниках размещения были записаны в файл отчета без учета значений пределов, примененных к плите
- TpaLangs закрывалось с ошибкой, если была нажата клавиша ALT без открытых файлов сообщений

Версия 2.4.15 (20 декабря 2022 г.)

Новинки:
- Оптимизация прокрутки операций в текстовой зоны ASCII
- Добавлены сочетания клавиш [CTRL+2] и [CTRL+3] для включения зоны данных операции и текстовой зоны ASCII.
- Операция Ряд Отверстий по X, Ряд Отверстий по Y, Повторить по X и Повторить по Y: обновлены база данных операций и макросы Repeaty.tmcr и Repeatx.tmcr с добавлением параметра для дублирования операции на противоположной стороне
- Добавлено окно описания списка горячих клавиш: вызывается нажатием на иконку  .
- STOOL Выемка прямоугольника: добавлена проверка значения радиуса. Если заданное значение меньше значения радиуса инструмента, появляется сообщение об ошибке.

Исправления:
- Исправлена проблема, из-за которой горячие клавиши [CTRL+W] для вызова команды **Масштабировать Окно**, [CTRL+Shift+W] для команды **Масштабировать Предыдущий** и F6 для команды **Масштабировать Продления** активировались только после вызова команд из контекстного меню.

Версия 2.4.14 (25 октября 2022 г.)

Новинки:
- Функциональность раскроя: добавлено в раскрое *TrueShape* управление поворотом краев этикеток при повороте на 180 или 270 градусов. При прямоугольном раскрое поворот краев составляет 0 или 90 градусов.

Исправления:
- Исправлена ошибка, из-за которой в **Предварительном просмотре для оптимизации** смоделированная деталь отображалась некорректно
- Функциональность раскроя: в таблице определения деталей раскроя столбец для включения зеркального отображения отображался только в том случае, если были включены все типы файлов
- Функциональность раскроя: изменен шрифт с «Verdana» на «Microsoft Sans Serif» при печати отчета о Раскрое, чтобы лучше определить символ
- Функциональность раскроя: устранена проблема поворота на 90 градусов в случае Раскроя True Shape
- Присвоение **Свойства С или Комментария** нескольким обработкам: если операции, присутствующие в списке последовательностей, были прокомментированы, то при снятии флага комментария они больше не присутствовали в списке последовательностей
- Раскрой: в случае Раскроя True Shape, если детали были повернуты, размеры длины и высоты отображались заменены местами на этикетках (label)

Версия 2.4.13 (15 июля 2022 г.)

Новинки:
- В таблицу инструментов добавлен фильтр визуализации по типу инструмента

Исправления:
- Устранена проблема, при которой флаги обработки, заданные во вставке или редактировании, были неправильно оценены
- Палитра операций: в некоторых случаях обработка, которая не применялась к активной грани, все еще отображалась
- Функциональность раскроя: нумерация деталей в графической области увеличена до 9999
- Функциональность раскроя: если управление метками было отключено, назначение вращения не менялось
- Функциональность раскроя: в отчете о раскрое, даже в случае форм, общая занятая область отображалась на основе размера панели. Таким образом занимаемая площадь была больше площади листа
- Чтение файла ISO: поля T, F, S и исправление, которые записаны в файле ISO, никогда не читались, даже если они не были назначены на внешней или внутренней setup.
- Чтение файла ISO: если применялось преобразование единиц измерения, оно не применялось к координатам центров и радиусам дуг
- Исправлены различные проблемы: опустошение областей, коррекция радиуса фрезы, геометрическая библиотека, инструменты для прикрепления к профилю.
- Модуль экспорта: в дюймовой программе значение поля скорости F экспортировалось как целочисленное значение. Теперь он сохраняет 3 десятичных знака.
- Макрос ISO.TMCR: исправлена проблема назначения значений коррекции и радиуса инструмента
- Операции: корректировка обработок 5-осевой кривой, Подпрограммы ISO и Кривой

Версия 2.4.12 (10 мая 2022 г.)

Новинки:
- Все файлы прототипов сохраняются только в собственном формате TpaCAD, а не в формате ASCII.
- Улучшена интерактивная процедура захвата 2D-представления при наличии индуцированных вызовов.
- Табуляция граней: добавлена индикация индуцированных линий
- Табуляция граней: для всех присутствующих фиктивных граней отображается изображение с индикацией фиктивной грани
- Функциональность раскроя: в конфигурацию Раскроя добавлена опция **Формы: оценить геометрию раскроя**. Этот выбор влияет на детали, вставленные в проект раскроя, как Профильные детали. Оценивает пространство за пределами прямоугольника посадочного места *Геометрии раскроя* для назначения окружающей области уважения, чтобы защитить места, прилегающие к *Геометрии раскроя*
- Функциональность раскроя: добавлено управление операциями Nesting-flip
- Функциональность раскроя: в прототипе файла было добавлено управление головной и концевой работой с использованием специфического формализма. Считываются операции на грани 1 и 2 (если активирована функция Nesting-flip), описания также сохраняются в файлах * .TCN решения раскроя.
- Функциональность раскроя: в конфигурацию раскроя добавлена опция **Минимальная площадь**, которая применяет прикрепления только к небольшим деталям. Таким образом, мелкие детали исключаются из управления оптимизированным разрезом.
- Модуль импорта из формата ISO: значения, назначенные при setup, такие как поле T и поле S, также распространяются на профиль
- Модуль импорта из формата ISO: при наличии большего количества G0 значения поворотных осей распространяются, если они не назначены последующим инструкциям G0
- Приложение TpaLangs: длина идентификаторов текста сообщений увеличена до 250 символов

Исправления:
- Инструменты раскроя: решены ситуации ошибок размещения в случае ручного кластера островов.
- Функциональность раскроя: решены ситуации, в которых подачи по глубине применялись как к предварительному, так и к вырезу.
- Функциональность раскроя: программы завершения не записывались, если верхняя грань не подвергалась механической обработке
- Модуль экспорта в формате ISO: не сохранялись поворотные оси при экспорте ориентированного профиля с разнесением входного и выходного участков
- Приложение TpaLangs: кнопка «Сохранить» не активировалась, когда менялись только идентификаторы сообщений.

Версия 2.4.11 (28 марта 2022 г.)

Исправления:
- Исправлена ошибка при открытии программы, если бит функции раскроя не был включен на аппаратном ключе

Версия 2.4.10 (21 марта 2022 г.)

Новинки:
- Изменения в функционировании кода SSIDE
- Изменения в функциональности *Раскроя* (добавлена возможность применения экспорта формата к результатам раскроя; исключено внутреннее использование поля K)
- Изменения в применении *Исключений* (применить в оптимизации, экспорте, предварительном просмотре оптимизации)
- Дополнение в модуле экспорта формата DXF (добавлен выбор экспорта 2D)

Исправления:
- Исправлены ошибки при обработке вложенных вызовов (использование переменных j)

Версия 2.4.9 (11 января 2022 г.)

Новинки:
- Редактирование именованной программы при сохранении (зарезервированные имена Windows отфильтрованы. Например: "con", "null",..)
- Изменения в функциональности *Раскроя* (угловой шаг раскроя *True Shape* изменил до 5 ° от 10 °; изменения в управлении ярлыками)
- Изменения в импорте формата *DXF* (импорт объектов с распознаванием уровня: добавлена возможность подключения профилей)

Исправления:
- Исправлена проблема в функциональности изменения среды Станка / Чертежа (для выбора команды меню требовался ключ **Профессиональный**)
- Исправлены ошибки при импорте из формата *DXF* (уменьшение *сплайновых* кривых)
- Устранены мелкие проблемы
- Исправлены ошибочные ситуации в *Раскрое* (сортировка профилей предварительной резки фигур, создание профилей резки обрезков, создание этикеток с примененным поворотом)

Версия 2.4.8 (29 июля 2021 г.)

Новинки:
- Изменения в процедуре создании этикеток, программе и раскрое (улучшенное разрешение файлов изображений)

Исправления:
- Устранена проблема в операции *SZSSHAPE: Снижение формы профиля* (с первой внешней назначенной технологией, не регулировал развитие сокращения)
- Устранены мелкие проблемы

Версия 2.4.7 (06 июля 2021 г.)

Новинки:
- Изменения в функциональности *Раскроя* (создание профилей предварительной резки для *форм*)
- Изменен критерий оценки запрограммированной технологии (см. параграф: *Операции-> Типы операции ->Технология*)
- Незначительные изменения в конфигурации приложения

Исправления:
- Исправлена ошибка в применении привязки на сложных операциях
- Исправлена ошибка в *Раскрое* (решения размещений *True Shape*, создание профилей резки отходов, удаление временных файлов)
- Исправлена ошибка на импорте формата *ISO* (удаление строк вставленных в профиль)
- Исправлены ошибки в приложении *TpaLANGS* (обновление статуса меню, особые случаи при назначении эталонного языка)
- Устранены мелкие проблемы

Версия 2.4.6 (14 мая 2021 г.)

Новинки:
- Уменьшено максимальное количество копий TpaCAD (до 4)
- Общие инструменты грани: исключена возможность использовать данные в *буфере обмена*

Исправления:
- Исправлена ошибка при использовании инструмента *Изменить сечение профиля* (случай: изменение касательной в конце дуги)
- Исправлена ошибка при использовании процессов разработки текста (*Угол наклона* применялся только при запросе на размещение текста на геометрическом элементе)
- Исправлены ошибки в *Раскрое* (при применении крепления к профилям формы: опция компенсации инструмента применялась дважды; использование *форм*, полученных из файлов DXF: с объектами сплайн-типа, острова могли быть устранены)
- Исправлены ошибки в решении параметрического программирования (функции, которые работают со строками, используя переменную r, указанную с именем)

Версия 2.4.4 (11 февраля 2021 г.)

Исправления:
- Исправлены ошибки в *Раскрое* (создание профилей резки: не использовалась технология предварительной резки, в случае оптимизированного профиля; в профилях резки с зигзагообразным управлением входом / выходом, размер Z был записан неверно)
- Исправлены ошибки в графическом представлении поверхностей

Версия 2.4.3 (25 января 2021 г.)

Новинки:
- Добавления в функциональности *Раскроя* (поле QR-код на этикетке; поля добавлены в присвоение деталей)
- Изменение в функциональности *Напечатать ярлык программы* (возможность назначить поле QR-код)
- Добавление в модуле экспорта в формат ISO (добавлена настройка для преобразования в стандартную систему XY)

Исправления:
- Исправлена ошибка в модуле экспорта в формат *DXF* (случаи дуга со скругленным углом >180°)

Версия 2.4.2 (02 декабря 2020 г.)

Новинки:
- Изменения/ Различные добавления

Исправления:
- Устранены мелкие проблемы

Версия 2.4.1 (15 сентября 2020 г.)

Новинки:
- Изменения/ Добавления в функциональности *Раскроя* (добавлены опции: *Применить предварительную резку ко всем деталям*, *Предвидеть нарезку маленьких деталей*; добавлено управление оптимизированными профилями предварительной резки)
- Добавления в функциональности *Раскроя* (края деталей)
- Изменениия в продвинутом инструменте *Создать шрифт, исходя из геометрии* (графическое выделение при выборе профилей)

Исправления:
- Исправлена ошибка в использовании *Клиентских функций*
- Исправлена ошибка в тмпорте из формата *PZA* (случай прямого исполнения файлов PZA с чтением устаревшей версии файла)
- Исправлены ошибки в графическом представлении (решения для обрезки детали и / или грани в случаях: криволинейных поверхностей или граней, поворотного лезвия)

Версия 2.4.0 (22 мая 2020 г.)

Новинки:
- Изменение в конфигурации TpaCAD (устранение настройки *Опытный пользователь,* считается всегда активной)
- Изменение в конфигурации TpaCAD (устранение настройки *Оптимизировать графическое представление*, считается всегда активной,)
- Добавление в чтение и импорт файлов в формате ISO (задание вращающихся осей, задание подготовок)
- Изменение в команде *Напечатать ярлык программы* (возможность сохранеия файла была добавлена)
- Добавлена команда в функциональности *Раскроя* (Сохранить отчет (.XML))
- Добавление в параметрическое программирование (функция: *geo[plface; ..]*
- Добавление в импорт из формата *DXF* (*Преобразовать сегментированную полилинию в круг)*

Исправления:
- Исправлены ошибки в графическом представлении (случай Заблокированного вращения детали)
- Исправлены ошибки в решении трехмерной дуги (прямое задание направления вращения)

2.2 Версии 2.3.1 (08 марта 2019 г.) - 2.3.20 (02 декабря 2020 г.)

Версия 2.3.20 (02 декабря 2020 г.)

Исправления:
- Соответствует версии 2.4.1

Версия 2.3.14 (31 марта 2020 г.)

Новинки:
- Добавления в параметрическое программирование (функции:*prrot*)

Исправления:
- Исправлены ошибки в отчете о *Раскрое* (сообщается о неправильном размере листа)
- Решена проблема при операции LAMELLO (при нанесении на боковую поверхность соответствие инструмента не было полностью проверено)
- Исправлена ситуация в Назначении последовательностей (особые случаи отображения точки применения для сложных кодов)

Версия 2.3.13 (10 марта 2020 г.)

Исправления:
- Исправлены ошибки в решении *Раскроя* (случая расширенных операций профиля)

Версия 2.3.12 (27 февраля 2020 г.)

Новинки:
- Обновление сертифицированной подписи для исполняемых файлов с алгоритмом классификации *sha256*
- Изменения в импорте из формата *PZA* (выделение опорной стороны панели, разделение на двух программах при операциях на нижней грани)
- Изменения в управлении импортом форматов (случая импорта с созданием нескольких файлов)

Исправления:
- Исправлены ошибки в решении направления поворота трехмерной дуги
- Исправлена проблема в конфигурации операции *STZREPATT*
- Устранены мелкие проблемы

Версия 2.3.11 (30 января 2020 г.)

Исправления:
- Исправлены ошибки в стандартном окне представления технологической Таблицы (разные при управлении оснащенных параметров)
- Исправлены ошибки при импорте из формата *PZA* (назначение в окне конфигурации; различные в выборе инструмента типа фрезы)

Версия 2.3.10 (14 января 2020 г.)

Исправления:
- Исправлены ошибки при задании *местной Технологии* (выбор рабочей грани инструмента на всех боковых гранях: не работал)
- Исправлены ошибки в графическом представлении (сетка смогла применить неправильные задания)

Версия 2.3.9 (02 декабря 2019 г.)

Новинки:
- Изменения/ Дополнения в функциональность *Раскроя* (чтение файла спецификации "*.csv": добавленные поля, управление динамичеким *заголовком*)
- Изменения/ Дополнения в решении *Раскроя* (Изменения в принятии незамкнутых профилей раскроя)
- Приложение TpaLANGS: добавлено окно отчета
- Изменения в импорте из формата *PZA* (добавлено управление *Вращение шпинделя*, изменения в операции паза на краю панели)
- Добавлена операция в группу "Запрограммированные Инструменты" (SWEEP)
- Изменения/ Добавления в операции в группе "Запрограммированные Инструменты"

Исправления
- Исправлены ошибки в решении параметрического программирования (функции "*geo[alfa], geo[beta]*" не работали в случае изогнутых граней или поверхностей)
- Исправлены ситуации в решении раскроя (создание оптимизированного профиля резки: случая сечений, которые были устранены)
- Исправлены ситуации в функциональность *Раскроя* (чтение в форме из файла "*.dxf": особые ситуации импорта, задания размеров)
- Исправлены ошибки в графическом представлении (случая: преувеличенный зум с обновлением графики, ошибка в повороте рисунка)

Версия 2.3.8 (29 октября 2019 г.)

Новинки:
- Добавлены команды в функциональность *Раскроя* (Список последних открытых проектов, Добавлены команды просматривая решения)
- Изменения / дополнения в решении *Раскроя* (технологии резки: скорость, зигзагообразное вхождение)
- Добавление в функциональность приложения *TpaLangs* (нечисловые идентификаторы сообщений)
- Изменения в экспорте в формат *ISO* (добавлена страница настроек, обновление технологии, опции перевода профиля)
- Изменения в импорте из формата *PZA* (обновление от заводской технологии, интерпретация сущностей POLICURVA с уже примененной компенсацией инструмента)

Исправления:
- Разрешенные ошибки в приложении *TpaLangs* (отфильтрованные случаи языков)

Версия 2.3.7 (23 сентября 2019 г.)

Исправления:
- Исправлены ошибки в графическом представлении (случаи: моделирование с наполнением)
- Исправлена ошибка в окне *Сохранить файл* (случай сохранения на диске)
- Исправлены ошибки в основных операциях (общая вставка, Cabineo)

Версия 2.3.6 (1 августа 2019 г.)

Новинки:
- Добавлены команды в функциональность *Раскроя* (Сохранить неиспользованные детали, Сохранить результаты по завершении)
- Добавление в *Индивидуализацию* TpaCAD (таблица цветов, используемых в функциональности *Раскроя*)
- Добавления в *Индивидуализацию Раскроя* (*Покрытие между профилями* для решения True Shape, различные в *Мастере ярлыка*)
- Добавление в функциональность *Раскрой True Shape* (размещения по сетке)
- Добавлена функциональность *Напечатать ярлык программы*
- Добавлены операции в группе *Вставки* (Lamello, Cabineo)

Исправления:
- Устранены ситуации ошибки при выполнении *общих программных трансформаций* (Повернуть деталь, Зеркальное отражение детали, Перевернуть деталь)
- Устранены ситуации ошибки в решении раскроя

Версия 2.3.5 (27 мая 2019 г.)

Новинки:
- Добавления в функциональность *Раскрой True Shape* (зерно деталей)
- Изменения в сложных операциях профильных решений (добавлен параметр *Технологический приоритет*)
- Добавлена функциональность в операции *Повторить по X*, *Повторить по Y* (распределение отверстий по бокам)

Исправления:
- Устранена ситуация ошибки в обмен на *единицу* измерения программы (больше не работала)
- Устранены ситуации в решении раскроя
- Устранены ситуации ошибки в графическом представлении (случаи: дизайн добавленных элементов в *небольших* разделах)
- Устранена ситуация ошибки *при экспорте в формат DXF* (экспорт не удался, если расширение по умолчанию для преобразованных файлов не было назначено)

Версия 2.3.4 (9 апреля 2019 г.)

Новинки:
- Добавления во *Внутренней Технологии* (вращение шпинделя)
- Добавлена функция *Компенсировать диаметр в коррекции конического инструмента*

Исправления:
- Устранены ситуации в решении раскроя

Версия 2.3.1 (08 марта 2019 г.)

Новинки:
- Добавлена функция *Раскрой True Shape (Фигурный Раскрой)*
- Добавления в проект *Раскроя программ* (типологии деталей, поля и предварительный просмотр в списке деталей)
- Изменения на Конфигурации *Раскроя программ*
- Пересмотр инструментов *Раскроя профилей*
- Добавлено управление *Зерна панели*
- Добавление в параметрическое программирование (переменный аргумент: *cnf;* функции: *rempty*, *pngru*, *pnstool*, *pntool*)
- Добавлено графическое представление фигурного инструмента
- Добавлено управление технологической заменой из-за износа инструментов фрезерования
- Добавлены операции в группу "Специальные фрезы" (*Выемка прямоугольника по меандру*, *Дверь*)
- Изменения в операциях *Дверь* (возможность поворота разработки вдоль горизонтальной оси)
- Добавление в модуль импорта DXF (случай: полилиния, подобная кругу)

2.3 Версии 2.2.0 (04 декабря 2017 г.) - 2.2.15 (28 марта 2019 г.)

Версия 2.2.15 (28 марта 2019 г.)

Новинки:
- Добавлены руководства и контекстная справка на языке

Исправления:
- Устранены ошибки в графическом представлении (цвет линейки, курсор на поверхность)

Версия 2.2.14 (27 февраля 2019 г.)

Новинки:
- Добавлены руководства и контекстная справка на языке

Исправления:
- Устранены ошибки в графическом представлении (случай: использование низкоэффективных графических плат)
- Устранена ошибка в окне *Сохранить файл* (начальное имя файла не предлагалось)
- Устранена ошибка в модификации *Глобальных Технологий* (изменение активации не может быть приобретено)

Версия 2.2.13 (9 января 2019 г.)

Исправления:
- Устранена ошибка, возникшая в результате обновления ОС до версии 1809 (сообщения об ошибках: "*Visual styles-related operation resulted in an error because visual types are currently disabled in the client area*")
- Устранена ошибка в запросе *Общего инструмента* для больших программ (случаи: замедление запуска команды)
- Устранена ошибка в *импорте* формата (технологическое применение для операции подготовки и / или точечной обработки)
- Устранены ошибки при импорте из формата PZA (случаи: канавка интерпретируется как фрезерование)

Версия 2.2.12 (06 декабря 2018 г.)

Новинки:
- Добавлены сообщения на турецком языке
- Приложение TpaLANGS: добавлены руководства на языке (французский, испанский, русский)

Исправления:
- Устранена ошибка при выделении строк в тексте ASCII
- Устранена ошибка в графическом отображении операций, примененных к поверхности

Версия 2.2.11 (16 ноября 2018 г.)

Исправления:
- Исправлены ошибки в функциональности *Раскрой панелей* (обнуление исполнительных последовательностей)
- Устранены ошибки при импорте из формата PZA (случаи: канавка интерпретируется как фрезерование)
- Исправлена ошибка в окне *Сохранить Файль* (фильтр недопустимых символов)
- Устранена ошибка ситуации в применении *Инструмент Зеркального отражения* к обработкам группы Специальные фрезы (прямоугольник, многоугольник, ..)

Версия 2.2.10 (31 октября 2018 г.)

Исправления:
- Устранены проблемы в функциональности *Раскроя панелей* (задание переменных r, применение края внутри размещения, отчет о пути длиной> 100 символов, удаление операций по геометрическим размещениям)
- Устранены проблемы при импорте из формата PZA (случаи: паз прорези с профилем, калибровка отрезков на входе/выходе профиля при выполнении формирования детали)
- Улучшенная функциональность графической печати (размер изображения сводится к полезной области представления)

Версия 2.2.9 (28 сентября 2018 г.)

Исправления:
- Устранена проблема при печати ярлыков в функциональности *Раскроя панелей* (случай: печать с растянутым изображением)

Версия 2.2.8 (19 сентября 2018 г.)

Исправления:
- Устранена проблема в окне справки параметрического программирования (некоторые записи в списке «pr \ ...» были неверными)
- Устранена ситуация неправильного графического представления (случаи: операций на закрытых поверхностях)
- Устранена проблема при импорте формата DXF (случай: неправильный импорт окружности как точки)

Версия 2.2.7 (27 июня 2018 г.)

Новинки:
- Добавлено управление линейкой в графическом представлении с помощью Каркасного представления
- Расширение работы кода SSIDE (разработка также по типу подпрограмме)
- Добавление в параметрическом программировании (форматирование в переменной типа строки)
- Добавлено управление заданием с символами, как правило, Unicode в функции *Раскроя панелей* (список материалов, задание в проект раскроя и е в ярлыках)

Исправления:
- Устранена проблема, которая привела к отключению клиентских разделов (введена с версией 2.2.3)
- Устранена проблема в решении *Раскроя панелей* (случай: создание равных панелей)
- Устранена проблема при чтении файлов клиентских сообщений имеющих разделенный формат (один файл для каждого языка)

Версия 2.2.3 (22 марта 2018 г.)

Новинки:
- Добавлено управление линейкой в двухмерном графическом представлении
- Добавление в меню Чертеж (Прямоугольник с центром и ребром, Наклонный прямоугольник, Звезда)
- Добавление операции в группу "Многоугольники" (звезда)
- Добавление операции в группу "Специальные фрезы" (звезда)
- Добавлен модуль экспорта в формат DXF (конфигурируемый модуль: *TpaSpa.DxfCad.v2.dll*)

Исправления:
- Устранена проблема графического отображения вертикальных/горизонтальных габаритов (медленное отображение в результате неправильного критерия фрагментации и линеаризации закругленных элементов)
- Устранена проблема графического отображения вертикальных габаритов (случаи отображения ложных триангуляций)
- Устранена проблема при вызове модулей экспорта (после экспорта было невозможно открыть конфигурацию модуля)

Версия 2.2.2 (30 января 2018 г.)

Новинки:
- Добавлены команды *Общих трансформаций детали* (меню *Применить*: Повернуть, Зеркальное отображение, Перевернуть)
- Добавлены команды *Общих инструментов* (меню *Инструменты*: Центрировать и Выровнять с гранью)
- Добавление в *Раскрой программ* (возможность исключения операций вне размещений)
- Добавления в параметрическое программирование (функции: "sign[nn]", "geo[angm;..]")
- Добавление операции в группу "Специальные фрезы" (тригонометрическая функция)
- Добавление в операции в группе "Специальные фрезы" (задание внешней технологии)
- Изменения в применении настройки индивидуализации *Всегда показывать начальную точку на отрезках профиля*

Исправления:
- Устранена проблема импорта формата DXF (задание размеров детали из одного эллипса)
- Устранена проблема импорта формата ISO (задание размеров детали)
- Устранена проблема Задания последовательностей (выбор графического представления не был активным при отображении операций, которые не могут выстраиваться в последовательность)
- Устранена проблема графического отображения вертикальных габаритов (случай накладки на грань с инструментом, ориентированным параллельно грани)
- Устранены проблемы графического поворота детали
- Устранены проблемы в приложении TpaLangs.exe (случаи распознавания системного языка)

Версия 2.2.1 (19 декабря 2017 г.)

Исправления:
- Устранена проблема импорта формата ISO (импорт не выполнялся)

Версия 2.2.0 (4 декабря 2017 г.)

Новинки:
- Реорганизация распределения локализованных файлов (сообщения приложений, контекстуальная и документальная справка)
- Реорганизация сообщений базовой базы данных операций в файле DLL
- Добавление функции в *Окно сигналов* (отображает коды комплекта, указанные на ключе)
- Добавлена команда *Новая по модели* (создание программы по модели)
- Добавлено меню команд в пунктах *Последние открытые программы*
- Добавление в *Раскрой программ* (возможность не генерировать профили резки, новые поля задания ярлыков)
- Добавление в конфигурацию: уровень изменения настроек в (Уставки детали, Клиентские разделы)
- Добавления в функцию *Essential* (опытный пользователь, инструменты построения профиля)
- Добавление функции *Моделирование* из профилей, запрограммированных на грани
- Добавление в Персонализацию (Активации графических представлений: *Отключить графические элементы, добавленные для большой программы*)
- Добавление в меню "Отобразить" (меню, добавленное команде *Габарит в трехмерном изображении*)
- Добавление в *Общие инструменты* (графическое взаимодействие воспроизводит также операции, задействованные трансформацией)
- Добавление в общий инструмент *Повернуть* (вращение на минимальный габарит)
- Добавление разделов в конфигурацию импорта из DXF (*Подпрограммы и уровни, Подпрограммы и блоки*)
- Добавление операций в группу "Специальные фрезы" (различные типы створок)
- Добавления в операции применения *Прямоугольник* (опция решения скругления или фаски)
- Добавлено приложение TpaLangs.exe для управления файлами клиентских сообщений

Исправления:
- Устранены проблемы в продвинутом инструменте *Раскрой профилей* (распознавание профиля внутри группирования маршрутов)

2.4 Версии 2.0.0 (08 апреля 2016 г.) - 2.1.18 (26 сентября 2018 г.)

Версия 2.1.18 (26 сентября 2018 г.)

Новинки:
- Изменения ненормальных ситуаций в результате обновлений ОС Windows 10
- Добавлен специяльный импорт из формата PZA (переворачивание детали; различные дополнения)

Исправления:
- Исправлены ошибки в функции *Раскроя программ* (создание профилей резки с несколькими проходами; случаи ошибок, возникающих в результате взрыва сложных кодов)
- Исправлены ошибки при импорте из формата DXF (импорт части эллипса)
- Устранены мелкие проблемы

Версия 2.1.11 (29 ноября 2017 г.)

Новинки:
- Изменения порядка задания типов программы приложению TpaCAD (случай обновленных версий ОС Windows 10)

Исправления:
- Устранены проблемы неправильной навигации в таблице технологии (случаи линии и/или станка с несколькими группами)
- Устранена проблема импорта формата DXF (открытие файла требовало эксклюзивного доступа)

Версия 2.1.10 (27 октября 2017 г.)

Новинки:
- Определенные добавления импорта из формата DXF (поворот детали с обменом осей xy, добавление значимых префиксов "&..")
- Определенные добавления импорта из формата ISO (настройки: *Код G размеров*, *Квадрант станка*)

Исправления:
- Устранены проблемы импорта из формата DXF (задание координаты Z для полилинии, связанной с Макропрограммами и уровнями)

Версия 2.1.9 (13 октября 2017 г.)

Новинки:
- Определенное добавление импорта из формата DXF (общая настройка: *Импортировать пустую панель*)
- Определенное добавление импорта из формата ISO (линеаризация малых *дуг*)

Исправления:
- Устранены проблемы применения общих инструментов *Преобразование единиц измерения детали*
- Устранены проблемы применения активизации графического отображения *Адаптировать габарит профилей к длине отрезков*
- Устранена проблема управления чертежом *Эллипс по 3 точкам* (наблюдалась ошибка применения указателя на осях X/Y)
- Устранены проблемы решения *Раскроя программ* (дублирование листов, решение нескольких групп решения, технологические задания профилей резки)
- Устранены проблемы в функции *Essential* (использование *Глобальных технологий*)
- Устранена проблема распознавания *формата файла ISO* (случай: файл "TCN" в формате Unicode)

Версия 2.1.8 (18 сентября 2017 г.)

Новинки:
- Определенное добавление импорта из DXF (общая настройка: *Не перепозиционирует элементы*)

Исправления:
- Устранены проблемы задания листов решения *Раскрой программ* (случаи выбора оптимального решения, случай *серьезного предупреждения,* которое больше не является блокирующим; оценки, связанные с габаритом примененных операций)
- Устранена проблема в операции STOOL: STMULTI (начальные настройки менялись в решении следующих профилей)
- Устранены проблемы в продвинутом инструменте *Раскрой профилей* (случаи использования в грани-детали)
- Устранена проблема разрешения параметрического программирования (случай смешанного использования символов десятичного разделителя)

Версия 2.1.7 (7 июля 2017 г.)

Новинки:
- Добавление операции в группу "Подготовка" (ПОДГОТОВКА ISO 5x)
- Добавление операции в группу "Подпрограммы" (ПОДПРОГРАММА ISO)
- Добавление операции в группу "Инструменты" (Подачи по Z + Точки крепления)
- Добавления в операции применения паза (задание положения *центра*)

Исправления:
- Устранены проблемы задания листов решения *Раскрой программ* (случай ошибок, обуславливаемых *вращением* сложных операций и *Поправка на инструмент*)
- Устранена проблема в общем инструменте *Повторения с круговой разверткой* (всегда применялось вращение повторений)
- Устранена проблема в инструменте профиля *Разрезать отдельные элементы профиля* (случаи, когда они не выполнялись)
- Устранена проблема в инструменте построения *Дублировать профиль* (случай исключения при запросе дублирования операций)

Версия 2.1.6 (23 июня 2017 г.)

Исправления:
- Устранены проблемы невосстановления компоновки программного приложения
- Устранена неправильная ассоциация типа инструмента в *Таблице технологии* (случаи: типы прорезей)

Версия 2.1.5 (15 июня 2017 г.)

Новинки:
- Добавления в управление грань-деталь (блок, импорт)
- Добавления в функцию *Раскрой программ* (управление ярлыками)

Исправления:
- Устранены проблемы интерактивного сопоставления с операциями (грань-деталь)
- Устранены проблемы задания компоновки ярлыков в функции *Раскрой программ* (подключение кнопок,..)
- Устранена проблема импорта из формата PZA (развертка множественных проходов фрезерования и/или прорези; максимальное значение шпинделя или инструмента изменено с 1000 на 10000)

Версия 2.1.3 (27 апреля 2017 г.)

Исправления:
- Устранены проблемы при открытии вспомогательных окон (открытие на вспомогательном экране)
- Устранены проблемы при открытии окна *Файл Открыть* (сбой решения с маршрутом, заданным строкой длиннее 180 символов)
- Устранена проблема в операции *STOOL: Подача по Z* (случаи задания числа шагов и конечной координаты Z: возможно генерирование ложной ошибки)
- Устранены проблемы в окне конфигурации импортера формата DXF (задания в окне)

Версия 2.1.2 (4 апреля 2017 г.)

Исправления:
- Устранена проблема в функции *Раскрой* (реализация прототипа)

Версия 2.1.1 (20 марта 2017 г.)

Новинки:
- Добавления в параметрическое программирование (функции: "geo[beta;..]", "geo[alfa;..]")
- Добавлены инструменты *Раскроя профилей*
- Добавление в задание ориентируемых геометрий (интерполяция 4 или 5 осей)
- Добавления в общий инструмент *Повернуть*
- Добавление в меню "Графическое представление" (команда *Правка*)
- Добавление в инструмент "Применить точки крепления к профилю" (опция точек крепления/прерываний)

- Обновления руководств и перевод сообщений

Исправления:
- Устранена проблема при открытии окна сохранения программы (фильтры локальных дисков, сетевые папки, облако)
- Устранена проблема в общем инструменте *Повернуть* (случаи поворота, не соответствующие графическому взаимодействию)
- Устранена проблема в общем инструменте *Зеркальное отображение по вертикальной оси* (случаи дуги с заданной координатой Y центра)
- Устранена проблема в инструменте *Отсоединить каждый отрезок профиля* (ошибка в порядке ввода и графическом представлении)
- Устранена проблема в инструменте *Сместить подготовку на замкнутый профиль* (профиль, заданный с: подготовкой + дугой A01)
- Устранена проблема в *Поправке на инструмент* (случаи изменения стороны правки применительно к кругу)
- Устранена проблема в продвинутом инструменте *Создать поверхность, исходя из геометрии* (случай круга)
- Устранены проблемы в функции *Раскрой* (ступенчатая реализация)
- Устранены проблемы графики (отображение зазоров в трехмерном отображении)

Версия 2.1.0 (2 декабря 2016 г.)

Новинки:
- Добавлена функция Раскрой программ
- Добавлен общий инструмент *Повторения на профиле*
- Добавлено управление Глобальными технологиями (настройка в *Индивидуализировать TpaCAD*)
- Добавлено управление в Импорт внешнего формата (настройка в *Индивидуализировать TpaCAD: Общие подключения программы,Подтвердить профили*)
- Добавление в функцию чертежа (длина отрезка для Линии, введенной на непрерывной касательной)
- Добавление в Персонализацию (Активации графических представлений: Адаптировать габарит профилей к длине отрезков)
- Добавление в задание ориентируемых геометрий (начальный и конечный отрезки: возможность исключить *Преследование касательной*)
- Добавление в задание ориентируемых геометрий (интерполяция 4 или 5 осей)
- Добавление опции в конфигурацию импорта из DXF (возможность фрагментирования полилинии на отдельные отрезки)
- Определенное добавление импорта из DXF (возможность ассоциации любого маршрута с функцией *Операции и уровни, Макропрограммы и уровни*)
- Добавление раздела в конфигурацию импорта из DXF (*Логические операции и блоки*)
- Добавление в режим ДЕМОНСТРАЦИОННЫЙ (возможность изменения частей конфигурации, доступных на уровне пользователя)

Исправления:
- Устранена проблема замедлений в графике программы
- Устранена проблема выполнения интерактивного режима *Привязка на грани* (прерывания TpaCAD)
- Устранена проблема управления чертежом *Дуга с радиусом* (не удавался ввод радиуса непосредственно в меню)
- Устранена проблема в программируемом инструменте STZLINE (каждый отрезок профиля был задан как отдельный профиль)

Версия 1.4.14 (6 октября 2016 г.)

Новинки:
- Обновления руководств и перевод сообщений

Исправления:
- Устранена проблема в инструменте *Инверсия профиля* (профиль, заданный с единственным отрезком, мог утратить изменение глубины)
- Устранена проблема, связанная с управлением ситуациями предупреждения в программировании поля в клиентском разделе
- Устранена проблема в инструменте *Применить точки крепления к профилю* (в случае профиля с маленькими первыми отрезками могло не применяться никакого крепления)
- Устранены проблемы в командах группы **Найти/ Заменить** (не выполнялась замена в программах с большим числом строк)
- Устранена проблема в выемке зон (очень фрагментированные профили и генерирование ложных замкнутых зон)

- Устранена проблема, связанная с функционированием с некоторыми языками (напр., турецким) (не удавался запуск Клиентского оптимизатора)
- Устранена проблема в выполнении импорта формата (импорт символов не ANSI)

Версия 2.0.0 (08 апреля 2016 г.)

Новинки:
- Добавление в конфигурацию: подключение "Поверхности" (Уставки детали, Геометрия детали)
- Добавление в конфигурацию: "Применить трансформации к ориентированной геометрии" (Уставки детали, Общее)
- Добавление в конфигурацию: "Насечка с острыми ребрами" (Уставки детали, Общее)
- Добавление в конфигурацию: "Технологический компонент установки" (Среда, Компоненты)
- Добавление в конфигурацию: "Преобразовать скорости в [м/мин.] или [inch/sec]" (Открыть и сохранить, Задать в матрице детали и Экспорт)
- Добавление функции: "Технология оснащения"
- Добавление в функцию "Глобальные переменные" (команды меню)
- Добавление в меню Файл (Оптимизировать архив программ)
- Добавление в меню Чертеж (Дуга по радиусу, Круг по 2 точкам, Круг по 3 точкам, Эллипс по 3 точкам, Винт, Спираль)
- Добавление в инструмент построения "Применить точки крепления к профилю" (расстояние между креплениями)
- Добавление в инструмент "Сокращение профиля" (линейная минимизация и прочее)
- Добавлен инструмент профиля "Отсоединить каждый отрезок профиля"
- Добавлен инструмент построения "Разделить пересекающиеся вектора"
- Изменено локальное меню графического контроллера (группа Выбрать, команды: Вырезать, Копировать, Вклеить)
- Добавление в палитру команд ввода данных операции (пункт: "Применить к операциям грани (если автоматическая)")
- Добавление операции в группу "Отдельные дуги" (A27: (tgin, R, A, CW))
- Добавление операций в группу "Многоугольники" (A48: Винт; A49: Спираль)
- Добавление операций в группу "Специальные фрезы" (HELIC: Винт; TWIST: Спираль)
- Добавление операций в группу "Специальные фрезы": фрезерные обработки постепенного сокращения прямоугольника, многоугольника, овала, эллипса, круга, паза, общего профиля
- Добавление операций в группу "Инструменты": фрезерные обработки постепенного профильного сокращения общего маршрута
- Добавление операций в группу "Инструменты": Подтвердить профиль
- Добавления в параметрическое программирование (функции: "geo[sub;..]", "geo[param;..]", "geo[lparam;..]")
- Добавление функции Печать в приложение TpaWorks.
- Добавление опции в конфигурацию импорта из DXF (TpaSpa.DxfCad.v2.dll): возможность применения множителя к заданию параметров, экстраполированных из Уровня или Блоков.

2.5 Версии 1.4.0 (10 апреля 2015 г.) - 1.4.9 (30 марта 2016 г.)

Версия 1.4.9 (30 марта 2016 г.)

Новинки:
- Добавление в импорт из формата ISO (TpaSpa.IsoToTpa.v2.dll): начальные символы расширены для распознавания правильного формата
- Изменение применения технологии по умолчанию, в импорте из внешнего формата (DXF, ISO,..): задание не меняет значение свойства *Конструкция* (поле B)

Исправления:
- Устранены проблемы графического представления ориентированных профилей
- Устранены проблемы графического представления конусов (избыточная фрагментация в представлении вертикального габарита)
- Устранены проблемы графического применения накладки
- Устранена проблема, связанная с выполнением команды *Сгруппировать* (возникали ситуации неправильного выбора)
- Устранена проблема, связанная с выполнением команды *Вставить*, выбранной из меню графического контроллера (текст ASCII не обновлялся)
- Устранена проблема в окне открытия программы (при импорте из внешнего формата больше не было видно части, относящейся к индивидуализации)

Версия 1.4.8 (26 января 2016 г.)

Новинки:

- Добавление опции в конфигурацию импорта из DXF (TpaSpa.DxfCad.v2.dll): возможность принудительного задания направления вращения замкнутых профилей

Исправления:
- Устранены проблемы отображения при работе на операционной системе с восточным языком (см.: Меню, качество шрифтов)
- Устранены проблемы развертки ориентированных текстов (размер и размещение относительно символов)

Версия 1.4.7 (14 декабря 2015 г.)

Исправления:
- Устранена проблема в инструменте профиля Продлить (случай: удлинение отрезка до горизонтальной прямой, задавала координату x)
- Устранена проблема в инструменте профиля Задать технологию (не сохранялся параметр Преследование касательной)
- Устранена проблема в окне Файл Открыть (навигация между папками)
- Устранена проблема в Конфигурации (случай: изменение настроек в Клиентских разделах)
- Устранены проблемы в параметрическом программировании (математический оператор "?" и функция "geo[sub;..]")
- Устранена проблема запуска с заданным системным языком (в случае испанского)

Версия 1.4.6 (19 октября 2015 г.)

Новинки:
- Добавлены функции к коду операции "STOOL: Фрагментировать и Линеаризовать" (возможность фрагментировать также линейные отрезки и не линеаризировать дуги)

Исправления:
- Устранены проблемы графического представления ориентированных профилей

Версия 1.4.4 (15 октября 2015 г.)

Новинки:
- Добавление в режим ДЕМОНСТРАЦИОННЫЙ (команда выбора уровня функционирования: Essential, Базовый, Профессиональный)
- Добавление выбора цвета для применения к Стилю приложения
- Добавления в инструменты "Масштаб" и "Растянуть" (интерактивное считывание масштабного коэффициента)
- Добавления в "Общие инструменты" (интерактивное автоматическое считывание на вершинах габаритного прямоугольника)
- Добавления в меню "Отобразить" (группа "Информация": замкнутый профиль, многократные подготовки, входные/выходные отрезки)
- Добавление в Персонализацию (Добавить выборы)
- Добавление в Персонализацию (Восстановить закладки)
- Добавление в Персонализацию (Длина инструмента для графики подготовки при нулевом габарите)
- Изменено локальное меню графического контроллера (добавлены команды для профилей, созданы группы "Зум" и "Навигация")
- Добавлена команда "Сгруппировать" (меню Редактирование, группа Изменить): обеспечивает последовательность выборов в списке
- Добавлена команда "If...ElseIf..Else...EndIf" (меню Редактирование, группа Блоки)
- Добавление в ввод логического блока (меню Редактирование, группа Блоки): применение блока к группе выборов
- Добавление в задание закругленной грани (выборы начала/конца по касательной)

Исправления:
- Устранены проблемы разрешения параметрического программирования (функция: geo[pxf/ pyf/ pzf;..], geo[lparam;..])
- Устранена проблема в управлении Последовательностями: выбор в графической зоне ошибался
- Устранена проблема команды Отменить (Undo), выполняемой после ввода Логического блока
- Устранена проблема выполнения команд Замены, вызываемых из Общего вида (выполнялись замены только на грани-детали)

Версия 1.4.2 (13 июля 2015 г.)

Новинки:
- Добавление в инструмент "Создать текст" (Угол наклона)
- Добавление в коды создания текста (Угол наклона)
- Добавление в развертку замкнутых кардинальных кривых (развертка оценивает касательные на начальной/конечной точке для сохранения этой развертки при изменении начальной/конечной точки)
- Добавление модуля импорта файла DXF (ТраSpa.DxfCad.v2.dll)

Исправления:
- Устранена проблема создания надписей: пересмотрены пределы преобразования ребра в дугу (случаи слишком ребристого профиля)
- Устранена проблема в Конфигурации геометрии детали (выбор в списке начала координат грани)
- Устранена проблема в окне Файл Открыть: при множественном выборе типа (случай модуля импорта формата) не удавалось подтвердить выбор файла
- Устранена проблема применения графической команды *Расширенный зум* (которая больше не работала)

Версия 1.4.1 (5 мая 2015 г.)

Новинки:
- Добавление в конфигурацию: "Эпсилон, используемый в логических сравнениях"
- Добавление в персонализацию: персонализация графики для геометрических профилей, выемок, конструкций
- Устранена проблема управления последовательностями: перемещение линий в режиме буксировки (drag&drop) применяет выбор вставки сверху/снизу
- Добавление в окне задания фиктивной грани: добавлена расчетная информация для грани

Исправления:
- Устранена проблема управления последовательностями: перемещение нескольких линий в режиме буксировки (drag&drop) инвертировало порядок ввода
- Устранена проблема в Сокращении профилей (инструмент и операция STOOL): случаи сокращения до дуги с неизменной координатой (могла выбираться неправильная координата для программирования центра)
- Устранена проблема применения команд "Создать фиктивную грань из геометрии", "Создать моделирование, исходя из геометрии" (случай круга)
- Устранены проблемы графического представления: применение накладки габаритов

Версия 1.4.0 (10 апреля 2015 г.)

Новинки:
- Добавление в конфигурацию: подключение для управления форматом Unicode для программ TCN
- Добавление в конфигурацию: подключение для управления файлами XML для сохранения конфигурации среды
- Добавление в конфигурацию: отображение маршрутов и значимых файлов
- Добавление в конфигурацию: подключение **"**Завершить операции считывания внешней программы**"**
- Добавление в конфигурацию: задание размера для полей Клиентских разделов
- Добавление в конфигурацию: изменение профилей для Насечки с острыми ребрами
- Расширение функции "Автономный режим"
- Увеличение числа языков, предусмотренных в базовом функционировании
- Добавления в реализацию файла прототипа (случаи дифференцирования в зависимости от типа детали; реализация уровней доступа и типологии)
- Добавлена реализация распознавания первичной или вторичной копии, в случае нескольких копий ТраCAD
- Добавлена реализация автоматических восстановлений при считывании файла TCN
- Добавление в группу команд Правки: команда Выполнить повторно
- Добавление комплексного инструмента "Утвердить профили"
- Добавление в комплексный инструмент "Соединение профилей" опции "Сократить поиск соответствий"
- Добавление в группы Найти/Заменить управления соответствиями также по Имени операции
- Добавление в инструмент Перенести (размещение габаритного прямоугольника)
- Добавления в инструмент Создать текст (режим определения расстояния и распределения; распределение на конусе; RightToLeft)
- Добавления в коды создания текста (режим определения расстояния и распределения; распределение на геометрическом элементе; RightToLeft; восстановление технологии из внешней операции)
- Добавление в инструмент "Создать шрифт, исходя из геометрии" задания положения для последующего размещения
- Добавление в инструмент Начало/Конец профиля (тип: Приближение)
- Добавление в Начало/Конец профиля подготовки (типы: Приближение, Удаление, Перекрытие)

- Добавлена функция в операцию типа STOOL (возможность восстановления технологии из внешней операции)
- Добавлена функция в операцию типа STOOL (возможность восстановления операций с уровня предыдущего вызова)
- Добавлена функция к коду операции “STOOL: Подача по Z” (выбор оси развертки)
- Добавление операций в группу STOOL: “STMULTI: Повторение профиля”, “STFILLET: Скруглить профиль”, “STCHAMFER; Выполнить фаску на профиле”
- Добавлена операция “NOP: Null operation”
- Добавления в реализацию операций подготовки, для специальных геометрий (закругленные грани или поверхности)
- Добавление в операцию “A32: Двойная дуга” (промежуточное скругление)
- Добавлены функции в параметрическое программирование (функции: geo[param;..], geo[lparam;..])
- Добавления в функцию интерактивного размещения (указатель на вершинах грани)
- Добавления в отображение профилей подключения и опций накладки габаритов
- Добавление в меню Отображение опции "Показать фиктивные грани"
- Добавление в строку состояния приложения (*Применить к копии операций*)
- Добавление в настройки Индивидуализации возможности подтвердить ввод операции также при наличии ошибки
- Добавление в использование свойства операции V ("Удалить по завершении разработки"; распространение нулевого значения)

Исправления:
- Заблокирована интерпретация буксировки программы в ТраCAD (открытие буксировкой) при выполняемой команде

2.6 Версии 1.3.0 (10 февраля 2014 г.) - 1.3.11 (31 марта 2015 г.)

Версия 1.3.11 (31 марта 2015 г.)

Исправления:
- Устранена проблема, связанная с окном запуска ТраCAD (случай ТраCAD, которая остается на втором плане)
- Устранена проблема технологического задания, применимая при импорте формата (не задавались свойства)
- Устранена проблема в конфигурации (случай задания Клиентского раздела, со перемещением в список пунктов раздела)
- Устранена проблема разрешения вложенных вызовов (случай нерешения)
- Устранена проблема разрешения закругленных граней (случай закругления грани, заданной на оси Y)

Версия 1.3.10 (27 февраля 2015 г.)

Исправления:
- Устранена проблема, связанная с окном запуска ТраCAD (случай ошибок памяти при нескольких запусках подряд)
- Устранена проблема применения инструмента Повторение (случай числа повторений свыше 32767)
- Устранена проблема считывания программы (случай параметров операции типа строки, с заданием, содержащим подстроки типа “WC”, “WB”: они могли привести к интерпретации прямых заданий некоторых свойств, например, комментариев, конструкции)
- Устранена проблема графического представления дуг, запрограммированных на закругленных гранях (случай радиуса закругления грани меньше радиуса дуги. Дуга могла отображаться с избыточной линеаризацией)

Версия 1.3.9 (27 января 2015 г.)

Новинки:
- Изменения в применении указателя на вершинах грани (положение мыши притягивается на сторонах грани с добавлением возможности блокировки одной из двух осей координат)
- Добавлена опция в конфигурацию импорта из DXF (возможность исключения всех блоков)

Исправления:
- Изменения, связанные с применением коррекции на инструмент (сокращение профиля при корректировке конусов)
- Решены ситуации неправильного определения направления поворота трехмерной дуги
- Устранены проблемы импорта из формата DXF (максимальное число элементов профиля, исключение незаданных блоков)

Версия 1.3.8 (04 декабря 2014 г.)

Новинки:
- Изменение в графическом представлении недействительной операции (случай точечной операции или операции подготовки с недействительным рабочим кодом): не применяет технологическую информацию, считанную операцией

Исправления:
- Устранены проблемы в командах группы **Найти/Заменить** (случай замены недействительного кода)
- Устранены проблемы управления инструментом **Применить подачу Z** (дуга #xy ошибалась при переносе центра*)*

Версия 1.3.7 (11 ноября 2014 г.)

Исправления:
- Устранена проблема на этапе импорта программы с применением технологии (случай ошибочной обработки текстов, содержащих сложные коды)
- Устранены проблемы подключения в окне Конфигурации (подключался уровень Ремонтника даже при доступе Изготовителя)
- Устранены проблемы в окне инструмента *Создать тексты* (не был реализован выбор *Инвертировать отрезок)*
- Устранены проблемы при применении **Общих инструментов** (перенести, повернуть, зеркальное представление) (случай профиля в отсутствие программирования начальной точки)
- Устранены проблемы в решении Отрезка начала/конца профиля типа Трехмерная дуга
- Устранена проблема аномального замедления в графике ориентированного профиля

Версия 1.3.6 (09 октября 2014 г.)

Исправления:
- Устранена проблема отображения операций, примененных на закругленной грани

Версия 1.3.5 (18 сентября 2014 г.)

Новинки:
- Добавлено управление учетной записью локального производителя
- Добавлено управление автоматическими обновлениями (т. е. для изменения базы данных операций)
- Изменение в управлении окна ввода данных текущей операции (в случае автоматического подтверждения управление не переходит к следующей операции)
- Добавление в интерактивные функции обучения позиции (реализованы клавиши направления для дискретного перемещения мыши)
- Добавление в конфигурацию: различные в функции технологических проверок
- Добавление функции "Офлайн"

Исправления:
- Устранены некоторые проблемы в окне Конфигурации

Версия 1.3.4 (27 августа 2014 г.)

Исправления:
- Устранена проблема управления окном упорядочения, используемым из раздела клиентских настроек

Версия 1.3.3 (31 июля 2014 г.)

Новинки:
- Добавление подключений на уровне Сервиса/Ремонтника (возможность изменения состава меню и палитр операций в конфигурации)
- Добавление в конфигурацию: различные в функции технологических проверок
- Добавление в конфигурацию: уровень доступа к Глобальным переменным
- Добавления в персонализацию: подключения вспомогательных столбцов текста ASCII
- Добавления в персонализацию: подключение автоматического подтверждения в обучении операции
- Добавления в персонализацию: подключение автоматического сохранения программы

- Добавления в интерактивное расположение главного рабочего окна
- Изменения, связанные с распознаванием и управлением языками с написанием справа налево
- Добавление в реализацию последовательностей: добавлена функция буксировки (drag&drop), столбец цвета

Исправления:
- Добавление распознавания папки или файла, заданного со ссылкой (например, быстрый выбор на рабочем столе)

Версия 1.3.2 (15 мая 2014 г.)

Новинки:
- Добавление взаимодействия в строке состояния (выбор текущей строки)
- Изменения в *трехмерном отображении* вертикальных подготовок
- Изменения в отображении профилей в Коррекции на инструмент (теперь подготовка приводится на скорректированном профиле)
- Добавление в группу команд для *Позиционирования на строку*
- Добавление в группу *Измерения*
- Добавление в группу *Индивидуализировать виды* (отображение габарита профилей)
- Добавление в Строку кнопок в *Зоне задания операций* (команда: "Перезадать операцию")
- Добавление в *Список предпочтительных операций* (быстрый выбор)
- Добавление в функцию *Зона текста ASCII* (изменение свойств из ячейки заголовка столбца, столбца цвета)

Исправления:
- Неправильное соответствие команд в меню Графика (были перепутаны команды "Зум Увеличение" и "Зум Уменьшение")

Версия 1.3.1 (2 апреля 2014 г.)

Новинки:
- Добавление в конфигурацию: выбор для интерпретации сигналов при реализации *Клиентских разделов* программы
- Изменение реализации диагностических сигналов при минимизированной зоне отображения

Исправления:
- Устранены проблемы при использовании нестандартного разрешения
- Устранена проблема при запуске конфигурации экспортера в формат DXF

Версия 1.3.0 (10 февраля 2014 г.)

Новинки:
- Добавлена функция "Essential" (распознавание аппаратного ключа USB)
- Добавлено управление двойной конфигурацией ("Среда чертежа", "Среда станка")
- Добавлена функция "Автономный режим"
- Добавление в конфигурацию: выбор для применения "Запрограммированные вложенные вызовы"
- Добавление в конфигурацию: выбор для "Автоматическое присвоение переменных r"
- Добавление в конфигурацию: выбор для параметрического программирования свойства "V" операции
- Добавление в конфигурацию: выбор для прямого изменения в тексте ASCII свойства "B" операции
- Добавление в конфигурацию: подключение для управления "Глобальными переменными" и команда задания соответствующего списка
- Различные добавления в конфигурацию в задании грани-детали
- Добавление в конфигурацию: подключение "Закругленных граней"
- Добавления в конфигурацию Задания матрицы детали и модулей Преобразования формата при записи (фрагментация дуг, заданных на закругленных гранях)
- Добавления в конфигурацию модулей Преобразования формата при считывании (Импорт как подпрограмма)
- Добавлена возможность отключения всех реальных граней
- Добавление в конфигурацию: реализация палитры операций в единой группе
- Добавления в реализацию файла прототипа (случаи дифференцирования в зависимости от типа детали; реализация уровней доступа и типологии)
- Изменения в персонализации технологических операций по умолчанию: задание теперь возможно также для нереализованных реальных граней
- Добавлены команды трансформаций в окне задания Переменной геометрии (фиктивная или автоматическая грань)

- Добавление геометрических типов в разделе Моделирование (дуга и линия с решением непрерывности касательной; скругление на ребрах прямоугольника; овал)
- Расширение в автоматическом распространении переменных "r" (выполняет распространение также переменных, используемых с символическим именем)
- Расширение в применении "Запрограммированных вложенных вызовов" (применение во всех гранях, распознавание задания подключенных или исключенных граней, программирование точки применения)
- Добавление в команды Найти / Заменить (кнопка для автоматического задания текущей операции; команда для создания перечня найденных соответствий)
- Добавление в команде "Создать шрифт, исходя из геометрии" (управление с множественным заданием символов)
- Добавления в окне Общие инструменты (реализация меню справки параметрического программирования)
- Добавлена группа "Информация" в меню "Отобразить"
- Добавления в реализацию окна ввода данных текущей операции (сброс выбора в палитре команд; запись в память состояния открытия узлов)
- Изменения в окне "Клиентские операции" (возможность задания новых кнопок в палитру операций)
- Добавления в реализацию блокировки секций программы (возможность скрытия заблокированных секций)
- Параметрическое программирование, добавление функции
- Изменения в базе данных операций: задан размер параметров сложных операций
- Изменения в базе данных операций (операция SSIDE [2021]: добавлены параметры для задания точки применения)
- Изменения в базе данных операций (операция NSIDE [2020]: добавлены параметры для задания закругленной грани)
- Изменения в базе данных операций (операция КООРДИНАТЫ [1112]: добавлены параметры для задания высоты шрифта и количества цифр дробной части в надписях; добавлен интерактивный режим координат; изменен шрифт записи)
- Изменения в базе данных операций (операция ПРЕДУПРЕЖДЕНИЕ [2019]: добавлен параметр для переключения в состояние ошибки в режиме выполнения)
- Добавлен пункт управления "Глобальными переменными" в меню справки параметрического программирования
- Добавление в конфигурации импорта из DXF (единица измерения импорта)

Исправления:
- Решены случаи решения плоскостей из моделирования (случаи совпадающих плоскостей)
- Решены случаи реализации быстрого выбора команд правки (вырезать, копировать, вклеить)
- Устранена проблема персонализации операций в строке состояния
- Устранены некоторые небольшие проблемы графического представления (габарит профили)
- Устранены проблемы при выполнении "Смены станка" (восстановление общего вида и стиля)
- Устранены некоторые проблемы в приложении ТраWorks (при выполнении команды "Проверить операцию"; создание / удаление узла; управление сменой станка)
- Устранена проблема в приложении ТраWorks (при выполнении команды "Проверить операцию")
- Устранены некоторые мелкие проблемы

2.7 Версии 1.1.0 (06 декабря 2012 г.) - 1.2.4 (10 октября 2013 г.)

Версия 1.2.4 (10 октября 2013 г.)

Исправления:
- Устранена проблема решения кода операции STZLINE (Линеаризовать по Z): решался отдельный профиль для каждого отрезка исходного профиля

Версия 1.2.3 (10 сентября 2013 г.)

Новинки:
- Изменен модуль экспорта в формат Edicad (изменения формата созданного текста так, чтобы его могли читать даже очень старые версии Edicad)
- Добавлена интерпретация коррекции на инструмент в модуле экспорта в формат Edicad

Исправления:
- Устранена проблема импорта программы в формате Edicad (не выполняла задания рабочего кода и/или параметров для первой соответствующей операции базы данных операций, обычно - Прорезь X)

Версия 1.2.2 (22 июля 2013 г.)

Исправления:

- Неправильное графическое представление инструмента в программировании ориентированной подготовки

Версия 1.2.1 (08 июля 2013 г.)

Новинки:
- Добавлена команда "Найти и выбрать" (меню Редактирование, группа Изменить): выполняет селективные выборы операций
- Добавлены команды в локальное меню Текста ASCII (выбор части профиля)
- Добавлена команда "Зум Увеличение/Уменьшение"
- Добавлены персонализации по графическому взаимодействию
- Усовершенствован выбор графического представления текущей операции в трехмерном изображении (3D)
- Добавление в конфигурацию: настройка для подключения функции "Автономный режим"
- Добавления в конфигурацию модулей Преобразования формата (при считывании и записи)
- Добавление в конфигурацию: настройка для подключения окна "Технология"
- Усовершенствовано управление окном ошибок (отображает максимальное число ошибок)
- Изменены критерии ассоциации изображений в Свойстве 'O'
- Усовершенствовано интерактивное управления в Инструментах профиля (состав меню Привязка, графические построения)
- Усовершенствовано управление меню "Предпочтительные операции" (команды)
- Усовершенствовано распознавание текущего языка (более высокая гибкость в кодировании языка)
- Добавления в загрузку программы в формате Edicad/ TpaEdi32 (восстановление клиентских кодов профиля)
- Изменены критерии инициализации Оптимизатора программы (подключения для преобразований, оснащения)
- Добавлена инсталляция модуля для Оптимизации, совместимого со средой VB6
- Добавлены характеристики загрузки стандартной технологии (опрос держателя инструментов и устройства для смены инструментов)
- Обновлена справка операций на (английском) языке

Исправления:
- Устранены некоторые проблемы управления вложенными вызовами (блокировка приложения или неполное графическое представление)
- Устранена проблема в инструменте "Зеркальное отображение относительно горизонтальной оси" (не задавал в окне узнанное положение)
- Устранена проблема применения инструмента "Соединение между профилями" (случай профилей с изменением глубины)
- Устранены некоторые проблемы в окне Конфигурации
- Устранена проблема в окне Открытие программ (исключение генерировано наличием ресурсов типа "облако")
- Устранена проблема в стандартном окне представления технологии (блокировка программы в случае конфигурации при отсутствии подключенных групп)
- Устранены некоторые проблемы оптимизации программы (задание текущего оснащения)
- Устранены некоторые проблемы в приложении TpaWorks (окно атрибута "Вспомогательные" параметра операции: управление таблицами задания "Активного состояния" параметра)
- Устранена проблема в окне задания текущей операции (случай ограничения активированного состояния параметра при сравнении со значением -равно или не равно-)
- Устранены некоторые мелкие проблемы
- Устранена проблема деинсталляции приложения (случай "безмолвного" выполнения, в любом случае, были реализованы опросы)

Версия 1.2.0 (23 апреля 2013 г.)

Новинки:
- Добавление в конфигурации раздела "Режим выполнения" (конфигурирует активные режимы выполнения)
- Добавление в конфигурации: выбор для осей поворота в "Ориентированной подготовке" (конфигурирует плоскость вращения для оси B)
- Усовершенствовано соответствие между конфигурацией и функциональностью аппаратного ключа (сохраненная конфигурация выравнивается с оперативной функцией)
- Добавление в конфигурацию при работе с ключом Enterprise (пункт выбора "Защищенная конфигурация", в меню Среда, группа Общие уставки)
- Добавлено управление "Продвинутой конфигурацией"
- Добавлено управление моделированием детали для экструзии (требует подключения специального устройства)
- Добавлена команда "Решить" (меню Редактирование, группа Изменить): решает параметрические программирования

- Параметрическое программирование, добавление нескольких функций (geo[sub;..], geo[param;..])
- Усовершенствовано восстановление общего вида при запуске приложения TpaCAD (синхронизировано на изменении второй цифры основной версии)
- Усовершенствовано управление в окне Открытие программ
- Усовершенствовано управление командами навигации логических ветвей текста ASCII
- Усовершенствовано отображение координат операций (ограничивается отрезками профиля)
- Усовершенствовано управление программируемыми клиентскими ошибками (добавлено разграничение между базовыми ошибками и персонализируемыми клиентскими ошибками)
- Усовершенствованы диагностические сигналы при вызове Клиентского оптимизатора (использует более конкретное сообщение)
- Усовершенствовано управление в окне изменения персонализируемых сообщений
- Усовершенствован интерфейс при задании поля типа ЦВЕТ клиентского раздела
- Усовершенствовано управление командой "Создание фиктивной грани из запрограммированной геометрии" (выбор в режиме грань-деталь)
- Усовершенствовано управление в окне содействия при параметрическом программировании
- Усовершенствовано применение технологии по умолчанию для операций подготовки (различие на основе подтипа)
- Добавлены характеристики загрузки стандартной технологии среды Albatros
- Усовершенствовано управление в стандартном окне представления технологии (соответствие типа и изображения инструмента)
- Добавлена функция в компоненты Преобразования из/в формат ISO (персонализация кода G-ISO точечной операции)
- Изменения в базе данных операций (операция ПОДГОТОВКА РЕЗЦА [95]: задает подтип 2)
- Различные изменения в текстах базовых макропрограмм
- Добавлена справка для операций на (итальянском) языке
- Добавлено руководство на (английском) языке по приложению TpaWorks
- Добавлено специальное руководство по моделированию (на итальянском, английском языках)

Исправления:
- Устранена проблема запуска TpaCAD двойным щелчком на файле-детали из Управления ресурсами (если маршрут содержит пробелы)
- Устранена проблема при задании фиктивных граней (ситуации исключения при выполнении команды "Удалить все")
- Устранены некоторые небольшие проблемы в окне задания текущей операции
- Устранена проблема обновления окна "Переменных <j>"
- Устранена проблема оптимизации программы (ошибка при регистрации клиентских разделов в матрице детали)
- Устранены некоторые небольшие проблемы графического представления (представление стандартной сетки, ситуация покрывающих цветов)
- Устранена проблема в конфигурации импорта из DXF (на странице "Макропрограммы и уровни" утрачивались префиксы параметров)
- Устранены некоторые мелкие проблемы

Версия 1.1.4 (05 марта 2013 г.)

Новинки:
- Усовершенствовано управление окном ошибок (кнопка "Перейти к строке" активна даже в случае ошибки из общего раздела программы)
- Усовершенствовано управление заданием активизированного вида при смене "Активизированного вида" (активный табулятор общей информации программы)
- Усовершенствованы процедуры загрузки файла изображения (изменены критерии доступа)
- Добавления в персонализацию отображения программы
- Добавление в базу данных операций (операция WARNING, рабочий код 2019)

Исправления:
- Устранена проблема при открытии программы в формате Edicad/TpaEdi32 (случай замены рабочих кодов)
- Устранена проблема при повторном открытии файла сообщений персонализаций
- Устранены проблемы применения "Поворота в картезианской плоскости" (инструмент и операция)
- Устранена проблема отображения выходного отрезка на профиле (расширенный случай последнего запрограммированного отрезка)
- Устранены некоторые небольшие проблемы графического представления
- Устранены некоторые мелкие проблемы

Версия 1.1.3 (11 февраля 2013 г.)

Новинки:

- Добавлена команда "Преобразовать архив программ" (меню Файл): для считывания и архивирования партии программ
- Добавления в базу данных операций (доведены до трех узлы логических условий кодов SUB, STOOL, "Глобальные функции")
- Усовершенствовано управление командой "Создание фиктивной грани из запрограммированной геометрии" (выбор в режиме грань-деталь)
- Добавление в конфигурацию импорта из DXF (значения по умолчанию параметров, распознаваемых на уровнях)
- Добавлено руководство на (французском) языке

Исправления:
- Устранены проблемы изменения типа программы
- Устранена проблема операций выемки (доступность некоторых технологических параметров)
- Устранены некоторые мелкие проблемы

Версия 1.1.2 (22 января 2013 г.)

Новинки:
- Добавления в графические персонализации текущей программы (цвет, текстура)
- Добавление в отображение текстур
- Усовершенствована интерактивность при задании операции "Разметка"
- Добавлен предварительный просмотр текста при вводе операции создания текста из системных шрифтов
- Усовершенствовано управление созданием надписей с использованием клиентского шрифта (автоматическое распределение)
- Усовершенствовано управление файлом отладки (для применения подпрограммы/микропрограммы)
- Усовершенствована криптография макропрограмм
- Усовершенствовано управление окном отображения расширенного списка операции
- Добавления в дополнительное управление Просмотром графического представления
- Добавления в конфигурации модулей экспорта формата
- Добавления в конфигурацию команды "Предварительный просмотр для оптимизации"
- Добавлено руководство на (испанском) языке

Исправления:
- Устранена проблема предварительного просмотра надписей с выбором системных шрифтов (случаи неподдерживаемых стилей)
- Устранена проблема ассоциации сообщений с операциями (импортер из DXF)
- Устранены некоторые мелкие проблемы

Версия 1.1.1 (17 декабря 2012 г.)

Новинки:
- Добавления в реализацию привязки к стандартной сетке
- Добавлена реализация ввода операций с автоматическим подтверждением
- Добавления в реализацию поля типа ЦВЕТ клиентского раздела
- Параметрическое программирование, добавление переменных аргументов (prgnum)
- Параметрическое программирование, добавление функции (geo[sub;..])
- Добавления в конфигурации модулей экспорта формата

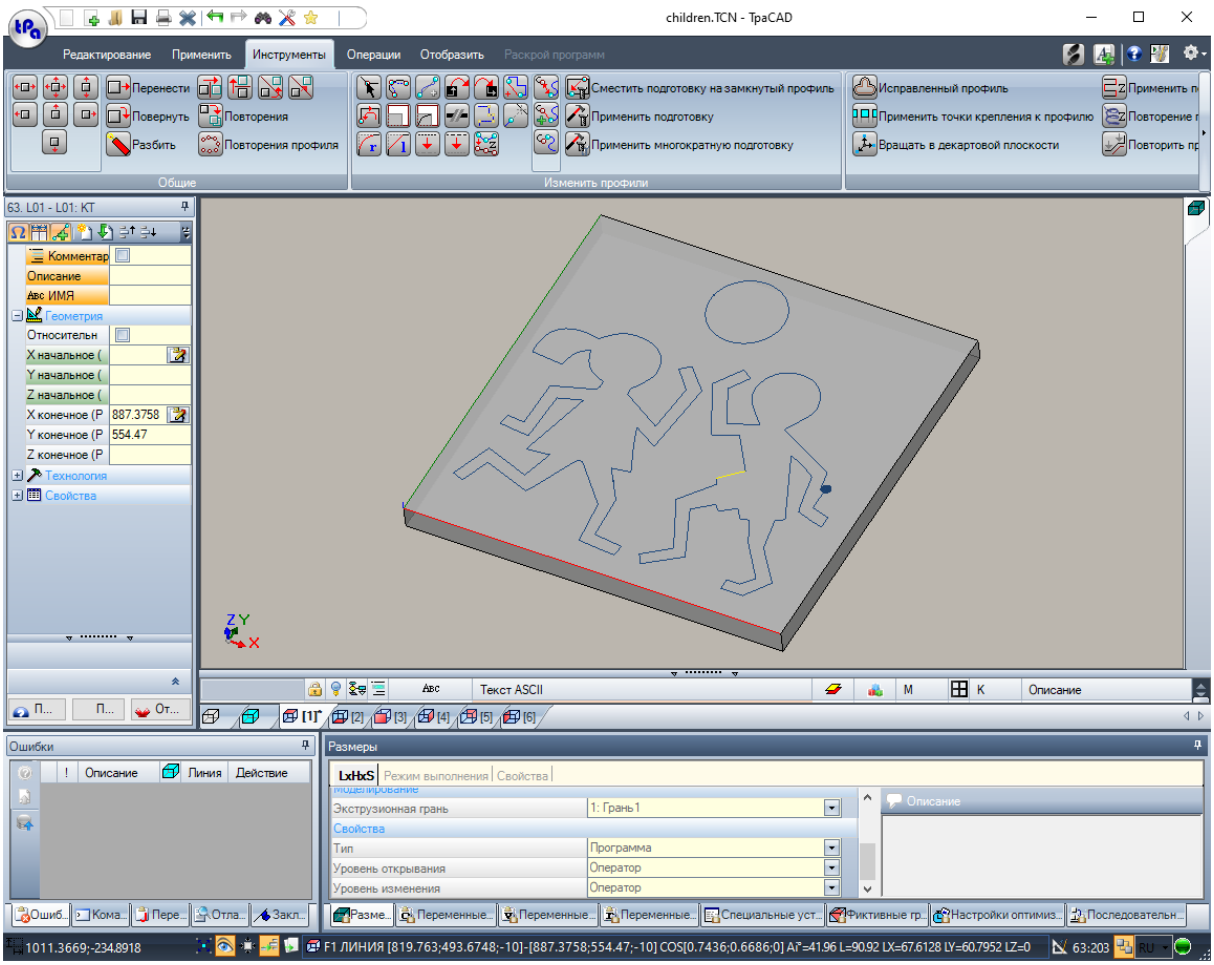
Исправления:
- Устранена проблема управления языками, добавленными в дополнительные компоненты (оптимизатор, импортер из DXF)
- Устранена проблема при записи файла клиентских сообщений
- Устранена проблема управления табуляторами в таблице текста ASCII
- Устранена проблема технологического задания для точечных операций
- Устранены некоторые мелкие проблемы

Версия 1.1.0 (06 декабря 2012 г.)

- Первый официальный выпуск приложения

3 Графический интерфейс

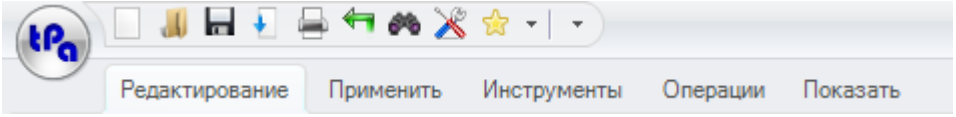
3.1 Порядок отображения



Рабочая зона TpaCAD разделена на следующие основные зоны:
меню Применение
Это главное меню и соответствует выбору кнопки с пометкой ***tpa*** (как на рисунке) или со словом ***Файл*** (в зависимости от выбранного стиля меню). Оно содержит команды для управления программными файлами: Новый, Открыть, Сохранить, Печать, Закрыть. Некоторые из этих команд представлены в строке меню быстрого доступа.
В дополнение к командам, касающимся непосредственного управления программами, меню позволяет получить доступ к окну Настроек и команде закрытия TpaCAD.

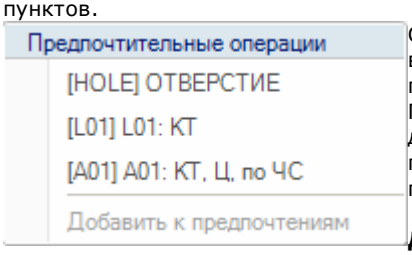
строка быстрого доступа

строка быстрого доступа - это строка инструментов с совокупностью команд, не зависящих от отображенной в настоящий момент вкладки команд. Эта строка находится в верхнем левом углу рядом со значком приложения:



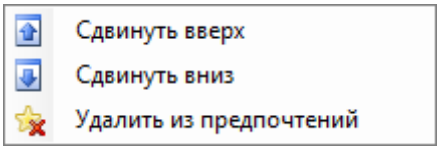
Содержащиеся в этой строке команды частично взяты из меню Применение, а также Команды вкладки Редактирования (Отмена, Восстановить, Найти и Заменить) и Таблицы технологии.

Выбор значка  открывает выпадающее меню с перечнем предпочтительных Операций, максимум 15 пунктов.



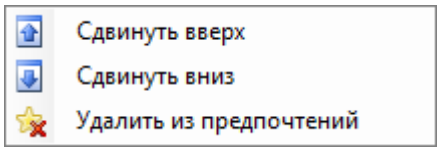
Список на рисунке является примером: при создании программы можно вводить указанные операции непосредственно из списка без доступа к палитре операций.
Первым 9 пунктам в списке присвоены горячие клавиши: от (Shift+F1) до (Shift+F9). Задание горячих клавиш зависит от положения в списке: при перемещении положения на одну операцию в списке меняются горячие клавиши.

Добавить к предпочтениям: добавляет текущую операцию к списку. Эта команда активирована на виде грани при непустом программном списке: если текущий список уже содержит максимально предусмотренные 15 пунктов, новый вводимый параметр стирает первый пункт списка.
Эта команда недоступна в меню, если палитра операций определена как невидимая в конфигурации TpaCAD: в этом случае предполагается, что список операций, требуемых для прямого ввода, подготавливается на этапе конфигурации приложения и не может меняться здесь напрямую.
При нажатии правой кнопки мыши на пункте операции открывается локальное меню, управляющее порядком и удалением пунктов в списке:

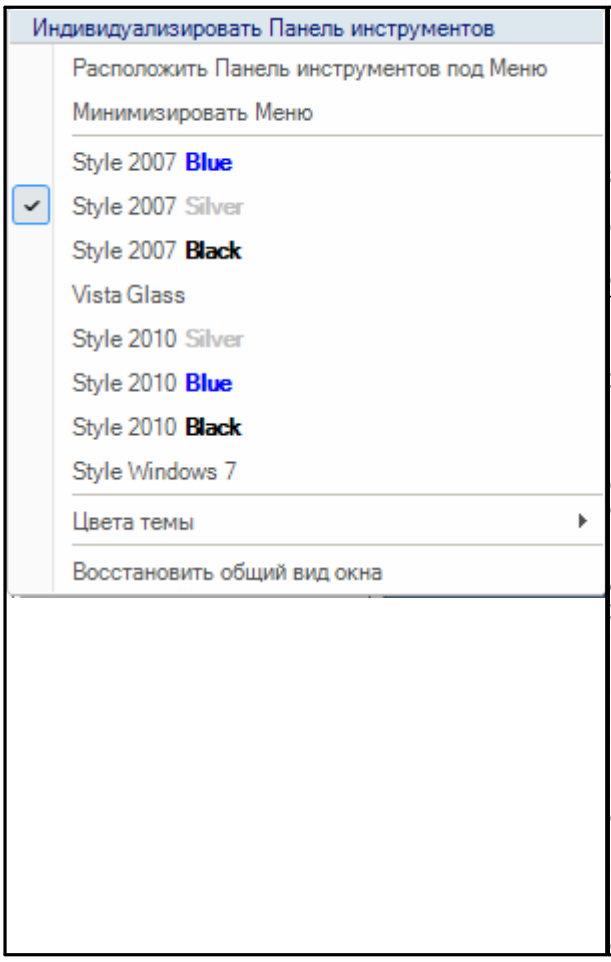


Команда для удаления недоступна в меню, если палитра операций определена как невидимая в конфигурации TpaCAD.

Выбор значка  открывает выпадающее меню со списком Предпочтительных инструментов.
Значок виден, если активировано управление предпочтительными инструментами из конфигурации TpaCAD.
Пункт меню **Добавить к предпочтениям** добавляет последний выбранный инструмент в список предпочтительных инструментов. Команда активна в виде грани и с непустым списком программ.
При нажатии правой кнопкой мыши на инструмент в списке открывается локальное меню, которое управляет порядком и удалением элементов в списке:



Выбор значка  в строке быстрого доступа открывает выпадающее меню



первые команды позволяют менять позиционирование многофункциональной строки (см. ниже)

Style 2007 Blue ..
Следует список пунктов для выбора стиля представления TpaCAD.

Цвета темы
Эта команда позволяет персонализировать цвет темы, связанный с текущим стилем.

Восстановить общий вид окна
Эта команда позволяет восстановить исходный графический вид программы.
При запуске TpaCAD восстанавливает общий вид, с которым было выполнено закрытие, восстанавливая размер и положение средств управления.

Многофункциональная строка
Команды, для похожих функций, организованы во вкладки в многофункциональной строке (Лента).
Эта строка содержит до 6 вкладок:
Редактирование, Применить, Инструменты, Операции, Показать, Раскрой,
которые, в свою очередь, содержат группы команд.
Состав вкладок может быть разным в зависимости от конфигурации TpaCAD.

Вкладка Редактирование:
- группа Примечания
- группа Правка
- группа Расположить в линию
- группа Задать свойство
- группа Задать

вкладка Применить:
- группа Применить к детали
- группа Начертить
- группа Блоки
- группа Продвинутые
- группа Измерение
- группа Проставление размеров

вкладка Инструменты:
- группа Общие
- группа Изменить профили
- группа Построения
- группа Раскрой профилей

вкладка Операции
- группы операций организованы исходя из конфигурации TpaCAD

вкладка Показать
- группа Навигация
- группа Виды
- группа Индивидуализировать виды
- группа Индивидуализировать вид при введении коррекции на инструмент
- группа Информация

вкладка Раскрой
- см. документацию по функции Раскроя

Некоторые группы команд, применяющиеся немедленно, имеют кнопку . Нажать эту кнопку для открытия руководства TpaCAD в разделе с описанием этих команд.
В правой части многофункциональной строки имеется две кнопки:

Открывает справку онлайн по программе

Открывает меню команд для доступной конфигурации. Состав меню зависит от предусмотренной для TpaCAD конфигурации. Меню может содержать команды для прямого доступа к другим файлам справки онлайн.

Многофункциональная строка может сворачиваться до значка из Персонализировать строку быстрого доступа , выбирая пункт в списке Свернуть меню, или с клавиатуры, нажимая сочетание клавиш [CTRL+F1], или двойным щелчком на названии вкладки этой строки.

Для восстановления многофункциональной строки из Персонализировать строку быстрого доступа выбрать в перечне пункт Развернуть меню, или с клавиатуры, нажимая сочетание клавиш [CTRL+F1], или двойным щелчком на названии вкладки этой строки.

Зона данных операции
Это зона, в которой отображаются и редактируются геометрические и технологические данные запрограммированной операции. Орган управления может перемещаться в пределах рабочего окна с различными подключениями анкеровки на краях этого окна или агрегироваться в пространстве других органов управления (Ошибки, Размеры,..). Для перемещения органа управления достаточно щелкнуть на строке заголовка органа управления и, удерживая нажатой левую кнопку мыши, переместить указатель на ячейку, которая из всех отображаемых ячеек меню соответствует требуемому положению.

Графическая зона представления детали
Это зона, в которой графически отображается программа, и которая соответствует текущему виду.
Общий вид
Можно делать выбор из следующих трех опций:
- графическое трехмерное (3D) представление детали: деталь представляется в трехмерном изображении (xyz) со всеми образующими деталь плоскостями (гранями) и со всеми примененными операциями.
- каркасное представление: отображается панель с разбивкой только на подключенных гранях детали параллелепипеда с примененными операциями представленных граней. Выбор каркасного представления может быть недоступен в зависимости от конфигурации TpaCAD.

Вид грани
Можно делать выбор из следующих трех опций:
- трехмерное графическое представление (см. общий вид) Текущая грань и операции грани выделяются другим цветом по сравнению с другими гранями и операциями. Операции других граней представлены серым или их отображение отключено.
- каркасное представление: текущая грань и операции грани выделяются другим цветом по сравнению с другими гранями и по сравнению с представленными операциями. Выбор каркасного представления может быть недоступен в зависимости от конфигурации TpaCAD.
- двухмерный вид: двухмерное графическое представление на плоскости xy грани и только запрограммированных на ней операций.

Зона списка операций в текстовом формате ASCII
Это зона, в которой отображается программа текущей грани в формате ASCII. Эта зона компилируется на виде грани и представлена в виде таблицы:
- каждая строка, соответствующая одной линии – запрограммированная операция;
- информация по каждой линии приводится в столбцах. В частности, приводятся отдельные поля свойств.

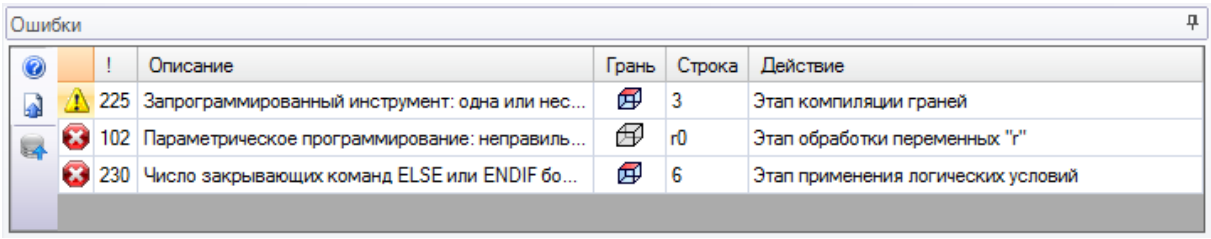
Из приведенных в таблице данных изменить можно только некоторые, также в зависимости от конфигурации TpaCAD.
Орган управления может перемещаться с различными подключениями анкеровки вокруг органа управления графического представления детали. Для этого щелкнуть правой кнопкой мыши на строке, ограничивающей зону, и выбрать пункт из меню.

Зона общих заданий детали

Это зона, предназначенная для отображения и задания общей информации программы: Размеры, переменные, Специальные разделы, Изменяемые геометрии, Последовательности.

Зона отображения ошибок, команд, переменных j, закладок
Эта зона предназначена для отображения вспомогательной информации, организованной во вкладках: ошибки, команды, переменные <j>, закладки и файлы отладки. TpaCAD меняет отображение окна в зависимости от выполняемой операции. Для принудительного изменения отображения достаточно выбрать соответствующую вкладку.

Зона **ошибок** содержит ошибки и предупреждения, возникшие при выполнении программы. Отображенные ошибки и предупреждения относятся к активизированному виду, например: в редакторе переменных <r> приводятся ошибки компиляции переменных <r>; на виде грани приводятся ошибки, выявленные на грани, и т. д.
На рисунке показан пример с тремя сигналами:



первый и третий - это сигналы ошибки, второй - предупреждение (warning).
Приведем описание столбцов, образующих окно ошибок:

- и : соответственно, значок ошибки и значок предупреждения. Третий значок может сигнализировать предупреждение в TpaCAD, но с сигналом, который переключается на ошибку при запросе выполнения программы (серьезное предупреждение).
- **!**: номер, определяющий сигнал
- **Описание**: описание сигнала
- **Грань**: графическое представление вида грани, к которой относится сигнал. При проведении курсора мыши по графическому представлению отображается сообщение справки с указанием номера грани
- **Строка**: номер программной строки или переменной, к которой относится сигнал
- **Вложенная операция на грани..**: графическое представление вложенной грани, от которой генерируется сигнал (столбец отображается только при необходимости)
- **Действие**: описание этапа обработки, на котором возникла проблема

Нажать кнопку для вызова контекстуальной справки для указанной ошибки.
Выбрать для перехода на операцию, на которой произошла ошибка
Выбрать для запроса комплексной обработки программы. При этом снова генерируется список ошибок.
Если на строке сбоку отображается также кнопка , это означает, что при обработке программы было подано слишком много сигналов: в этом случае в окне представляются только первые 100 ошибок.
Если зона отображения свернута, при возникновении ошибок эта зона становится видимой для привлечения внимания оператора.

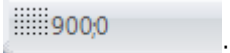
Зона **команд** отображает:

- исход команды, например, "Создание новой программы" или "Открытие существующей программы"
- последовательность выполненных команд.
- последовательность операций, которые должны выполняться при выполнении команды.

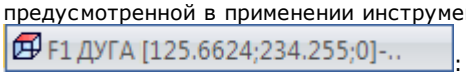
Зона **переменные J** отображает таблицу переменных <j>, состоящую из 10 строк и 10 столбцов.
Зона **Закладки** отображает таблицу, в которой можно задавать значимые вспомогательные позиции интерактивным процедурам. Эта зона может быть недоступна в зависимости от конфигурации TpaCAD.

строка состояния
В строке состояния отображается различная информация, которая зависит от выполняемых операций.



- 

: отображает положение мыши на виде грани. Если подключена интерактивная процедура, например, ввод элемента из меню Чертеж, отображаемый значок может означать тип активной привязки (на сетке или на объекте) и положение мыши принимает во внимание привязку

- : при выборе указывает, что подключена привязка к сетке
- : при выборе указывает, что активизировано отображение программы. Эта команда приводится в строке состояния в зависимости от конфигурации TpaCAD
- : при выборе указывает, что запрошено отображение прицела привязки в интерактивной процедуре с активным запросом привязки к граням и/или объектам
- : при выборе указывает, что следующий ввод операции будет выполняться после текущей операции. Удалите выбор, чтобы сделать запись до текущей операции: после вставки статус выбора будет автоматически изменен на Активный.
- : содержит последнее задание, выполненное для пункта *Применить к копии операций*, предусмотренной в применении инструментов. Можно напрямую изменить состояние.
- 

: отображает геометрическую и технологическую информацию о текущей операции, а также указывает грань применения (значок и номер грани)
- : указывает, что предыдущее поле отображает реальные координаты выбранной операции. При выборе Вида в Коррекции на инструмент при щелчке на значке осуществляется переход к отображению скорректированных координат и отображается значок . При повторном щелчке на значке происходит возврат к отображению реальных координат и снова отображается значок .
- 

: отображает номер по порядку текущей программной строки и общее число строк
- : изображение имеется только если в одной среде выполняется несколько экземпляров TpaCAD и активной не соответствуют первому запущенному экземпляру.
- : если он зеленый, информирует, что TpaCAD находится в ожидании команд. Если он красный, информирует, что TpaCAD выполняет этап обработки (например, выполняется обновление графики).

3.2 Список горячих клавиш и клавиш мыши

Окно Вид

[CTRL+W]: Окно увеличения
[CTRL+правая кнопка мыши]: Зум Увел.-Умен.
[CTRL+Shift+W]: Масштабировать Предыдущий
F6: команда Масштабировать Продления
X: вращение вверх вокруг горизонтальной оси
[Shift+X]: вращение вниз вокруг горизонтальной оси
Y: вращение влево вокруг вертикальной оси
[Shift+Y]: вращение вправо вокруг вертикальной оси
Z: вращение по часовой стрелке
[Shift+Z]: вращение против часовой стрелки

F3: активирует каркасное представление
F2: активирует 3D-вид
F4: активирует 2D-вид
F5: перечертить
F7: активирует отображение коррекции на инструмент
F8: активирует отображение логических условий

Фиктивные грани

[CTRL+щелчок левой кнопкой мыши]: выбирает или отменяет выбор строки в ячейке заголовка строки

Последовательности

[CTRL+щелчок левой кнопкой мыши]: выбирает или отменяет выбора линии в ячейке заголовка строки
[Shift+(левая кнопка мыши нажата)]: запускает выбор зоны
[Shift+CTRL+(левая кнопка мыши нажата)] : операции в выбранной области добавляются к текущим операциям в таблице
[CTRL+(щелчок левой кнопкой мыши)] : выбирает или отменяет выбор операции, ближайшей к положению мыши
[щелчок левой кнопкой мыши]: перемещает текущую линию к операции, ближайшей к положению мыши, удаляя все выбранные элементы

Меню для вставки геометрических объектов и получения координат

I: меняет направление вращения дуги
L: переключает от дуги на линию при создании полилинии
A: переключает от линии на дугу при создании полилинии
C: закрывается в начальной точке при создании полилинии
P: использует последнюю введенную точку
F: привязка между гранями
D: привязка к глубине
G: привязка к сетке
X: заблокирует ось X
Y: заблокирует оси Y
T: принудит выход касательного отрезка от отрезка
Z: отменяет последний штрих
S: привязка к объекту, активирует команды привязки
[Enter]: завершает вставку
[Escape]: отменяет фазу вставки

Меню Привязка к объекту

[CTRL+P]: привязка к запрограммированной точке
[CTRL+N]: привязка к координатам точки ближайшей к курсору
[CTRL+M]: привязка к середине
[CTRL+C]: привязка к центру дуги
[CTRL+I]: привязка к точке пересечения
[CTRL+O]: привязка к точке, перпендикулярной отрезку
[CTRL+T]: привязка к точке, касательной отрезку
[CTRL+Q]: привязка к точке смены квадранта
[CTRL+E]: привязка на ребро грани
[CTRL+F]: привязка к закладке
[CTRL+X]: привязка к точке в профиле моделирования

Выбор операции

Стрелка вверх: перемещает активную строку к предыдущему блоку
Стрелка вниз: перемещает активную строку к следующему блоку
Предыдущая страница : перемещает активную строку на 10 строк раньше
Следующая страница: перемещает активную строку на 10 строк позже
Начало: перемещает активную строку на первую строку программы
Конец: перемещает активную строку на последнюю строку программы

Выбор в графическом виде

[Shift+левая кнопка мыши]: выбирает операции указанной зоны
[Shift+левая кнопка мыши+ALT: выбирает операции указанной зоны, расширяя выделение на весь профиль
[Shift+левая кнопка мыши+CTRL: выбирает операции указанной зоны, сохраняя предыдущие выборы

Выбор в тексте ASCII

[Shift+левая кнопка мыши]: выбирает от активной строки до линии, на которую указывает мышь
[CTRL+(левая кнопка мыши)]: выбирает или отменяет выбор программной строки, указанной мышью

Общие команды выбора

[CTRL+A]: выбирает все операции грани

Общие команды изменения

[CTRL+C]: копирует в примечания выделенный текст или выделенные операции
[CTRL+V]: вставляет текст или операции из примечания
[CTRL+X]: вырезает выделенный текст или операции
[Удалить]: удаляет операции

[CTRL+ Z]: отменяет последнюю команду
[CTRL+Y]: восстанавливает последнюю отмену
[CTRL F1]: минимизирует или максимизирует ленту
[CTRL+2]: активирует зону данных операций
[CTRL+3]: активирует зону списка операций в текстовом формате ASCII

4 Работа с программами

4.1 Создать программу

TpaCAD создает программы, подпрограммы и макропрограммы командой **Создать** (значок ) из меню Применение.

Обычно выбирается создание **Программы**. При возникновении необходимости определения только один раз совокупности операций для неоднократного использования в программе создается **Подпрограмма**.
Возможность создания программы типа **Макропрограммы** активизирована только если уровень доступа равен уровню доступа производителя.
В зависимости от конфигурации команда **Создать** может напрямую запустить создание **Программы**, не открывая окно выбора.
Тип детали можно изменить непосредственно на этапе редактирования.

На этапе задания новая программа инициализируется при использовании программы прототипа по умолчанию (PIECE.TCN, в папке TPACADCFG\CUSTOM). Если программа по умолчанию отсутствует, создается программа размерами 500*500*20 мм. Программу-прототип можно разделить по типу: см. параграф Персонализировать файл "прототип"

Выбрать кнопку  окна, запускается контекстуальная справка текущего окна.

TpaCAD может создавать и/или открывать только одну программу за раз. Однако можно запускать несколько копий приложения, доходя до макс. 4.

Создать программу по модели

Чтобы создать новую программу, выбрав модель, выберите команду **Новое по модели...**  из меню Применение, выберите уже заархивированную программу в диалоговом окне и подтвердите.
Модель представляет собой прототип программы, который уже содержит желаемые настройки для инициализации новой программы. В диалоговом окне отображается содержимое папки по умолчанию модели.

Вы можете подтвердить создание программы без использования модели, щелкая кнопкой **Открыть без модели**.
В этом случае команда продолжается так же, как в предыдущем случае, используя программу прототипов по умолчанию.
Команда **Новое по модели...** создает только одну **Программу**.

4.2 Открыть и импортировать программу

TpaCAD позволяет открывать программы, подпрограммы и макропрограммы командой **Файл->Открыть** (значок ) из меню Применение.
TpaCAD записывает два своих типа файла:

- расширение TCN: умолчание для программ и подпрограмм (тип файла: **файлы TpaCAD)**
- расширение TMCR: умолчание для макропрограммы (тип файла: **TpaCAD Макро**).

Команда открывает пользовательское окно *Управления ресурсами*:

- два указанных типа приводятся в списке **Тип файла.** Тип макропрограммы приводится только если уровень доступа позволяет его открытие.
- выбрать тип "Все файлы (*.*)" с тем, чтобы не устанавливать фильтры отображения. При таком выборе типа можно открыть только программы, напрямую распознаваемые приложением TpaCAD.

Для программы или подпрограммы не обязательно задавать расширение, соответствующее выбранному типу. Однако может помочь для немедленного узнавания программ-деталей.
Если выбранная в списке программа распознается как программа-деталь, можно отобразить ее предварительное графическое представление **(Предварительный просмотр).**
В окне отображаются также комментарий и размеры.

Если выбранная программа имеет очень большие размеры, предварительный просмотр временно отключается для предупреждения слишком длительного графического отображения. Если вам все-таки требуется предварительный просмотр программы, необходимо подключить пункт **Предварительный просмотр**. Размер, свыше которого программа должна считаться "большой", определяется в ***Индивидуализировать ->Среда->Сохранение***.

Рядом с зоной, оставленной для комментария, могут представляться два изображения, указывающие, имеет ли выбранная программа защиту на считывании и/или на записи.
Выбрать пункт **Открыть копию** для открытия файла в виде копии: программа загружается из выбранного файла, к имени добавляется "(2)" и для целей сохранения она рассматривается как новая программа.

Импорт программы из внешнего формата

В этом окне, если они сконфигурированы производителем станка, можно выбрать типы форматов, отличные от формата TpaCAD. Процедуры импорта выполняются внешними компонентами, соединенными с TpaCAD.
Вначале выбрать требуемый тип из списка, приведенного в этом окне (например, "Файлы DXf (*.dxf)"), а затем - файл, соответствующий выбранному типу: подтверждение открытия напрямую запускает преобразование файла в формат TpaCAD.
Выполнение просмотра графического представления для данного файла зависит от настроек конфигурации, определенных производителем станка. Программа, открытая с преобразованием формата, считается как программа, созданная командой Новая программа.

При открытии файла с преобразованием из внешнего формата в программе могут активизироваться глобальные инструменты исходя из определенного в **Индивидуализировать->Среда->Импорт формата**.
В частности, активизация:
- может быть рабочей автоматически, т. е. без какого-либо запроса подтверждения
- может зависеть от запроса оператору сообщением "Применить заданные автоматические настройки?";

Автоматические задания обуславливают применение особых инструментов к программе:
- общие задания со считыванием из файла прототипа PIECE.TCN (архивированного в папке: TPACADCFG\CUSTOM); в частности: режим выполнения, переменные <o> и <v>, клиентские разделы.
- применение технологии к открытым профилям (которые не начинаются с операции setup) или профилям, заданным с геометрической setup
- применение технологии к точечным операциям, заданным с геометрическим кодом точки
- сокращение во фрагментации линейных отрезков профиля с оценкой угла совокупного сокращения
- фрагментация и линеаризация дуг профиля
- автоматическое соединение профилей с проверкой их геометрической непрерывности.

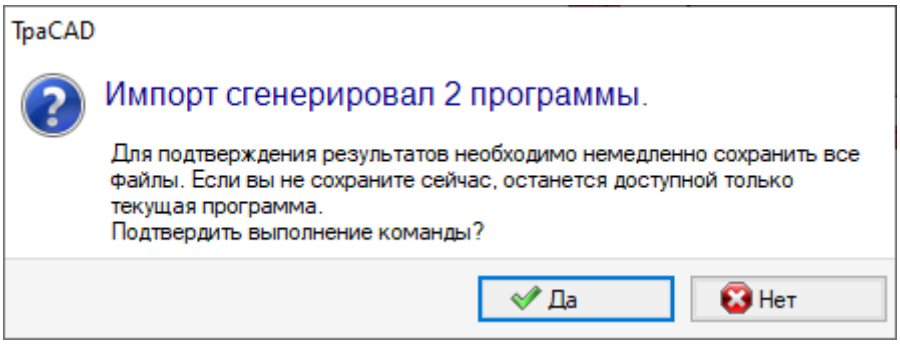
Перед подтверждением открытия файла во внешнем формате, исходя из конфигурации TpaCAD, может быть возможно:
- задать аргументы, которые могут использоваться для преобразования: в окне открытия файла отображается поле для задания аргументов. Поле инициализируется аргументами по умолчанию, как они заданы производителем станка. Важно, чтобы пользователь, задающий аргументы преобразования, знал их значение.
- запросить индивидуализацию отдельного преобразования, как она управляется модулем преобразования (например, если программа имеет формат DXF, можно указать, какие уровни должны преобразовываться, а какие должны быть исключены из преобразования)

ВНИМАНИЕ: если все шесть реальных граней отключены, импорт программы присваивает тип подпрограммы.

Импорт с созданием нескольких файлов

Конкретный случай может соответствовать импорту с созданием нескольких файлов TCN.
Например, ситуация может соответствовать созданию второй программы с обработками грани 2 (грань ниже).
В этом случае все происходит, как указано выше, и предупреждение указывает на возникновение конкретной ситуации.

Рисунок соответствует случаю создания 2 созданных программ:



Продолжите подтверждать немедленное сохранение, чтобы сохранить импорт обеих программ.
В противном случае будет доступна только первая импортированная программа, а вторая (или более поздняя) не будет восстановлена.
В нашем примере два файла сохраняются путем выбора местоположения и имени первого файла. Второй (или более поздние) сохраняется путем добавления прогрессивного суффикса к назначенному имени.

Импорт программы в формате TpaEdi32

Программа, написанная при помощи TpaEdi32, может напрямую загружаться и совершенствоваться. Распознавание формата программы выполняется автоматически с выбором одного из типов файла TpaCAD (*.TCN, *.TMCR, *.*). При открытии программы, написанной в TpaEdi32, можно автоматически задать режимы выполнения при использовании настроек, определенных в файле прототипа PIECE.TCN: для выполнения заданий достаточно подтвердить при открытии окна запроса.
Напоминаем, что программа, созданная при помощи TpaCAD, может читаться TpaEdi32 только при сохранении в формате TpaEdi32.

Импорт программы в формате EdiCad

Программа, написанная при помощи Edicad, может напрямую загружаться и совершенствоваться: распознавание формата программы выполняется автоматически с выбором одного из типов файла TpaCAD (*.TCN, *.TMCR, *.*).
Обратное неверно, т. е. программа, созданная при помощи TpaCAD, не может считываться EdiCad.
Для чтения макропрограммы, написанной при помощи EdiCad, необходимо сохранить ее в формате ASCII непосредственно в Edicad.
При открытии программы, написанной в Edicad, можно автоматически задать различные настройки, используя настройки, определенные в файле прототипа PIECE.TCN. Для выполнения заданий достаточно подтвердить при открытии окна запроса. Помимо режима выполнения теперь можно автоматически задать клиентские разделы (которые содержат Специальные уставки, Дополнительную информацию, Раздел привязок, Настройки оптимизации).

Информация, восстанавливаемая при импорте:

- Три сдвига восстанавливаются в первых трех переменных "o"
- Переменные цикла восстанавливаются в переменных "v"
- Информация о фиктивной грани: задание подобной грани восстанавливается в направлении, заданном для оси z
- В параметрическом программировании , (запятая) каждый раз заменяется ; (точкой с запятой)
- При считывании операций программы: параметр, относящийся к комментарию, восстанавливается как комментарий к операции (например, IF, FOR,..)
- При считывании операций программы некоторые коды операций присваиваются заново с другими эквивалентами
- Программы, заданные в EdiCad как подциклы, сохраняются как макропрограммы

Информация, утрачиваемая при импорте:
- Задание поля последовательности на одной грани
- Задание профильной детали

Информация, не решаемая при импорте:
- Технологические функции параметрического программирования, касающиеся многоцентровых головок
- Синтаксис вызова подциклов с символом "*" для адресования вызова подпрограммы в директорию TPACADCFG\SUB больше не поддерживается.

Открытие программы-детали, созданной во внешней среде

Программа, распознанная как программа-деталь, может идентифицироваться как не созданная в среде TpaCAD. Это обычная ситуация для программ, генерированных импортом или программой управления.

В таких случаях открытие программы может интегрировать программирование
- общей информации о программе (переменные, клиентские разделы) и/или
- операций с настройками по умолчанию.

Порядок работы определяется заданным в **Конфигурации TpaCAD**.

Запуск TpaCAD из Управления ресурсами

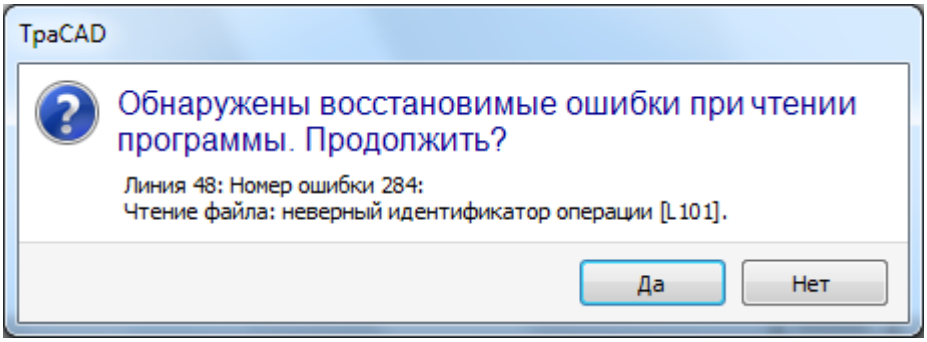
Приложение TpaCAD можно запустить также напрямую из Управления ресурсами, запрашивая открытие файла с расширением TCN.
В этом случае при запуске не реализуются функции Выбора оборудования или Рабочей среды даже при их подключении.

Сигналы при открытии программы

При открытии программы, распознаваемой форматом TpaCAD (включая файлы, зарегистрированные приложениями Edicad и TpaEdi32), могут возникать ситуации формальной ошибки. Это, в основном, ситуации несоблюдения необходимого для программы TpaCAD синтаксиса, которые относятся, в основном к файлам, генерированным внешним процессом.
Описание сигналов ошибки см. в главе Сообщения об ошибке.
Существует два случая ошибки:
1. неисправимые ошибки
2. ошибки, считающиеся исправимыми.

Первый случай всегда связан с отменой команды открытия программы.
Могут наблюдаться следующие ситуации:
- ошибка распределения памяти: это серьезная ситуация, указывающая на завершение свободной памяти в системе
- файл, зарегистрированный с неоднозначным формализмом: распознаются операции, архивированные как в формате ASCII, так и в исходном формате. Без сомнения, файл не был создан CAD TPA.
- имеются неуправляемые разделы без указания строки закрытия раздела. Без сомнения, файл не был создан CAD TPA.

Второй случай можно решить за счет подключения автоматической процедуры устранения ошибки. На рисунке показан возможный сигнал.



В окне отображается первая выявленная ситуация ошибки и дается предупреждение, что были выявлены только исправимые ошибки. При подключении процедуры восстановления в конце считывания пользователя предупреждают об успехе операции. Для подтверждения программы необходимо сохранить ее. Программа отмечается как измененная и при закрытии будет отображаться запрос, сохранять ли ее.

Эта процедура восстановления подключается только в TpaCAD. Считывание этой программы не удается при ее выполнении на машине.
Если программа была генерирована внешним приложением, открытие с процедурой восстановления может указывать на постоянные ошибки в создании файлов TCN, которые, в любом случае, рекомендуется исправить, т. к. процедура восстановления может привести к автоматическому удалению значимых частей программы именно вследствие неправильного исходного синтаксиса.

Формат регистрации программы-детали

Программа, распознанная как программа-деталь, независимо от расширения файла и распознанного типа, является файлом текстового формата, зарегистрированного с кодировкой ANSI или Unicode.
Обе кодировки всегда распознаются.
Кодировка программы используется, например, в программировании описания программы, в создании текста или в имени, которое может иметь подпрограмма.

Кодировка ANSI основана на оригинальной базовой схеме кодировки из 95 печатных символов:

```
 !"#$%&'()*+,-./0123456789:;<=>?
@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_
`abcdefghijklmnopqrstuvwxyz{|}~
```

она включает также еще 255 символов за счет использования особой системы кодовых страниц, соответствующих локальным системным настройкам. Такая система позволяет представлять символы, характерные для определенного языка или группы языков, с использованием при этом ограниченного числа кодов. Результат получается путем задания различных представлений одному коду при изменении заданной в системе кодировки. Если мы рассмотрим, например, один из символов с ударением исходя из локальной настройки, действительной для Западной Европы (**а**, **о**,..) с регистрацией символа в файле формата ANSI, этот же файл, открытый при задании кириллицы или иврита, будет представлять другой символ именно вследствие распознавания другой активной страницы кодов.

Однако этой кодировки недостаточно при работе, например, с азиатскими языками, которые, как правило, имеют намного больше символов по сравнению с другими языками, которые обычно можно закодировать при помощи 255 символов. Решением является управление файлами с кодировкой Unicode:
- можно читать файлы, зарегистрированные с обеими кодировками
- при создании новой программы автоматически задается кодировка по умолчанию Unicode
- при сохранении программы можно выбрать кодировку для использования.

Дополнительную информацию см. в параграфе Сохранить программу.

4.3 Открыть программу из меню Файл

Меню Файл отображает список ранее открывавшихся файлов и содержит максимум 10 пунктов. Двойной щелчок на имени в перечне напрямую открывает программу.
Щелкнуть имя правой кнопкой мыши для открытия простого меню:

Открыть путь к файлу: открывает диалоговое окно *Открыть* непосредственно в папке, где архивирован файл

Открыть копию: для открытия файла в виде копии. Программа загружается из выбранного файла, к имени добавляется "(2)" и для целей сохранения он рассматривается как новая программа

Удалить из списка: удаляет пункт из списка.

4.4 Буксировка (drag & drop)

Можно открыть программу буксируя файл, например, из Управления ресурсами, в рабочую зону TpaCAD. Если файл не распознается в формате TpaCAD, проверяется, возможно ли преобразование при помощи сконфигурированных модулей импорта. Если проверка дает положительный результат, выполняется преобразование с последующим открытием файла в порядке, описанном в предыдущих параграфах.

Буксировка игнорируется, если выполняется процедура, например, открыто окно и ожидается завершение команды.

4.5 Распечатать программу

TpaCAD печатает программу, активную в виде, представленном в графической зоне, командой **Файл->Печать->Печать графического изображения** (значок ) из меню Применение.

Соблюдаются активные зум и сдвиг, фильтры изображения и специальные виды и также все активные графические элементы (курсор, сетка, крайние точки, стрелки направления профилей и т. д.).
Печать программы может потребовать обновления графики, обуславливаемого особыми настройками сложных операций, которые могут потребовать индивидуализации печати. В этом случае вначале обновляется графическое представление, а затем запрашивается подтверждение для запуска процедуры печати. По завершении выполнения этой команды графика приводится в исходное состояние.
Индивидуализации печати соответствуют программным строкам со специальными условиями, которые могут, например, соответствовать добавлениям надписей и/или элементов проставления размеров, пунктира, вспомогательных контуров. В любом случае, речь идет об аспектах, касающихся программирования текстов макропрограммы.

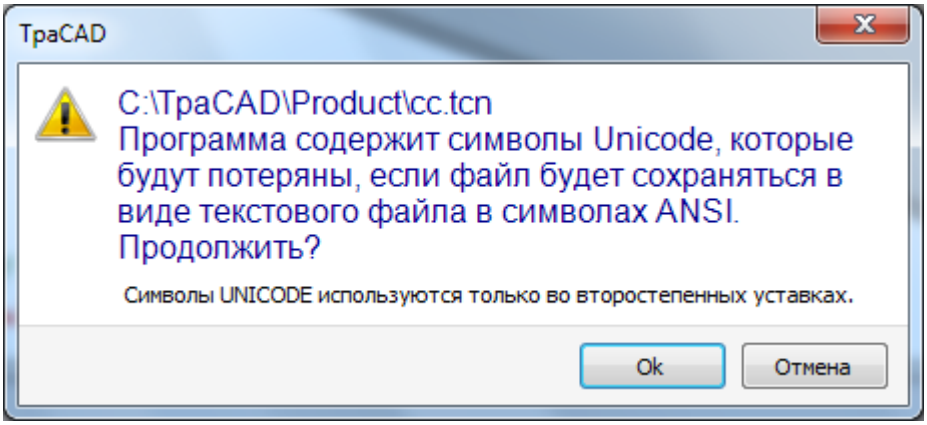
4.6 Сохранить программу

TpaCAD записывает в память текущую программу командой **Файл->Сохранить** (значок ) из меню Применение. Если редактируемая программа новая или выбирается из меню команда **Файл->Сохранить как**, открывается окно для задания имени файла и его размещения в архиве. Задаваемое для файла расширение может выбираться из предлагаемых или же определяться пользователем. Расширения по умолчанию: TCN для программ и подпрограмм, TMCR для макропрограмм. При архивировании макропрограммы предлагается только расширение TMCR, обязательное для правильного распознавания. Рекомендуется не задавать расширение TMCR (по умолчанию для макропрограмм) программам и подпрограммам.

Если программа оказывается изменена по сравнению с последней записью в память, эта ситуация сигнализируется сообщением и запрашивается подтверждение для сохранения.
При выполнении команды **Файл -> Сохранить как** при выборе существующего маршрута программы в формате TpaCAD архивирование не выполняется, если перезаписываемый файл защищен от записи с уровнем доступа выше текущего (в окне открытия появится изображение: ).
Можно выбрать кодировку для использования при сохранении из следующих вариантов:

- ANSI
- Unicode.

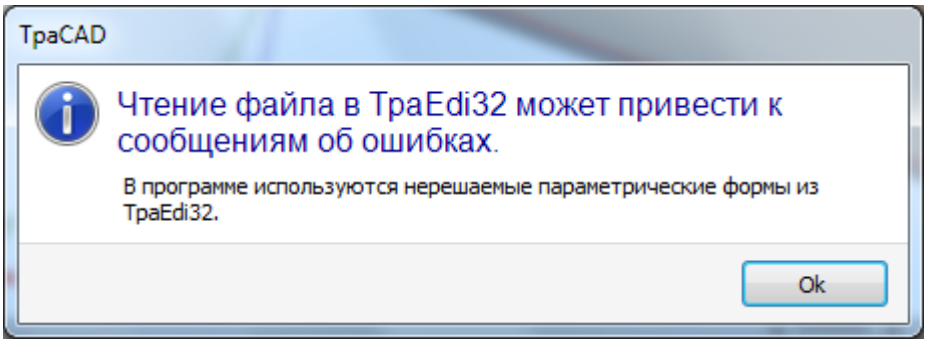
Если выбирается формат ANSI и программа использует символы Unicode, сообщение сигнализирует потерю информации



Этот сигнал, как видно на рисунке, указывает, что символы Unicode используются во второстепенных уставках, например, описание программы, переменных или операций. При потере при регистрации второстепенной уставки меняет представление надписи на экране, но не меняет интерпретацию программы.
Невторостепенная уставка, например, программирование переменной типа строки: потеря информации меняет интерпретацию программы, вызывая ситуацию ошибки.

При конфигурации производителем станка можно выбрать сохранение в формате, совместимом с TpaEdi32. В этом случае формат файла - только ANSI. Совместимость должна пониматься как возможность чтения

программы при помощи TpaEdi32 без сообщений о несовместимости версии. Возможность фактического толкования программы зависит от использованных операций и параметрического программирования, которые не должны содержать новых элементов, внесенных в TpaCAD. По завершении процедуры сохранения отображается сообщение, которое может также напрямую сигнализировать явную несовместимость интерпретации программы в TpaEdi32, как показано на рисунке



После процедуры архивирования могут быть активированы другие процедуры, конфигурируемые производителем станка.
В частности:
- преобразование программы во внешний формат.
- оптимизации программы

Эти процедуры не активизированы в случае сохранения макропрограмм.
Иногда для активизации этих процедур может потребоваться достаточно много времени. При щелчке внутри графической зоны пользователь, специальным окном с сообщением, предупреждается, что архивирование программы еще не завершено.

Имена, которые нельзя использовать

Некоторые имена использовать нельзя, потому что они зарезервированы операционной системой.
Вот список:

CON, PRN, AUX, NUL, COM1, COM2, COM3, COM4, COM5, COM6, COM7, COM8, COM9, LPT1, LPT2, LPT3, LPT4, LPT5, LPT6, LPT7, LPT8, LPT9

Приведенные выше имена нельзя использовать с какими-либо расширениями файлов: например, «con.tcn», «con.tmcr» или «con».

4.7 Оптимизировать программу

TpaCAD оптимизирует текущую программу командой ***Файл->Оптимизировать*** (значок ) из меню Применение.
Оптимизация программы выполняется внешним компонентом, соединенным с TpaCAD, и по критериям, определенным производителем станка.
Если программа была изменена или если это новая программа, перед оптимизацией выполняется архивирование. Параметры, используемые для оптимизации, - это параметры, заданные в программе на момент выбора команды: режим выполнения, исключения, размеры, переменные…
Уже говорилось, что оптимизацию программы можно выполнить и после сохранения.
Однако оптимизация, выполненная по прямому запросу, как правило, является более полной и имеет возможность архивирования и/или записи в память отчетов.

4.8 Напечатать ярлык программы

TpaCAD напечатает ярлык, связанный с текущей программой, используя команду ***Файл -> Напечатать ярлык программы*** (значок ) из меню Применение.

Ярлык создается в соответствии с форматом, определенным при настройке TpaCAD.

Первое окно позволяет Вам выбрать, печатать ли ярлык или сохранить на файл.
В случае печати второе окно позволяет подтвердить или выбрать принтер, который будет использоваться для печати.

4.9 Экспортировать программу

TpaCAD экспортирует активную программу в одном из форматов, сконфигурированных производителем станка. Процедуры экспорта выполняются внешними компонентами, соединенными с TpaCAD.
Команда подключается из меню **Файл->Экспорт** из меню Приложение. Отображается меню для выбора типа преобразования. Примеры возможного выбора:
- Файл Edicad
- Файл DXF
- Файл ISO.

При выборе преобразования при необходимости программа сохраняется и может открываться окно для задания имени файла и его помещения в архив. Параметры, используемые для экспорта, - это параметры, заданные в программе на момент выбора команды: режим выполнения, исключения, размеры, переменные.

4.10 Преобразовать архив программ

Выбрать команду ***Файл->Преобразовать архив программ*** (значок ) в меню Применение.

Эта команда запускает открытие списка программ, указывая один из трех различных типов формата, предусмотренных для считывания, и последующее архивирование, указывая один из различных типов формата, предусмотренных для сохранения.
Сохранение осуществляется без выполнения оптимизатора.

Рассмотрим подробно настройки:

- **Путь чтения**: инициализируется маршрутом открытия программ, задает диск и папку для считывания программ. Поиск маршрута запускается при нажатии кнопки в этом поле, с открытием из окна поиска файлов. В частности:

Можно выполнить множественный выбор программ
типы реализованных форматов соответствуют типам, рассмотренным для команды Открытие программы
  - расширение TCN: умолчание для программ и подпрограмм (тип файла: **файлы TpaCAD**)
  - "Все файлы (*.*)" с тем, чтобы не устанавливать фильтры отображения. При таком выборе типа можно открыть только программы, напрямую распознаваемые приложением TpaCAD
  - типы, соответствующие сконфигурированным модулям импорта.
- **Формат считывания**: содержит индикацию типа формата, выбранного в окне поиска файлов;
- **Имя файла**: содержит индикацию программ, которые были выбраны в окне поиска файлов;
- **Путь для сохранения**: инициализируется маршрутом регистрации программ, задает диск и папку для сохранения программ. Поиск маршрута запускается при нажатии кнопки в этом поле, с открытием из окна поиска маршрута;
- **Формат записи**: содержит список типов формата, возможных для сохранения программ:
  - расширение TCN: сохраняет программу в формате **Файлы TpaCAD** с заданным расширением TCN
  - "Все файлы (*.*)" сохраняет программу в формате **Файлы TpaCAD**: если программы были открыты без задания формата, расширение файла не меняется, в противном случае расширение удаляется;
  - типы, соответствующие сконфигурированным модулям экспорта.
- **Программное кодирование (*.TCN)**: только для сохранения в формате **Файлы TpaCAD**, выбирая из следующих вариантов:
  - ANSI
  - Unicode.
- **Перезаписать файл с таким же именем**: выбрать для перезаписи существующих программ
- **Включить интерактивную диагностику**: выбрать для управления взаимодействием в окне при возникновении любой ситуации ошибки. В этом случае каждый раз при подаче сигнала можно запросить немедленную отмену команды.

ВНИМАНИЕ: если все шесть реальных граней отключены, импорт программы в формате TpaCAD присваивает тип подпрограммы.

В случае импорта с созданием нескольких файлов TCN, все файлы сохраняются.

При подтверждении настроек в этом окне команда запускается и выполняется до своего завершения. По завершении выполнения предупреждение указывает число обработок, завершившихся правильно или неправильно.

В зоне "Команды" можно подробно проанализировать все выполнение процедуры и при необходимости определить причину каждой отдельной ошибки.

При запросе сохранения в формате **Файлы TpaCAD** по завершении выполнения команды и при по меньшей мере одной выполненной обработке маршрут, заданный для сохранения программ *.TCN, задается как последний открытый для последующего открытия программы.

4.11 Оптимизировать архив программ

Выбрать команду ***Файл->Оптимизировать архив программ***  из меню Применение.

Эта команда позволяет запускать оптимизацию списка программ, которые должны регистрироваться в формате **Файлы TpaCAD**. Оптимизация программы выполняется внешним компонентом, соединенным с TpaCAD, и по критериям, определенным производителем станка.

Рассмотрим подробно настройки:

- **Маршрут считывания**: поле инициализируется маршрутом открытия программ. Задает диск и папку для считывания программ. Кнопкой  открывается окно поиска файла.
  В частности:
  - можно выполнить множественный выбор программ
  - типы реализованных форматов:
    - расширение TCN: умолчание для программ и подпрограмм (тип файла: **файлы TpaCAD)**
    - "Все файлы (*.*)" с тем, чтобы не устанавливать фильтры отображения. При таком выборе типа можно открыть только программы, напрямую распознаваемые TpaCAD.
- **Формат считывания**: отображает тип формата, выбранного в окне поиска файлов
- **Имя файла**: отображает индикацию программ, которые были выбраны в окне поиска файлов

- **Подключить интерактивную диагностику**: выбрать для управления взаимодействием в окне при возникновении любой ситуации ошибки. В этом случае каждый раз при подаче сигнала можно запросить немедленную отмену команды.

При подтверждении настроек в окне команда запускается и выполняется до завершения или до первой ошибки, при выборе клетки **Подключить интерактивную диагностику**.

По завершении выполнения предупреждение указывает число обработок, завершенных правильно, или число обработок, завершенных неправильно.
В зоне "Команды" можно подробно проанализировать все выполнение процедуры и при необходимости определить причину каждой отдельной ошибки.

4.12 Показать предварительный просмотр для оптимизации программы

Можно запросить отображение программы, доходящей до этапа исполнения: команду ***Предварительный просмотр для оптимизации*** можно выбрать во вкладке **Показать**. Эта команда подключена в Общем виде.
При выборе команды при необходимости программа сохраняется и может открываться окно для задания имени файла и его помещения в архив.

Предварительный просмотр для оптимизации позволяет видеть, как будет совершенствоваться программа с запросом исполнения с текущим состоянием настроек (размеры, переменные). Результат может также сильно отличаться от того, который обычно отображается на этапе редактирования, вследствие применения:

- ✓ различных параметрических заданий,
- ✓ различных логических ограничений,
- ✓ функций Многократных подготовок, Коррекции на инструмент, Фрагментации дуг.

Если при обработке программы выявляются ситуации ошибки, открывается окно со всеми диагностическими сигналами и команда *Предварительный просмотр* отменяется.

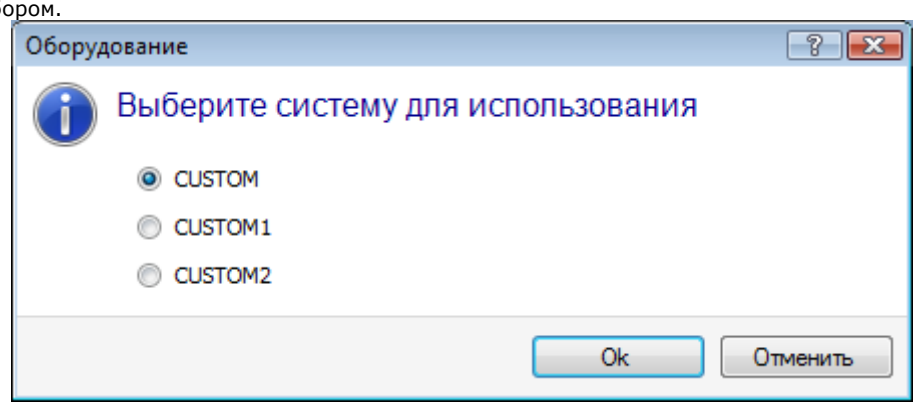
Открывается окно независимого отображения, в котором могут иметься специальные команды и функции. Закрытие окна определяет закрытие команды.

4.13 Установка

Установка - это совокупность одного или нескольких станков. Станки (или модули), в свою очередь, состоят из групп, которые состоят из подгрупп и устройств. Как правило, установка уникальна, поэтому не требуется возможность ее изменить. Иногда может потребоваться инсталлировать различные конфигурации установок.

окно выбора установки может вызываться из меню **->Установка** или же выбор может запрашиваться при запуске TpaCAD. Это дополнительная конфигурируемая команда, активная только если нет открытой программы.

В этом окне в алфавитном порядке приводится перечень сконфигурированных установок с выделенным текущим выбором.



Выбрать имя установки, которую вы желаете использовать, и подтвердить кнопкой **[ОК].**
Подчеркиваем значение, которое имеет изменение Рабочей установки: это, практически, эквивалентно работе с инсталляциями, выполненными на двух компьютерах и работающими на двух различных установках.
Работа на нескольких установках требует персонализированной инсталляции среды TpaCAD и, в целом, - всей инсталлированной программной среды TPA. Выбор в TpaCAD установки, отличной от выбранной по умолчанию, не меняет работу среды за пределами TPA.

4.14 Рабочая среда

TpaCAD может управлять второй рабочей средой, которая называется “Чертеж”, в качестве альтернативы обычно используемой среде, называемой “Станок". Переключение на среду “Чертеж” может определяться уровнем доступа.
При наличии команда переключения между этими двумя средами помещается на главную строку и подключается при закрытой программе и проверенном уровне, как задано в конфигурации TpaCAD:

 указывает, что активна среда “Чертеж”

 указывает, что активна среда “Станок”.

Как опция выбор может запрашиваться при запуске TpaCAD. Это дополнительная конфигурируемая команда, активная только если нет открытой программы.
Соразмерно с запрошенным уровнем доступа окно выбора рабочей среды, в любом случае, предлагается при запуске следующей копии TpaCAD.

Управление средой “Чертеж” может отвечать определенным потребностям, например:

- очень специфичная среда программирования, для подключения геометрий и/или сечений детали и/или состава меню
- среда разработки подпрограмм и/или макропрограмм, имеющая значительные отличия.

Реализация среды “Чертеж” может также выполняться чисто из удобства для дифференциации между обычной рабочей средой использования TpaCAD с упрощенными меню и более насыщенной и мощной, но требующей больше опыта в использовании программы.

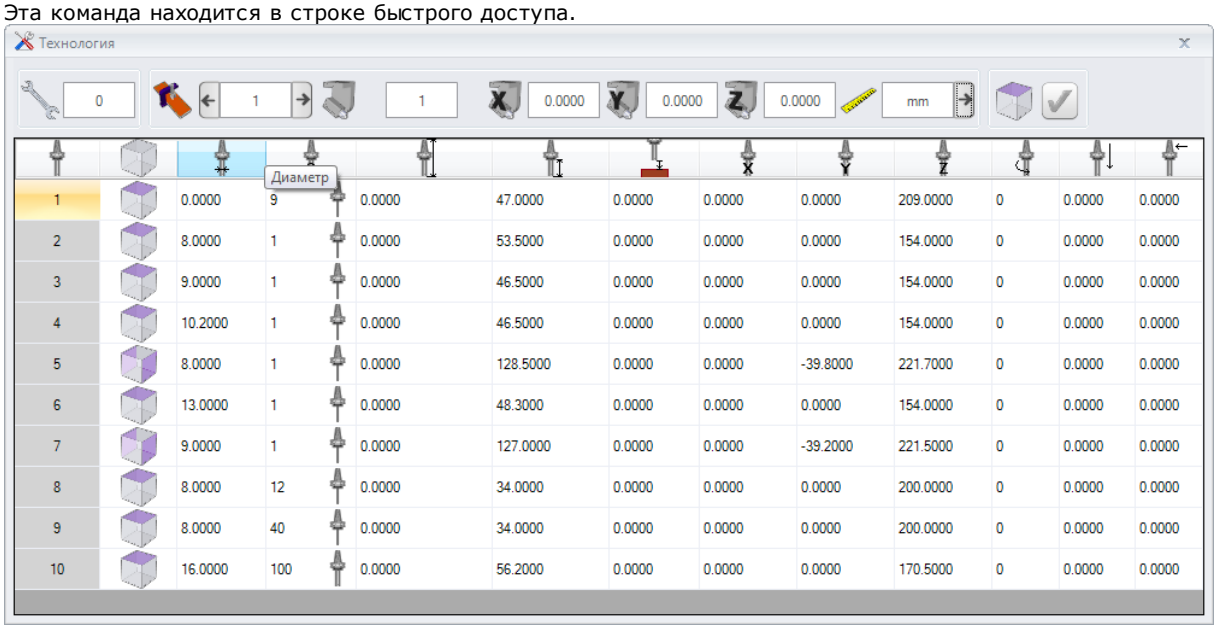
4.15 Несколько копий TpaCAD

Как уже отмечалось, TpaCAD может создавать и/или открывать только одну программу за раз, но можно запускать несколько копий приложения, максимум 4.
Для каждой копии можно выбрать Установку и/или Рабочую среду, если и как предусматривается конфигурацией TpaCAD.
В частности, можно выполнять операции Копирования/Вклеивания операций между разными копиями.

При запуске нескольких копий на одной Установке только первая имеет право сохранять изменения конфигурации и индивидуализаций TpaCAD.
При использовании непервичной копии на этапе закрытия приложения сообщение сообщит о том, что не будет сохранено ни одно изменения настройки и/или индивидуализации TpaCAD. Точно таким же образом открытие окон Конфигурации предупреждает о невозможности выполнить изменения, учитывая пределы доступа, заданные для всех копий, признанных вторичными.

4.16 Таблица инструментов

TpaCAD работает обычным способом в технологическом контексте установки. Этот программный продукт напрямую сопрягается с одним или более станками, для которых он знает задания, относящиеся к группам и рабочим инструментам.
Технологическое задание инструментов имеет первостепенное значение для выполняемых в программе операций, обычно можно отображать таблицу доступных для операций инструментов.
Эта команда находится в строке быстрого доступа.



1	0.0000	9	0.0000	47.0000	0.0000	0.0000	0.0000	209.0000	0	0.0000	0.0000
2	8.0000	1	0.0000	53.5000	0.0000	0.0000	0.0000	154.0000	0	0.0000	0.0000
3	9.0000	1	0.0000	46.5000	0.0000	0.0000	0.0000	154.0000	0	0.0000	0.0000
4	10.2000	1	0.0000	46.5000	0.0000	0.0000	0.0000	154.0000	0	0.0000	0.0000
5	8.0000	1	0.0000	128.5000	0.0000	0.0000	-39.8000	221.7000	0	0.0000	0.0000
6	13.0000	1	0.0000	48.3000	0.0000	0.0000	0.0000	154.0000	0	0.0000	0.0000
7	9.0000	1	0.0000	127.0000	0.0000	0.0000	-39.2000	221.5000	0	0.0000	0.0000
8	8.0000	12	0.0000	34.0000	0.0000	0.0000	0.0000	200.0000	0	0.0000	0.0000
9	8.0000	40	0.0000	34.0000	0.0000	0.0000	0.0000	200.0000	0	0.0000	0.0000
10	16.0000	100	0.0000	56.2000	0.0000	0.0000	0.0000	170.5000	0	0.0000	0.0000

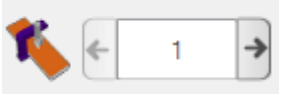
Если в Конфигурации TpaCAD включено управление универсальным инструментом, на первой странице отображается список универсальных инструментов, а поля станка и группы имеют значение 0.

Окно отображения может меняться в зависимости от конфигурации, заданной для TpaCAD. На рисунке показано стандартное представление оснащения установки: характеристики конфигурации инструментов группы приводятся в таблице с возможностью выбора групп каждого станка установки. Минимальная структура технологической конфигурации предусматривает станок только с одной группой.



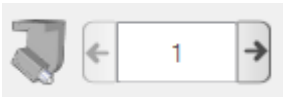
Оснастка

Содержит номер текущего оснащения. Эта зона отображается только если оснащение может выбираться в программе. Это поле неизменяемо и содержит целое числовое значение больше или равное 0. Оснащение - это фотография того, как огранизованы группы станка: это структура станка. Управление несколькими оснащениями типично для установки с только одним станком.



Станок

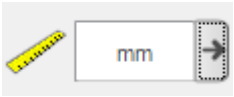
Поле выбора станка: позволяет прокручивать станки, сконфигурированные в установке. Если в Конфигурации TpaCAD включено управление универсальным инструментом, поле Станка может принимать значение 0.



Группа

Поле выбора группы на станке: позволяет прокручивать группы, сконфигурированные на выбранном станке. Если в Конфигурации TpaCAD включено управление универсальным инструментом, поле Группы может принимать значение 0.

Рядом отображаются корректоры X,Y,Z группы: 



Единица измерения отображения

Поле выбора единицы представления параметрических данных. Выбор осуществляется между [мм] и [дюймами]. При открытии этого окна предлагается единица измерения активированной программы.
В таблице приводится список инструментов, оснащенных в зависимости от активного выбора (оснащение, станок, группа). В каждой строке приводится значимая информация об инструменте: рабочая грань, диаметр, тип, длина, корректоры, частота вращения, скорость смещения. Порядок представления может упорядочиваться для всех столбцов.
окно Технология может отображаться и при вводе операции нажатием кнопки  в пункте **Инструмент**. Закрытие окна двойным щелчком на строке таблицы инструментов возвращает текущий выбор в технологических полях станка (станок, группа, инструмент).

окно Технологии может отображаться также в поддержку параметрического программирования. Закрытие окна двойным щелчком на одной клетке таблицы инструментов дополняет программируемое поле выбором технологической функции, соответствующей выбранным технологическим полям (станок, группа, инструмент, тип информации).

4.17 Информация о внешних компонентах, соединенных с TpaCAD

Как уже говорилось, некоторые сложные связанные с программами процедуры выполняются процедурами, запускаемыми внешними компонентами, соединенными с TpaCAD. Это действительно для: импорта из внешнего формата, экспорта во внешний формат.
Выполнение внешних процедур блокирует нормальное выполнение TpaCAD до завершения выполняемой процедуры.
Через несколько секунд после запуска процедуры, например, преобразования формата, если процедура не завершена, отображается окно с указанием ее состояния.

и может выполнять ее принудительное завершение. В этом случае последующее окно предупреждает о необходимости уделять ВНИМАНИЕ подобной операции.

Запрос на завершение процедуры может подаваться, если вы считаете, что выполняемая программа больше не реагирует, однако это всегда экстремальное решение. С учетом размеров выполняемой программы рекомендуется подождать несколько секунд перед принудительным завершением операции, которой требуется некоторое время для самостоятельного завершения. При выявлении реальной ситуации аномального функционирования прервать процедуру и сообщить об этой ситуации поставщику системы.

5 Порядок задания графического представления

5.1 Индивидуализировать виды

Команды для подключения или отключения отображения визуальных элементов в зоне графического представления панели, частично доступны в группе настройки в Индивидуализировать->Виды->Индивидуализировать виды. Ниже рассмотрим выборы в меню вкладки **Показать**.



Направление профилей: подключает или отключает отображение стрелок направления на отрезках профиля. Отображение применяется, если исключается отображение габаритных размеров диаметра инструмента. Для профилей, отмеченных конструкцией, геометрической или выемки, применение задано в ***Индивидуализировать->Виды->Индивидуализировать графическое представление***.



Точки на профилях: подключает или отключает отображение точек (небольшие кружочки) на концах отрезков профиля.Отображение применяется, если исключается отображение габаритных размеров диаметра инструмента. Для профилей, отмеченных конструкцией, геометрической или выемки, применение задано в ***Индивидуализировать->Виды->Индивидуализировать графическое представление***.



Координаты операции: подключает или отключает представление добавленных элементов и координат для текущей операции. Исключаются операции конструкции или сложные операции (подпрограммы, макропрограммы). Например, если текущая операция представляет собой дугу, отображаются координаты крайних точек дуги, центра и начального радиуса в виде линейного отрезка между точкой начала дуги и центром.



Габариты трехмерного графического изображения: подключает или отключает блоком отображение габаритов в трехмерном изображении (вертикальные размеры), согласно соответствующим настройкам в окне, открытом из ***Индивидуализировать->Виды->Индивидуализировать графическое представление***. Эта опция значима только в трехмерном изображении. Для профилей, отмеченных конструкцией, геометрической или выемки, применение задано в ***Индивидуализировать->Виды->Индивидуализировать графическое представление***.
Эта кнопка применяет заданные подключения меню, которое реализовано на этой кнопке:



Показать габарит точек и подготовки в трехмерном изображении



Показать габарит профилей в трехмерном изображении

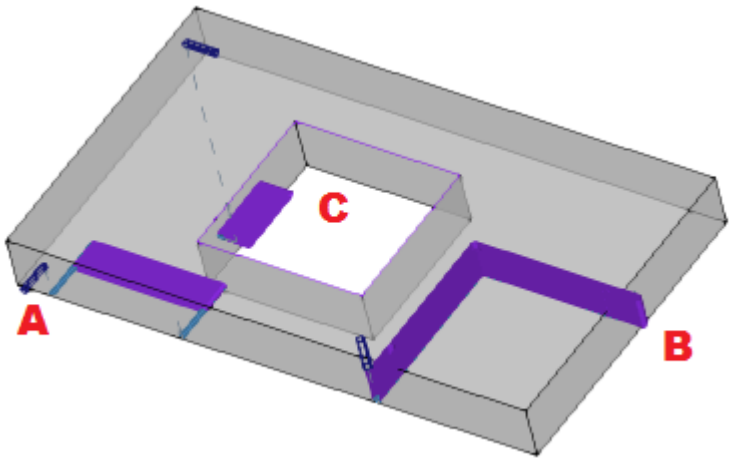
Для точечных операций, например, сверления, габарит трехмерного изображения приводится только в случае глубины в детали (не в воздухе) и соответствует представлению цилиндра диаметра, равного диаметру, заданному или определяется технологией.
Для профилей фактический порядок отображения габаритов трехмерного изображения регулируется дополнительной группой опций. Рассмотрим подробно:



Ограничить габариты трехмерного изображения: подключает или отключает оценку определенных условий ограничения (накладка) габаритов трехмерного изображения профилей. Если эта опция не выбрана, представляется габарит трехмерного изображения, соответствующий длине инструмента; единственная дополнительная оценка касается запрограммированной глубины грани:
• если в воздухе: отображает только отрезок пунктиром, без добавленных габаритов (фактическое представление определяется соответствующим заданием в ***Индивидуализировать->Виды->Индивидуализировать графическое представление***)
• в противном случае: отображает габариты, применяя технологические данные инструмента (длина и диаметр).
На рисунке ниже представлена деталь с заданными тремя профилями:

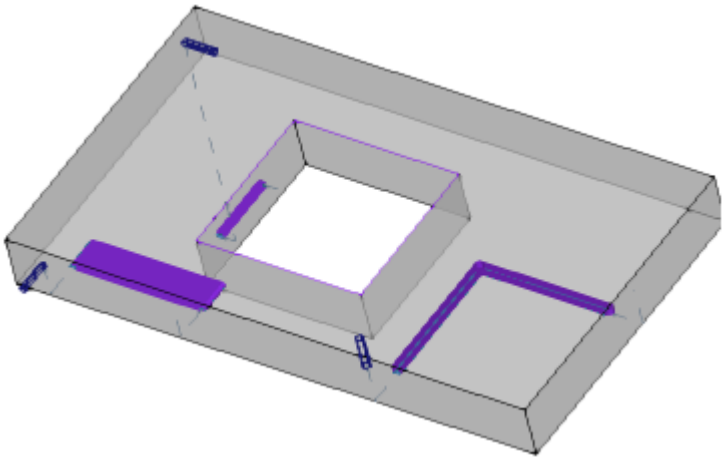
A. Ориентированный профиль, запрограммированный на грани 1, врезающийся на боковой грани с ориентированным по горизонтали инструментом. Профиль начинается и заканчивается за пределами плоскости XY грани 1 и на координате в воздухе (всегда относительно грани 1): отрезки с Z в воздухе представлены пунктиром, а отрезки только за пределами зоны XY грани представлены непрерывной линией
B. Вертикальный профиль, запрограммированный на грани 1. Профиль начинается и заканчивается за пределами плоскости XY грани 1, но с координатой не в воздухе: весь профиль представлен с примененными габаритами;
C. Вертикальный профиль, запрограммированный на фиктивной грани (напр., 7). Профиль начинается и заканчивается за пределами плоскости XY грани 7 и на координате в

воздухе: начальный и конечный отрезки грани программируются таким образом, чтобы предупредить столкновение с деталью. Только отрезки с Z грани в воздухе представлены пунктиром.

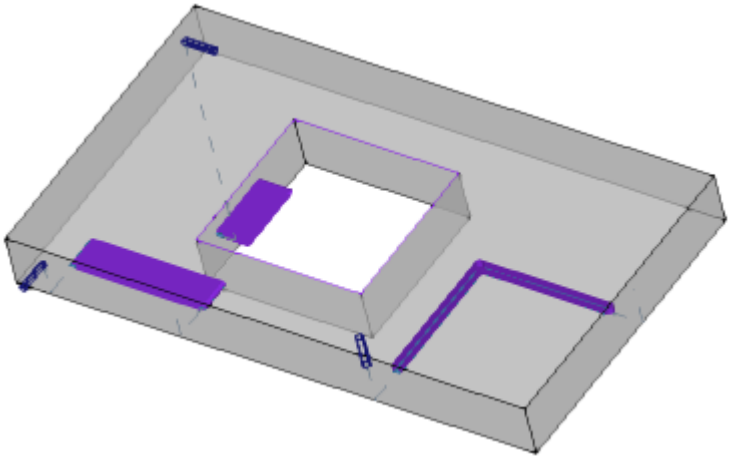


При выборе этой опции порядок оценки более четкий:
• прежде всего, для каждого отрезка профиля оценивается запрограммированная глубина, только применительно к полезной площади XY грани (ограниченная длиной и высотой грани): отрезки, запрограммированные с Z в воздухе или за пределами зоны XY грани, считаются в воздухе.
• затем выполняется оценка выбора, сделанного в группе трех опций:

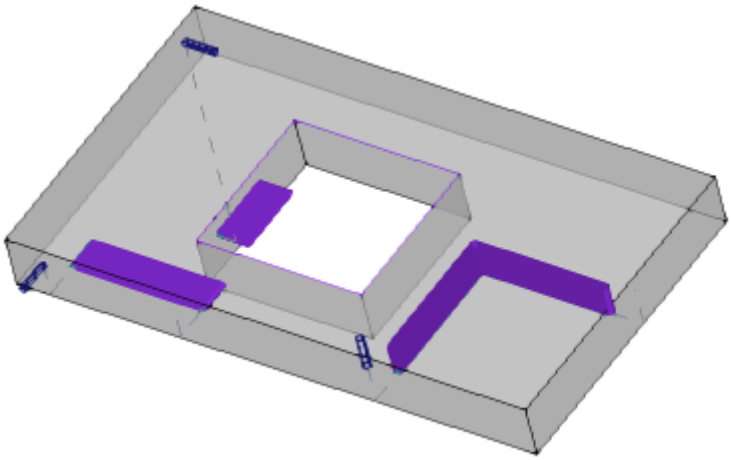
 **Ограничить габарит плоскостью грани:** представляет габариты только для отрезков, считающихся **не в воздухе** (см. предыдущий пункт), и с расширением вертикального габарита, ограниченным плоскостью XY грани. На рисунке показано, как отрезки в воздухе теперь находятся за пределами зоны XY грани



 **Ограничить габарит деталью:** представляет габариты только для отрезков, считающихся **не в воздухе** (см. предыдущий пункт) и с расширением вертикального габарита, ограниченного пересечением с деталью (исходный параллелепипед) и, в любом случае, - не больше длины инструмента. В случае закругленной грани или поверхности, вертикальный габарит интерпретирует как: **Ограничить габарит плоскостью грани.**



Ограничить габарит длиной инструмента: представляет габариты только для отрезков, считающихся **не в воздухе** (см. предыдущий пункт), и с расширением вертикального габарита, равного длине инструмента. На рисунке показан один габарит для всех отрезков не в воздухе, соответствующий длине инструмента каждого профиля



Габарит профилей: подключает или отключает отображение габарита диаметра инструмента на профилях (горизонтальный габарит), в зависимости от выбора в зоне **Индивидуализировать вид при введении коррекции на инструмент**. Эта опция применяется к исходным профилям (неисправленные профили).

Начальный и конечный отрезки: подключает или откл ючает отображение начальных и конечных отрезков в профилях оригинальных (неисправленные профили) и неконструктивных. Начальные и конечные отрезки всегда приводятся в представлении профилей конструкции или с активной коррекцией.

Сетка: подключает или отключает отображение сетки с шагом. Для получения информации о настройках сетки см. ***Индивидуализировать->Виды->Сетка и Рисунки***
ВНИМАНИЕ: сетка может не отображаться, если текущий зум не позволяет "различать" элементы самой сетки.

Специальная сетка: подключает или отключает отображение специальной сетки (эта команда может отсутствовать). Это сетка, заданная непосредственно для отдельных точек и определенная на этапе конфигурации производителем станка Подключение сетки интерпретируется только на виде верхней или нижней грани.
ВНИМАНИЕ: сетка может не отображаться, если текущий зум не позволяет "различать" элементы самой сетки.

Курсор: подключает или отключает отображение крестообразного курсора, определяющего активную операцию. Курсор отцентрован на точке применения операции и представлен в двух- или трехмерном изображении в зависимости от активизированного вида. Курсор использует цвета трех осей координат (RGB): ось X - красная (R=red), ось Y - зеленая (G=green), ось Z - синяя (B=blue).

На общем виде детали курсор отображается при задании последовательностей. Для получения информации о настройках курсора см. ***Индивидуализировать->Виды->Индивидуализировать виды***

Исходная точка операции: подключает или отключает отображение графического элемента исходной индикации, заданного для активной операции. Эта команда доступна только если распознается интерпретация Поля О в качестве исходного (ребро или сторона грани). На общем виде детали исходная точка отображается при задании последовательностей

Построение грани: в случае фиктивных граней, имеющих программирование трех значимых точек грани, не соответствующих трем прямоугольным координатам, подключает или отключает отображение построения между запрограммированной осью y (не перпендикулярной оси x грани) и расчетной осью y (перпендикулярной оси x грани).

Показать все фиктивные грани: в виде грани подключает или отключает отображение фиктивных граней, за исключением текущей грани. Если отображение отключено (выбор не активирован), все грани переменной геометрии исключаются из графического представления и это исключение относится к операциям, применяемым к самим граням. Эта команда позволяет упростить отображение в случае программы с заданием большого числа плоскостей.
Если текущая грань не является гранью-деталью, выбор применяется также к граням, автоматически заданным для грани-детали.
В виде грани-детали выбор игнорируется: все грани, заданные на детали, отображаются, и дается возможность исключить представление операций, запрограммированных на видах отдельных граней (см. команду **Операции из других видов**).

Операции из других видов: в виде грани подключает или отключает отображение операций, запрограммированных на других видах грани. Эта команда позволяет упростить отображение в случае сложной программы

На строке состояния отображаются следующие команды:

Привязка к сетке: при подключении ограничивает перемещение курсора вершинами сетки, отображенной или по умолчанию (сетка с шагом). Привязка к сетке определяет отображение положения мыши на строке состояния при следующих операциях:
- считывание координат в инструментах
- прямое применение геометрических элементов.

Показать программу: при выборе указывает, что активизировано отображение программы. Эта команда приводится в строке состояния в зависимости от конфигурации TpaCAD
При запуске TpaCAD это поле всегда активно, независимо от его состояния при закрытии приложения. Эта команда предназначена для считывания очень больших программ, позволяя сократить время управления ими. При отключении этого поля специальное сообщение запрашивает подтверждение. Аналогичным образом, если это поле не выбрано, подтверждение запрашивается также на этапе считывания программы.
Если это поле не выбрано, недоступны также интерактивные режимы при задании текущей операции и применение инструментов. В этих случаях специальное сообщение указывает причину невозможности выполнения этой команды.

5.2 Индивидуализировать вид при введении коррекции на инструмент

Теперь рассмотрим во вкладке Показать группу "Индивидуализировать вид при введении коррекции на инструмент": выбор, сделанный в этой группе, всегдаприменяется при активном виде Коррекции на инструмент, а при выбранном пункте **Габарит профилей**, - также в нормальном виде.

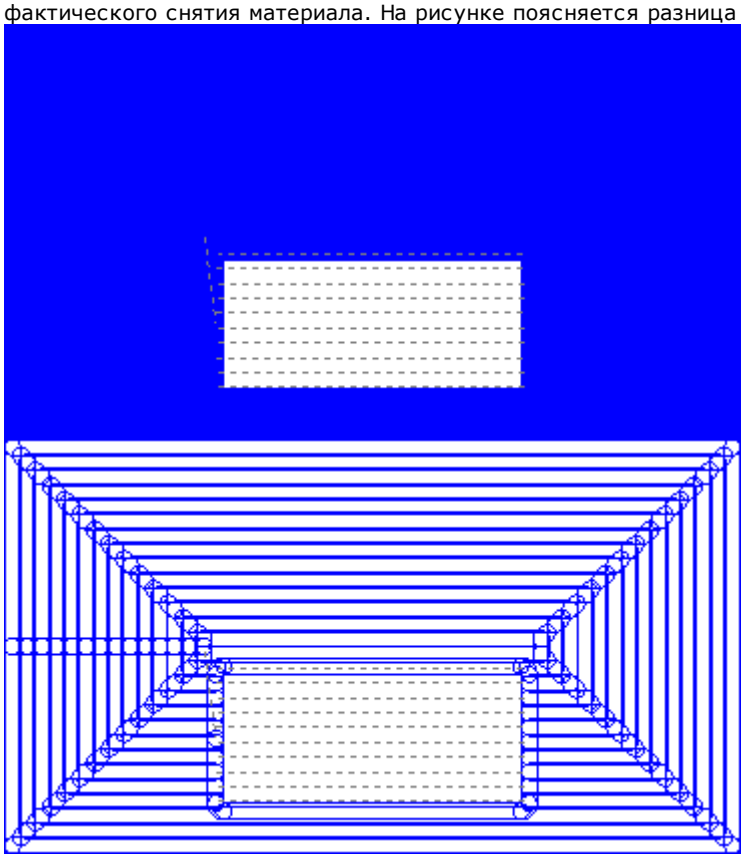
Не показывать габарит профилей: представляет все профили с унитарной толщиной.
Габарит профилей сплошным отрезком: профили представляются сплошным отрезком толщиной, равной габариту инструмента. Выбор исключает представление крайних точек и стрелок направления.



Габарит профилей линейным отрезком: профили представляются толщиной, равной габариту инструмента, но несплошным отрезком. Отображаются наружные контуры габаритов.



Отображение сплошным отрезком может оказаться особенно полезным для профилей выемки для оценки фактического снятия материала. На рисунке поясняется разница между этими двумя представлениями.



В любом случае, остаются представленными с унитарной толщиной:

- профили конструкции
- отрезки профиля, выполненные в воздухе (варианты см. **Индивидуализировать->Виды->Индивидуализировать виды**.



Коррекция исходных профилей: при подключении отображаются исправленные профили и исходные профили (неисправленные профили). Если не активизируется, отображаются только исправленные профили и профили, не применяющие какой-либо коррекции. Этот выбор влияет на вид коррекции на инструмент.

5.3 Регулировать вид

Команды "Масштабирование" и "Сдвиг" позволяют увеличивать, уменьшать, перепозиционировать все, что отображается на панели или на грани. "Масштабирование" и "Сдвиг" меняют только размер зоны, представленной в окне вида. Подключаются из контекстуального меню, вызываемого из зоны графического представления нажатием правой кнопки мыши.



Сдвиг: перемещает панель мышью в графической зоне. После выбора этой команды отображается специальный курсор: удерживать нажатой левую или правую кнопку мыши и

отбуксировать в нужном направлении. При отпускании этой кнопки выполнение команды завершается.
Для подключения этой команды напрямую мышью удерживать нажатой правую кнопку и переместиться в нужном направлении.



Вид по умолчанию: восстанавливает трехмерное (3D) отображение детали на вращениях соответствующее Виду по умолчанию.



Задать вид по умолчанию: присваивает текущие вращения детали (в трехмерном (3D) отображении) как отображение по умолчанию. Отображение по умолчанию присваивается при открытии программы при выбранном трехмерном (3D) отображении.



Зум Продолжения: отмерить область чертежа таким образом, чтобы отображались детали или грани во всем их габарите в максимально разрешенном масштабе представления. В частности, центруется на виде детали или грани с добавлением свободного пространства. Эта команда может подключаться с клавиатуры функциональной клавишей **[F6]**.



Зум окно: выбрать прямоугольное окно, содержимое которого будет увеличиваться, в максимально допустимом масштабе представления. После выбора этой команды появляется специальный курсор: удерживать нажатой левую кнопку мыши и буксировать до получения нужного окна. При отпускании этой кнопки выполнение команды завершается.
Эта команда может подключаться с клавиатуры сочетанием клавиш **[CTRL+W]**.



Зум Предыдущий: восстанавливает предыдущий вид (с памятью до 10 уровней). Эта команда может подключаться с клавиатуры сочетанием клавиш **[CTRL+Shift+W]**.



Зум Все: отмерить область чертежа таким образом, чтобы отображались детали и операции во всем их габарите.



Зум Увел.-Умен.: подключает команду динамического изменения масштаба представления. После выбора этой команды отображается специальный курсор: удерживать нажатой левую (или правую) кнопку мыши и буксировать верх для увеличения зума или вниз для его уменьшения.
Эта команда может подключаться с клавиатуры сочетанием клавиш **[CTRL+правая кнопка мыши]**.



Зум Увеличение: представление дизайна расширяет



Зум Уменьшение: представление конструкции уменьшается.

Зум Увеличение/Уменьшение с помощью мыши: эта команда всегда подключена. Увеличивает или уменьшает текущий масштаб отображения (приближение или удаление). Для увеличения изображения покрутить колесико мыши вверх, для уменьшения изображения покрутить колесико мыши вниз.

Команды для трехмерного поворота детали подключаются с клавиатуры или мышью.
Вращение вверх: деталь поворачивается вверх с горизонтальной осью вращения. Это вращение подключается с клавиатуры при нажатии клавиши **[X]** и завершается при отпускании этой клавиши.
Вращение вниз: деталь вращается вниз с горизонтальной осью вращения. Это вращение подключается с клавиатуры нажатием сочетания клавиш **[Shift+X]** и завершается при отпускании этих клавиш.
Вращение влево: деталь вращается влево с вертикальной осью вращения. Это вращение подключается с клавиатуры при нажатии клавиши **[Y]** и завершается при отпускании этой клавиши.
Вращение вправо: деталь вращается вправо с вертикальной осью вращения. Это вращение подключается с клавиатуры нажатием сочетания клавиш **[Shift+Y]** и завершается при отпускании этих клавиш.
Вращение по часовой стрелке: деталь вращается по часовой стрелке на плоскости вида с осью вращения, перпендикулярной этому виду. Это вращение подключается с клавиатуры при нажатии клавиши **[Z]** и завершается при отпускании этой клавиши.
Вращение против часовой стрелки: деталь вращается против часовой стрелки на плоскости вида с осью вращения, перпендикулярной этому виду. Это вращение подключается с клавиатуры нажатием сочетания клавиш **[Shift+Z]** и завершается при отпускании этих клавиш.
Вращение при помощи мыши: для вращения детали необходимо удерживать нажатой левую кнопку мыши и перемещать курсор в направлении, в котором вы желаете вращать деталь.

5.4 Трехмерное изображение

Команды для выбора режима графического отображения сгруппированы во вкладке **Отобразить** в группе **Навигация.**



Трехмерный вид: активизирует трехмерное представление. При подключенном трехмерном виде можно поворачивать деталь на трех плоскостях, заданных, подключая команды с клавиатуры или мышью, как описывается в предыдущем разделе.

Эта команда может подключаться из контекстуального меню, вызываемого из зоны графического отображения командами во вкладке **Отобразить** или с клавиатуры функциональной клавишей [F2].



Каркасное представление: активизирует двухмерное представление панели с разбивкой. Представляются только грани параллелепипеда. Эта команда может подключаться из контекстуального меню, вызываемого из зоны графического отображения командами во вкладке **Отобразить** или с клавиатуры функциональной клавишей [F3]. При подключенном каркасном представлении отключены все команды вращения детали.



Двухмерный вид: активизирует двухмерное представление выбранной грани. Эта команда может подключаться из контекстуального меню, вызываемого из зоны графического отображения командами во вкладке **Отобразить** или с клавиатуры функциональной клавишей [F4]. При подключенном двухмерном виде отключены все команды поворота детали.

Ниже приведен список команд вращения в соответствии с определенной плоскостью (с трехмерным изображением).
Все команды можно активировать из контекстного меню, вызываемого в области графического отображения (группа: Навигация) или на вкладке меню **Отобразить**.



Вид сверху: деталь позиционируется с трехмерным видом с верхней грани (грань 1).



Вид снизу: деталь позиционируется с трехмерным видом с нижней грани (грань 2).



Вид спереди: деталь позиционируется с трехмерным видом с боковой передней грани (грань 3).



Вид сзади: деталь позиционируется с трехмерным видом с боковой задней грани (грань 5).



Вид справа: деталь позиционируется с трехмерным видом с боковой правой грани (грань 4).



Вид слева: деталь позиционируется с трехмерным видом с боковой левой грани (грань 6).



Плоскость грани: деталь позиционируется с трехмерным видом с текущей грани.



Перечертить: воспроизводит общее отображение вида с применением всех заданных на данный момент графических заданий (визуальные элементы, зум, сдвиг, специальные виды и фильтры вида). Эта команда может выбираться во вкладке меню **Отобразить** или с клавиатуры функциональной клавишей [F5].

5.5 Специальные виды и Фильтры вида

Команды для подключения специальных видов и фильтров вида сгруппированы во вкладке **Показать** в группе **Виды.**
Специальные виды и фильтры вида применяются к детали в комплексе, даже при подключении на виде грани, и являются совокупными в представлении, которое определяется их применением.



Варианты выбора: активизирует отображение только выбранных операций.



Коррекция на инструмент: подключает или отключает отображение коррекции на инструмент. При возникновении ошибок в процедуре применения коррекции на инструмент:

- при выборе с вида грани: специальный вид не подключается. При выборе из общего вида: специальный вид подключается только для граней, которые были признаны исправленными;
- в зоне "Ошибки" можно просмотреть ситуации ошибки.

Эта команда может подключаться с клавиатуры функциональной клавишей **[F7]**.



Логические условия: подключает или отключает применение и отображение логических условий. При активизированном выборе отображаются только операции, отвечающие запрограммированным логическим условиям. В частности:

- операции конструкции могут полностью отключаться от отображения при конфигурации опции в ***Индивидуализировать->Виды->Индивидуализировать виды*.**
- открытые профили (без подготовки) могут полностью исключаться из отображения при таком задании в конфигурации изготовителем станка;
- при задании исключений они применяются и оцениваются так же, как логические условия.

При возникновении ошибок в процедуре применения логических условий:

- при выборе с вида грани: специальный вид не подключается. При выборе из общего вида: специальный вид подключается только для граней, которые были признаны исправленными
- в зоне Ошибки отображаются ситуации ошибки.

Эта команда может подключаться с клавиатуры функциональной клавишей **[F8]**.



Слои: подключает отображение только операций, заданных на уровнях с подключенным состоянием видимости, как задано командой Деталь->Продвинутые задания->Слои. Эта команда недоступна, если не активизирована реализация "Слои".



Специальные виды: задает состояние видимости операций в зависимости от соответствия одной или нескольким значимым заданиям (свойства, технология,..), как задано командой ***Индивидуализировать->Виды-Специальные фильтры***. Эта команда может быть недоступна в меню.

5.6 Информация о профиле

Группа **Информация** во вкладке **Показать** содержит значимую информацию, относящуюся к текущему профилю (в виде грани)**.**
Если текущая операция не относится к профилю: не заполнено ни одно поле и не установлен флажок ни на один значок.



Длина профиля: приводит длину 3D текущего профиля, включая возможные запрограммированные в подготовке входные/выходные отрезки.



Площадь: содержит площадь текущего профиля, если он замкнут. Из оценки замкнутого профиля исключаются возможные запрограммированные на подготовке входные и выходные отрезки



отображает изображение, показывающее перемещение инструмента для текущей строки. Информационное сообщение (подсказка), которое отображается при проведении указателя мыши по изображению, описывает перемещение инструмента: опускание на рабочую координату, опускание и подъем в воздухе, перемещение инструмента в детали.



Применить многократную подготовку: на значке установлен флажок, если профиль применяет многократные подготовки



Профиль замкнут: на значке установлен флажок, если профиль геометрически замкнут на всех координатах (XYZ). Из оценки исключаются возможные запрограммированные на подготовке входные/выходные отрезки.
В случае профиля с многократной подготовкой оценка выполняется на первой запрограммированной подготовке.



Применить вход к профилю: на значке установлен флажок, если входной отрезок запрограммирован на подготовке и правильно решен.
В случае профиля с многократной подготовкой оценка выполняется на первой запрограммированной подготовке.



Применить выход к профилю: на значке установлен флажок, если выходной отрезок запрограммирован на подготовке и правильно решен.
В случае профиля с многократной подготовкой оценка выполняется на первой запрограммированной подготовке.

6 Деталь

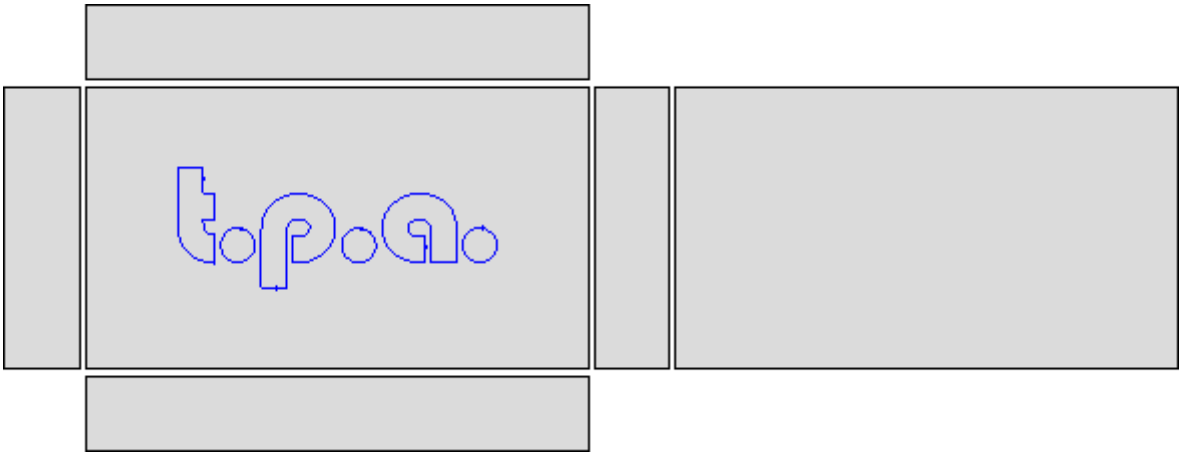
6.1 Графическое представление Общего вида

Представляется деталь в трехмерном изображении со всеми заданными фиктивными гранями, в т. ч. за пределами базового параллелепипеда, или в каркасном представлении, без фиктивных граней.

Представление детали в трехмерном изображении
Операции представляются в пространстве так, чтобы был виден реальный габарит во всех направлениях.



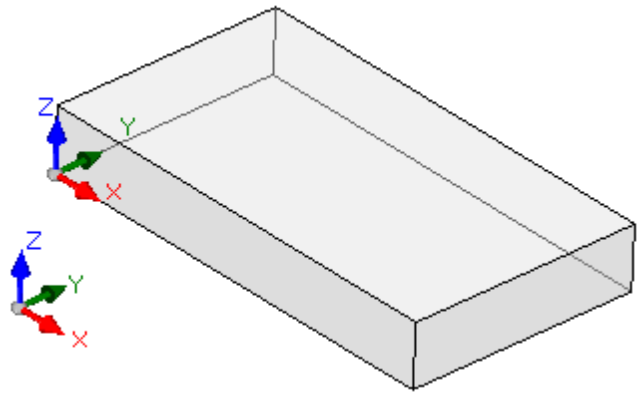
Представление детали в каркасном представлении
Деталь представляется с разбивкой граней базового параллелепипеда. Операции представляются на плоскости грани, на которой они применяются



6.2 Геометрия детали

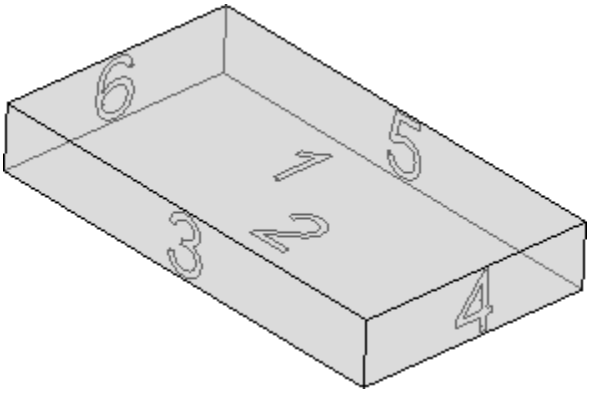
Деталь - объект в форме параллелепипеда, характеризующийся следующими данными:
- три размера: длина, высота и толщина. Эти три размеры детали обозначены буквами: **l, h, s**.
- шесть граней.

ТраCAD использует трехмерную неподвижную систему прямоугольных координат, которую мы назовем **Абсолютная система координат детали**, общую для всех деталей, которая задается, как показано на рисунке:



- оси обозначены как X, Y и Z
- начало координат системы расположено на ребре внизу слева детали
- ось X связана с длиной детали (обозначается l) и имеет положительное направление вправо
- ось Y связана с высотой детали (обозначается h) и имеет положительное направление внутрь детали
- ось Z связана с размером толщины детали (обозначается s) и имеет положительное направление вверх

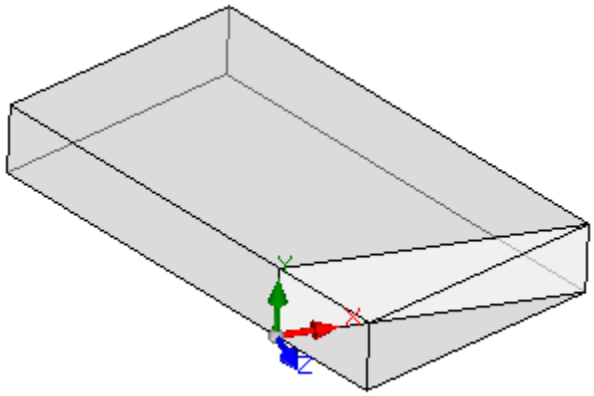
Шесть граней параллелепипеда мы будем называть **реальными гранями,** они пронумерованы с 1 до 6. На рисунке представлена **автоматическая** нумерация граней:



- верхняя грань имеет номер 1
- нижняя грань имеет номер 2
- передняя грань имеет номер 3
- боковая грань справа имеет номер 4
- грань, противоположная передней, имеет номер 5
- боковая грань, противоположная правой, имеет номер 6.

ТраCAD может конфигурироваться для работы с нумерацией, отличной от автоматической. В этом случае мы говорим, что задана **клиентская** нумерация, в которой, в любом случае, используются номера от 1 до 6.
ТраCAD может конфигурироваться так, чтобы не управлять одной или несколькими реальными гранями.

Дополнительно к шести реальным граням можно задать другие грани, обычно позиционируемые на детали, которые мы будем называть **фиктивными**.



Фиктивные грани нумеруются от 7 до 99 и могут:
- быть внутренними или находиться частично или полностью снаружи детали
- иметь любой наклон относительно абсолютной системы координат детали
- фиктивная грань может быть назначена одиночному (плоскому, криволинейному) или составному (поверхности) элементу

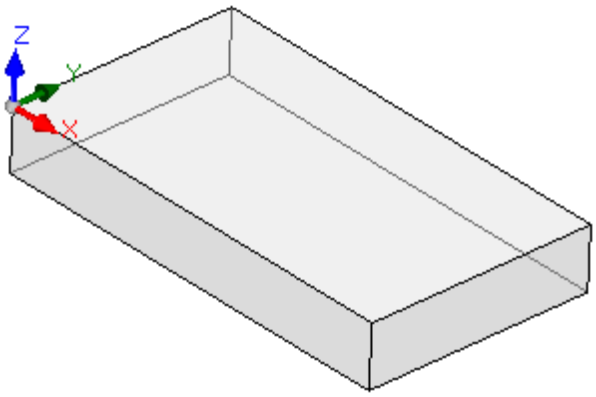
В некоторых оперативных функциях (см. программирование в режиме "грань-деталь") можно задавать дополнительные грани, обычно позиционируемые на деталь, которые мы будем называть **автоматическими**. Автоматические грани имеют нумерацию с 101 до 500 и могут быть назначены только на один элемент (плоский, изогнутый).

Как бы ни была назначена фиктивная (или автоматическая) грань, плоскость XY всегда можно проследить до прямоугольного листа.

Программирование операции в детали всегда соотносится с гранью и использует трехмерную систему прямоугольных координат грани. И в этом случае говорят о трех осях XYZ, где:
- плоскость грани задает оси X и Y
- направление, перпендикулярное плоскости грани, задает ось Z, которую мы назовем осью глубины.

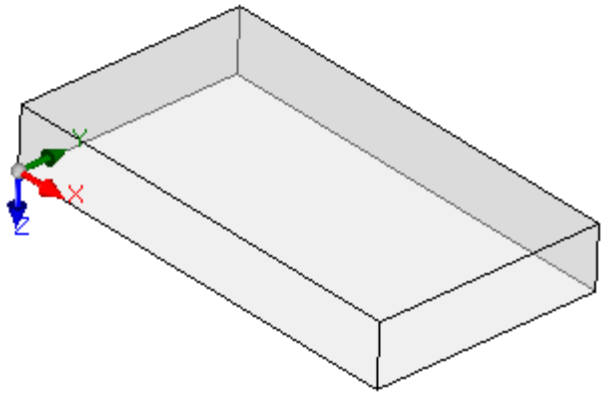
В **Системе координат грани**:
- ось X связана с размером длины грани (обозначим ее **lf**)
- ось Y связана с размером высоты грани (обозначим ее **hf**)
- ось Z связана с размером толщины грани (обозначим ее **sf**)

Рассмотрим системы координат реальных граней при их автоматическом задании:
Грани 1 и 2**:**



Грань 1:

размеры грани
lf=l
hf=h
sf=s



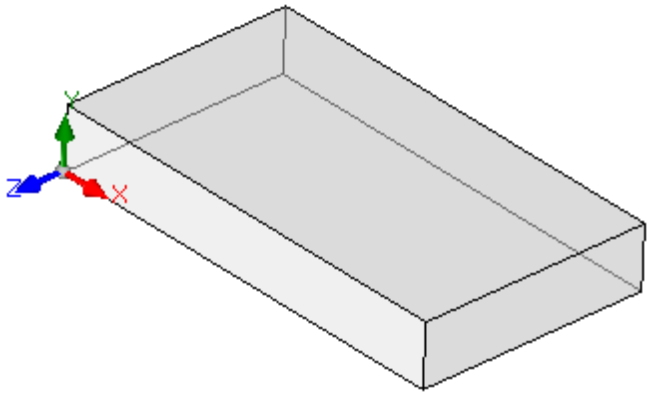
Грань 2:

размеры грани:
lf=l
hf=h
sf=s

локальные системы граней 1 и 2 подобны:
- ось X имеет такую же ориентацию и направление, что и ось X в абсолютной системе координат детали
- ось Y имеет такую же ориентацию и направление, что и ось Y в абсолютной системе координат детали
- ось Z имеет такую же ориентацию, что и ось Z в абсолютной системе координат детали, но на грани 2 имеет противоположное направление.

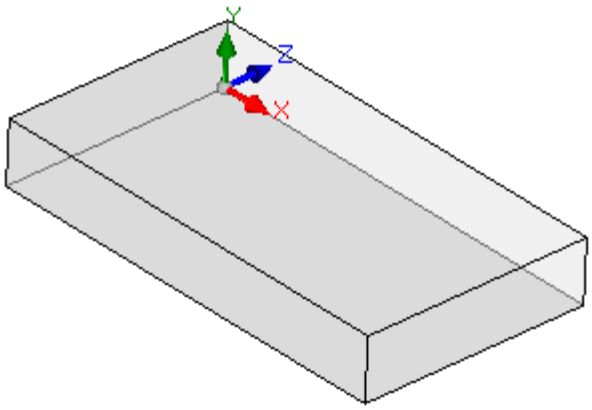
По сравнению с абсолютной системой координат детали начало координат граней:
- на грани 1 находится в (0; 0; s);
- на грани 2 находится в (0; 0; 0).

Грани 3 и 5**:**



Грань 3:

размеры грани
lf=l
hf=s
sf=h



Грань 5:

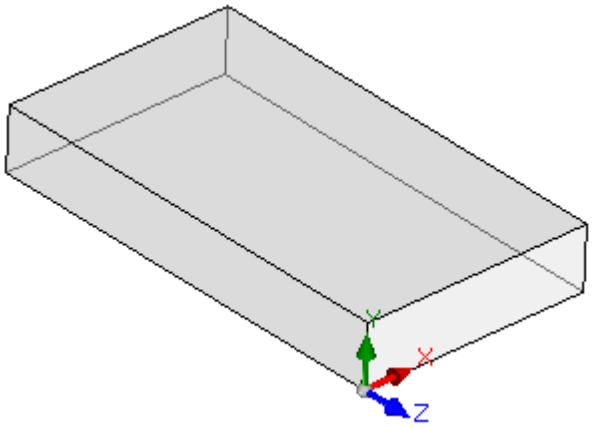
размеры грани
lf=l
hf=s
sf=h

Локальные системы граней подобны
- ось X имеет такую же ориентацию и направление, что и ось X в абсолютной системе координат детали
- ось Y имеет такую же ориентацию и направление, что и ось Z в абсолютной системе координат детали

- ось Z имеет такую же ориентацию, что и ось Y в абсолютной системе координат детали, но на грани 3 имеет противоположное направление.

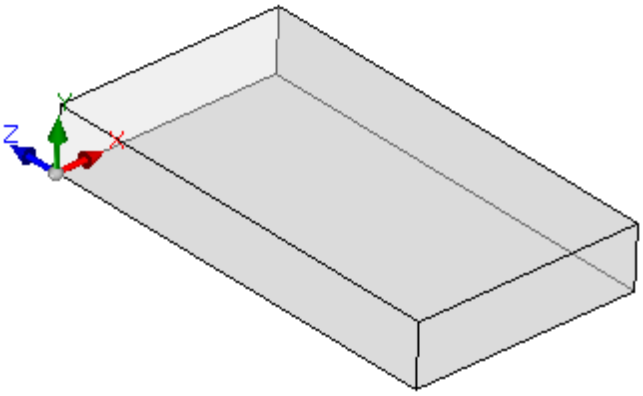
По сравнению с абсолютной системой координат детали начало координат граней:
- на грани 3 находится в (0; 0; 0)
- на грани 5 находится в (0; h; 0).

Грани 4 и 6**:**



Грань 4:

размеры грани
lf=h
hf=s
sf=l



Грань 6:

размеры грани
lf=h
hf=s
sf=l

Локальные системы граней подобны:
- ось X имеет такую же ориентацию и направление, что и ось Y в абсолютной системе координат детали
- ось Y имеет такую же ориентацию и направление, что и ось Z в абсолютной системе координат детали
- ось Z имеет такую же ориентацию, что и ось X в абсолютной системе координат детали, но на грани 5 имеет противоположное направление.

По сравнению с абсолютной системой координат детали начало координат граней:
- на грани 4 находится в (l; 0; 0);
- на грани 6 находится в (0; 0; 0).

На каждой из граней точка, указанная как начало координат, соответствует заданиям, одинаково равным нулю (значение: 0.0) для трех координат точки. Посмотрим, как меняются координаты общей точки, соотнесенной с гранью. Возьмем грань 1 с размерами детали (l=1000; h=800;s=20):
- координата X точки будет положительной при перемещении вправо от начала координат в направлении, показанном (красной) стрелкой оси X, а отрицательные значения позиционируют точку в противоположную полуплоскость, слева от оси Y грани;
- координата Y точки будет положительной при перемещении от начала координат в направлении, показанном (зеленой) стрелкой оси Y, а отрицательные значения позиционируют точку в противоположную полуплоскость, внизу относительно оси X грани;
- координата Z точки будет, *как правило,* положительной при перемещении вверх относительно начала координат в направлении, показанном (синей) стрелки оси Z, а отрицательные значения позиционируют точку “под” плоскость XY, определенную гранью.

Это означает, что точка, позиционированная в центре плоскости XY грани, на входе в деталь 10 мм относительно плоскости грани, имеет координаты: X=500; Y=400; Z=-10. Эта же точка при позиционировании в воздухе относительно детали будет иметь координату Z с обратным знаком: Z=10.

В отношении знака, присваиваемого координате Z, использовался термин *как правило*. Описана наиболее часто используемая ситуация: отрицательные значения глубины определяют врезку инструмента в деталь, а положительные значения задают положения в воздухе относительно детали с применением ко всем граням детали (реальным и нереальным).
Однако можно работать с условностями, прямо противоположными описанным, если так определяется в конфигурации TpaCAD.
Далее в руководстве, как правило, рассматривается работа с принятыми выше условностями.

Локально конфигурации TpaCAD можно по-другому задать локальные системы реальных граней детали, перемещая начало координат плоскости XY грани на другое ребро и/или поворачивая ориентацию осей X и Y. В этом случае говорят, что задана **клиентская** геометрия детали.

В конфигурации TpaCAD можно также выбрать "плоскую" рабочую геометрию, при которой значимым является программирование, заданное на одной или нескольких рабочих плоскостях, координаты: рабочие x, y с заданной глубиной координата z, но не размер по z. В этом случае говорят, что задана геометрия детали в **Абсолютной системе**:
- деталь задает два размера, длину и высоту, и на них представляется: минимальная толщина, в любом случае, задается детали, но автоматически;
- базовое программирование детали ограничивается гранью 1 и деталь представлена прямоугольником;
- в любом случае, можно задать другие рабочие плоскости, например, фиктивные или автоматические грани.

6.3 Задания

Зона "Задания"

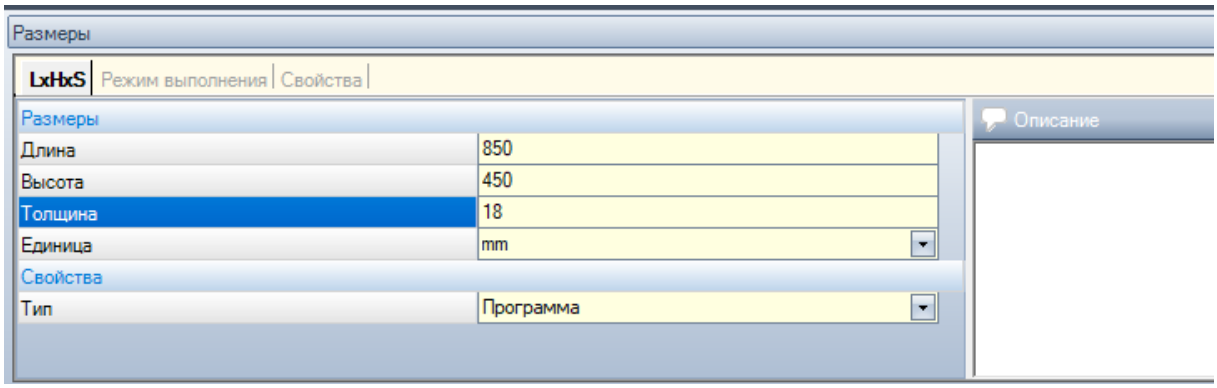
Зона заданий занимает несколько страниц и всегда отображается. Каждая страница содержит и задает группу заданий детали.
Режим редактирования напрямую подключается из общего вида. В виде грани подключается выбором пункта **Изменить** из контекстуального меню или двойным щелчком на заголовке зоны Задание. Для подтверждения или отмены изменений необходимо выбрать, соответственно, пункты **Применить** и **Отменить** контекстуального меню или нажать кнопку **[Enter]**. В последнем случае может быть представлено окно подтверждения.

Выбрать **[Да]** для применения изменений, **[Нет]** для их приостановки.
Выбрать пункт **Не показывать больше это сообщение** и подтвердить нажатием **[Да]** для изменения настройки подтверждения в Индивидуализации TpaCAD на **Автоматическое подтверждение**.
Если при выборе команды из меню или выборе вида грани выявляются несохраненные изменения, эти изменения могут сохраняться и применяться автоматически при подтверждении или отменяться, в зависимости от настроек в Индивидуализации TpaCAD.

Размеры, Режимы выполнения и Свойства

LxHxS

Задается информация общего характера, например, размеры и единица измерения, тип детали, уровни доступа и комментарий.



- **Длина, Высота, Толщина**: размеры детали. Эти три поля принимают только положительные числовые задания (>0.0) максимальной длиной 20 символов. Размеры детали могут использоваться в символической форме для задания переменных или параметров операции; символические имена размеров - соответственно l, h, s. (**См. гл. Параметрическое программирование**). Размеры детали

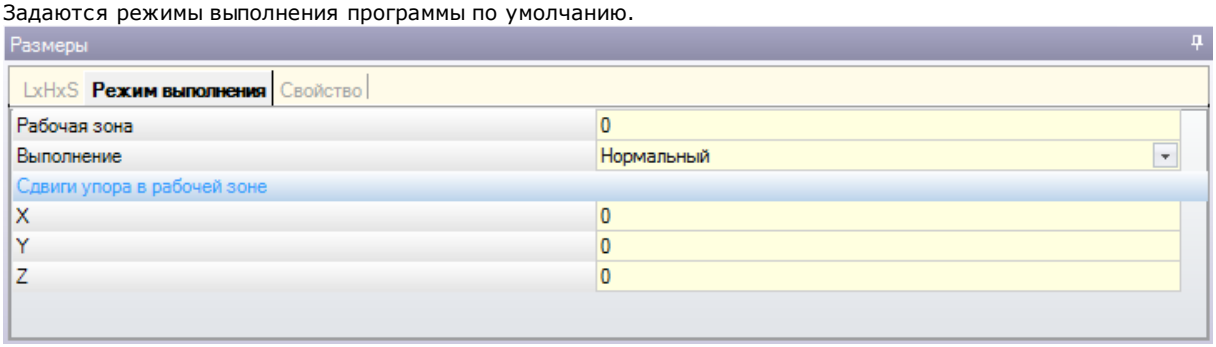
могут повторно задаваться при выполнении программы при сохранении без изменения настроек, записанных в память в исходной программе. **Толщина** детали может не указываться в зависимости от конфигурации TpaCAD.

- **Единица**: единица измерения детали ([мм] или [дюймы]). (Это поле может быть задано нередактируемо из конфигурации TpaCAD)
- **Тип** программы. Программа может быть следующих типов: программа, подпрограмма и макропрограмма с возможностью изменения. Тип макропрограммы предлагается только если уровень доступа равен или выше уровня производителя. Тип подпрограммы может не предлагаться, если на этапе конфигурации TpaCAD разрешается ее создание только с неминимального уровня доступа.
- **Уровни доступа и изменения:** задают, соответственно, минимальный уровень доступа для открытия и записи программы. Нельзя задать уровни выше текущего уровня доступа пользователя, и уровень изменения задается как минимум равным уровню доступа. Если текущий уровень доступа соответствует уровню Оператор, эти пункты не отображаются
- **Описание:** это текст, приводимый как комментарий к программе. Максимальная длина текста - 500 символов.

Зона **ошибок** содержит полный перечень сигналов (ошибок или предупреждений), выявленных при разработке программы.

Режим выполнения

Задаются режимы выполнения программы по умолчанию.



Это необязательная страница.

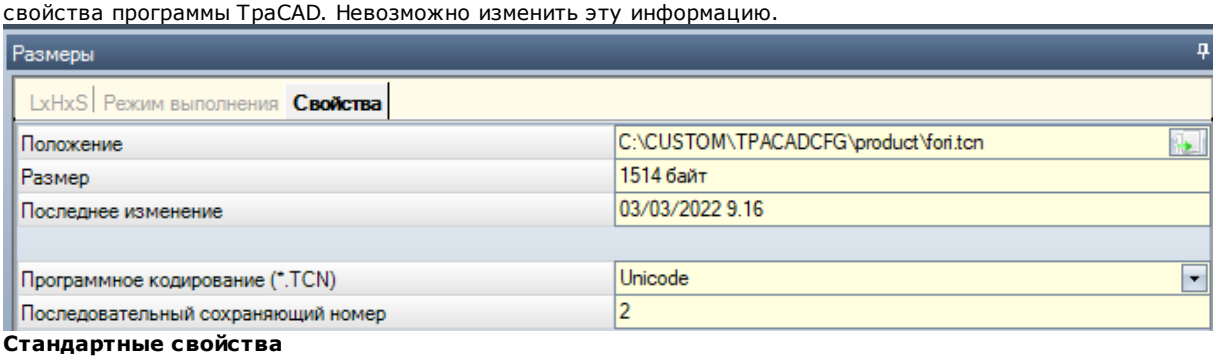
- **Рабочая зона:** задает идентификационный номер рабочей зоны. Это клиентский параметр, т. е. он принимает особое значение при каждом применении. При использовании параметрического программирования этот параметр соответствует переменным аргументам prarea.
- **Исполнение**: режим выполнения программы. Пункты в списке:
  - Нормальный
  - Зеркальное отражение по x
  - Зеркальное отражение по y
  - Зеркальное отражение по xy

  (Фактическое число пунктов в списке может уменьшаться в зависимости от конфигурации TpaCAD). При использовании параметрического программирования параметры соответствуют переменным аргументам prgn, prgx, prgy, prgxy.
- **Сдвиги отсчета в рабочей зоне:** задают положение отсчета выбранной рабочей зоны относительно нуля станка. При использовании параметрического программирования параметры соответствуют переменным аргументам prgx, prgy, prgz.

В качестве опции можно также задать режим выполнения с выбором, направляемым технологией установки: на списке пунктов выбирается рабочая зона, затем на основе этого выбора можно автоматически обновить тип выполнения и положение отсчета, считывая их из технологической конфигурации.
Режимы выполнения детали могут повторно задаваться при выполнении программы при сохранении без изменения настроек, записанных в память в исходной программе.

Свойства

Эта страница отображает некоторые стандартные свойства файла и дополнительные персонализированные свойства программы TpaCAD. Невозможно изменить эту информацию.



Стандартные свойства

- **Положение**: представляет полный маршрут файла. Если программа новая, это поле пустое. Значок в правой части поля активизирует команду копирования маршрута файла в Примечания системы.
- **Положение исходного файла**: представляет полный маршрут исходного файла в случае преобразования формата при считывании
- **Размеры**: представляет размер файла в байтах на момент считывания (0 при новой программе)
- **Последнее изменение**: представляет дату последнего изменения файла по данным на момент считывания (это поле пустое при новой программе)

Персонализированные свойства

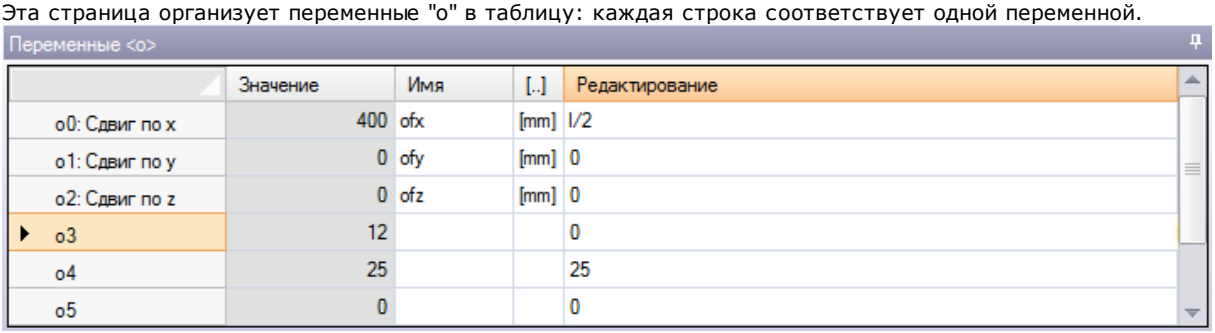
- **Программное кодирование (*.TCN)**: содержит заданную программе кодировку (ANSI или Unicode). Поле доступно для редактирования. Если программа новая, предлагаемое задание выполнено на Unicode, в противном случае оно соответствует кодировке файла в том виде, в каком он был считан. При изменении выбора с Unicode на ANSI при использовании программой настроек Unicode сообщение информирует о потере информации при сохранении программы в формате ANSI;
- **Прогресс сохранения**: представляет числовое значение, которое приращается автоматически при каждом сохранении файла в среде TpaCAD;
- **Признаки только чтения**: если отображается, следует перечень программной информации, которая представляется заблокированной для изменения и которая может касаться: размеров (LxHxS), режима выполнения, переменных (<o>, <v>, <r>), фиктивных граней, клиентских разделов, последовательностей. Ситуация блокировки для изменения может обуславливаться внешним генерированием программ в формате TpaCAD (например, при импорте из внешнего формата) или считыванием уже зарегистрированного файла. Ситуация блокировки всегда может меняться на уровне доступа изготовителя.

  В случае блокировки для изменения изображение с замком указывает это состояние.
  В частности, блокировка может касаться страницы Размеров и/или страницы Режимов выполнения. При неизменяемых размерах наименование программы остается изменяемым.

- **Не отображать, если только для считывания**: выбрать для отключения отображения разделов, переключенных в режим только для считывания. Это выбор, применяющийся ко всем заблокированным разделам за исключением страницы Размеры, которая отображается всегда.

Переменные "O"

Это необязательная страница.
На этапе конфигурации TpaCAD было определено максимальное число управляемых переменных "o" от 0 (нереализованная секция) до 16. Это переменные числового типа, значение которых обычно, но не обязательно, является однозначным для всех программ.
Эта страница организует переменные "o" в таблицу: каждая строка соответствует одной переменной.



- **Заголовок** содержит имя (o0,..., o15) и персонализированное название переменной (например, Сдвиг x).

- **Значение:** отображает значение, получаемое при решении выражения, определенного в столбце **Редактирование**. Это поле не может быть изменено
- **Имя:** отображает связанное с переменной символическое имя, которое может использоваться в параметрическом программировании. Это поле не может быть изменено и задается на этапе конфигурации TpaCAD. Как видно на рисунке, переменная o0 имеет символическое имя “ofx”: эта переменная может вызываться повторно как “o0” или “o\ofx”. Этот столбец не отображается, если ни одна переменная "o" не имеет заданного символического буквального имени.
- **[..]:** отображает единицы измерения переменной:
  - если переменная определяет координату, единица измерения выражается в [мм] или [дюймах]
  - если переменная определяет скорость, единица измерения может выражаться, в зависимости от конфигурации TpaCAD, в [м/мин.] или [мм/мин.] или [дюймах/с] или [дюймах/мин.]
  - если переменная безразмерна, не присваивается.

  Это поле не может быть изменено. Этот столбец не отображается, если ни одна переменная "o" не имеет заданной размерности.
- **Редактирование:** поле, в котором задается переменная. Это поле можно менять и можно задавать число или параметрическое или числовое выражение. Максимальная длина поля - 100 символов. На рисунке приводится пример параметрического выражения: переменная o0 задана ="l/2", где l означает длину детали. Расчетное значение для выражения составляет 300, как приводится в столбце **Значение**. Пример числового выражения ="500/2", которому соответствует значение = 250. Изменение поля в активном состоянии начинается при нажатии любой буквенно-цифровой клавиши или функциональной клавиши F2.
- **Описание: :** отображает описательный текст переменной, который может, например, указывать на значение переменной. Описание определяется на этапе конфигурации и не редактируется. В отсутствие описания этот столбец не отображается.

Задание переменной "o" может параметризироваться по размерам детали (l, h, s) и Режимам выполнения и не могут использоваться другие переменные (‘v’, ‘r’, а также те же ‘o’) или задания изменяемых геометрий или клиентских разделов.
Переменные ‘o’ детали могут повторно задаваться при выполнении программы при сохранении без изменения настроек, записанных в память в исходной программе.

Команды для изменения переменных содержатся в контекстуальном меню, которое можно вызвать нажатием правой клавиши мыши в зоне окна переменных

Импортировать из файла: импортирует задания переменных из выбранной программы.

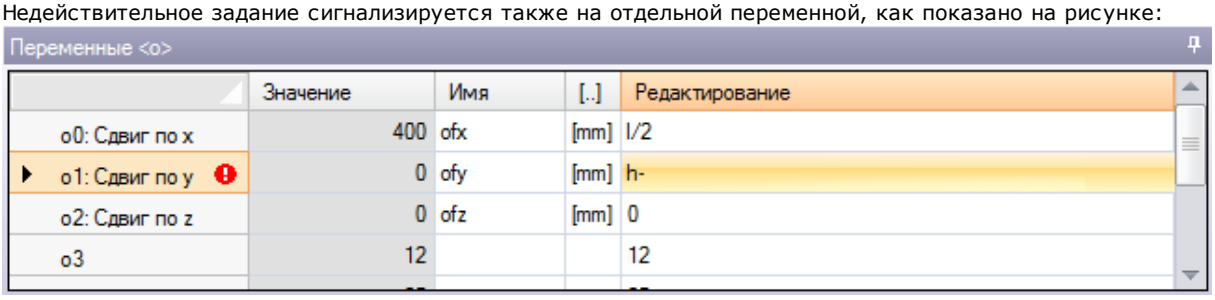
Копировать: копирует настройки выбранных переменных (текущая переменная, если нет выбранных линий) в Локальный буфер. Скопированные переменные доступны для выполнения последующей команды "Вставить" в ту же или другую программу.
Для переключения выбора переменной щелкнуть на ячейке заголовка соответствующей строки, удерживая нажатой клавишу **[CTRL]**. Для отмены выбора для всего списка щелкнуть в любом месте таблицы.

Вставить: вклеивает задания переменных, скопированных ранее в Локальный буфер, сохраняя имена переменных, например, 'o0' задает переменную 'o0', 'o13' задает переменную 'o13'. Эта команда подключена только при наличии в Локальном буфере копии одной или более переменных "o".

Удалить: обнуляет задание выбранных переменных.

Удалить все: обнуляет задание всех переменных.

Зона ошибок содержит перечень только ошибок, возникающих при задании переменных "o".
Недействительное задание сигнализируется также на отдельной переменной, как показано на рисунке:



На этапе программирования можно вызвать немедленную помощь по функциям и аргументам переменных, доступных для параметрического программирования.

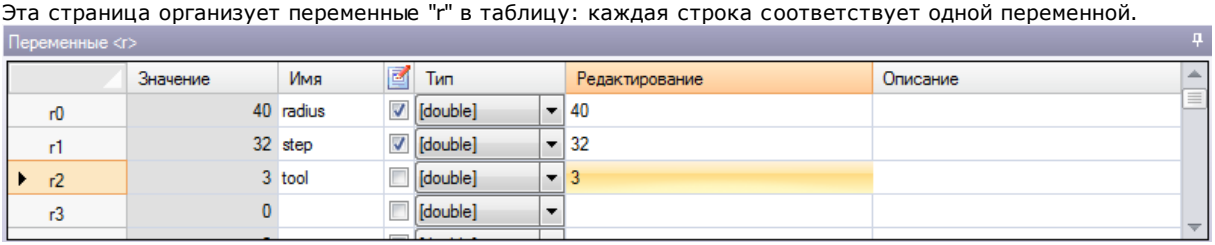
Переменные "V"

Это необязательная страница.
На этапе конфигурации TpaCAD было определено максимальное число управляемых переменных "v" от 0 (нереализованная секция) до 16. Это переменные числового типа, значение которых обычно, но не обязательно, является однозначным для всех программ.
Эта страница организует переменные "v" в таблицу: каждая строка соответствует одной переменной.

Эта страница абсолютно аналогична странице переменных “o”, на которую делается ссылка.

Переменные "R"

В таблице представлен список переменных "r" все также в количестве 300: Это переменные числового или текстового типа, значение которых, как правило, разное в разных программах.
Эта страница организует переменные "r" в таблицу: каждая строка соответствует одной переменной.



- **Заголовок** (r0,...,r299) содержит имя переменной
- **Значение**: отображает значение, получаемое при решении выражения, определенного в столбце Редактирование. В случае переменной типа *строки* (см. поле: **Тип**) результирующее значение приводится в двойных кавычках. Это поле не может быть изменено
- **Имя:** отображает связанное с переменной символическое имя, которое может использоваться в Параметрическом программировании. Синтаксис этого поля имеет максимальную длину 16 буквенно-цифровых символов со строчными буквами. Имя не принимается, если оно уже задано другой переменной r. Как видно на рисунке, переменная r0 имеет символическое имя “радиус”: эта переменная может вызываться повторно как “r0” или “r\radius”.
- : подключает и отключает возможность внешнего перезадания переменной. Переменная "r" может перезадаваться при выполнении программы или при использовании программы в качестве подпрограммы. Предположим, что r0 задает переменную координату для размещения операции сверления:
  - при вызове исполнения программы можно будет изменить значение r0 из внешнего меню
  - при вызове подпрограммы в другой программе можно изменить значение r0 непосредственно на этапе программирования.

  Неперезадаваемая переменная используется для вспомогательных заданий для определения программы. Неперезадаваемые переменные обычно используют перезадаваемые (для тестов, заданий). Можно сказать, что перезадаваемая переменная общедоступна, а неперезадаваемая - локальна (т. е. частная).
  Выбор, в любом случае, отключается, если значение переменной является параметрическим выражением, использующим другие переменные "r".
- **Тип:** задает тип переменной. Предусматриваются два числовых (двойная, целая) и один нецифровой (строка) тип переменных.
- **Редактирование:** поле, в котором задается значение переменной. Это поле можно менять и можно задавать число или числовое выражение или параметрическое выражение. Максимальная длина поля - 100 символов.
- **Описание:** текстовое поле, которое можно задать как комментарий для переменной.

Изменение редактируемого поля в активном состоянии начинается при нажатии любой буквенно-цифровой клавиши или функциональной клавиши F2.

Задание переменной "r" может параметризироваться на:
- размерах детали (l, h, s),
- переменных "o" или "v" (o0 – o15, v0 –v15),
- предыдущих переменных "r" в списке (например, r15, может использовать переменные r от r0 до r14).

Для более подробного рассмотрения возможностей параметризации переменных см. главу Параметрическое программирование.
Незаданная переменная (поле "Редактирование" не заполнено) имеет значение = 0.0, тип = двойная и не может перезадаваться.
Тип переменной не является фиксированным, а может быть:
- **Двойная**: числового типа; расчетное значение (после всех параметризаций, использованных для задания) сохраняет дробную часть. Примерами являются рабочие координаты, скорость смещения.
- **Целая**: числового типа подобно предыдущему, но в расчетном значении дробная часть отбрасывается. Примерами являются счетчики, выбор функционирования, скорость вращения.

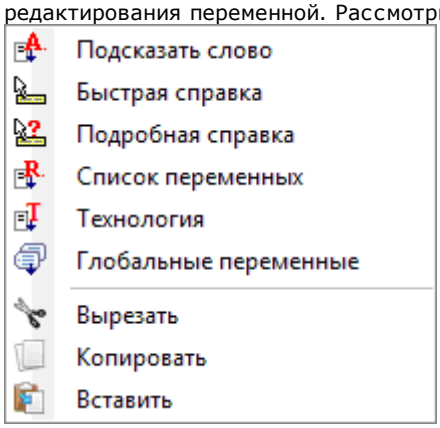
- **строка**: нечисловой тип. Примеры: имя подпрограммы, надпись. Расчетное значение также имеет тип строки.

Команды, доступные для изменения переменных, содержатся в контекстуальном меню, которое может вызываться нажатием правой клавиши мыши в зоне окна переменных: использование этих команд абсолютно аналогично использованию переменных “o”, на которые делается ссылка.

Зона ошибок содержит перечень только ошибок, возникающих при задании переменных "r".
Как уже говорилось для переменных "o", недействительное задание сигнализируется также на отдельной переменной.

Направляемое редактирование

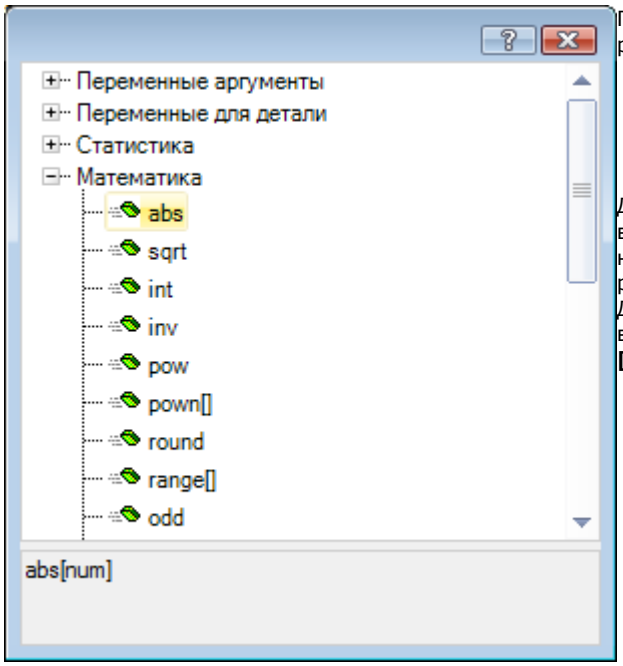
При программировании поля значения всегда можно вызвать меню немедленной справки функций, аргументов переменных и переменных, доступных для параметрического программирования, а также традиционные команды изменения поля редактирования (Вырезать, Копировать, Вставить). И в этом случае речь идет о контекстуальном меню, которое открывается при щелчке правой кнопки мыши в поле редактирования переменной. Рассмотрим полное меню:



Меню справки для направляемого редактирования можно вызвать: для переменных (<o>, <v>, <r>), для полей в клиентских разделах,
при задании фиктивных граней или для задания операций.

- **Подсказать слово:** открывает меню, в котором имеются все функции и аргументы параметрического программирования, сгруппированные в узлы. Приведенные узлы, как и состав каждого узла, зависят от нескольких факторов:

- контекст, в котором открывается меню (задание переменной, геометрия переменной или параметр операции);
- тип задаваемого поля: числовое или строка;
- конфигурация TpaCAD.



Пункты в перечне - двух типов и отличаются различными значками:

geo[alfa;] : функция

eps : переменный аргумент

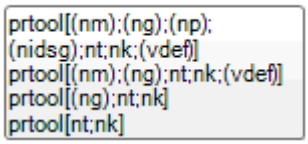
Для выбора пункта: открыть узел и переместить выбор на интересующую строку. В случае функции в нижней зоне окна приводятся форматы, распознаваемые при вызове этой функции.
Для вызова страницы справки с описанием выбранной функции или аргумента нажать кнопку **[F1]** или кнопка **?** окна.

Подтвердить выбор двойным щелчком мыши или нажатием [ENTER]: выбранный пункт вводится в поле редактирования, в положение курсора. В примере на рисунке:

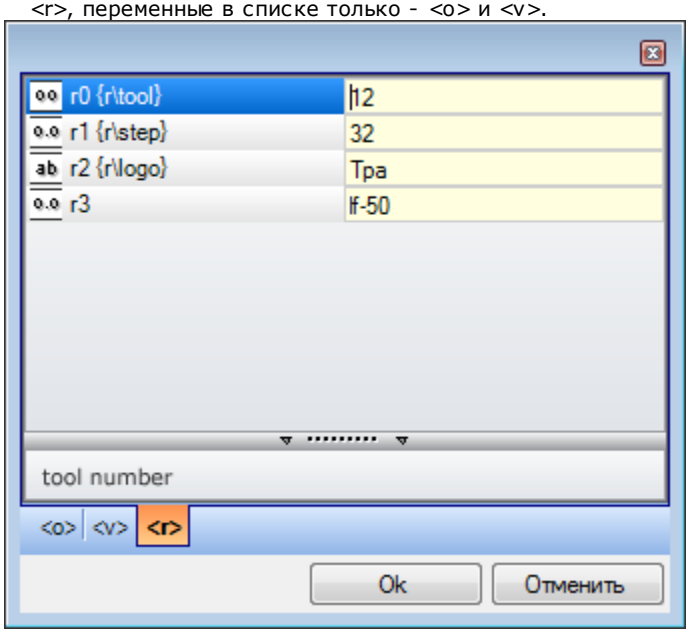
- подтверждение для функции *abs* вызывает ввод строки "abs" (формат этой функции не требует обязательного использования квадратных скобок);
- подтверждение для функции *pown* вызывает ввод строки "pown[]" (должны обязательно использоваться квадратные скобки).

- **Быстрая справка**: открывает окно справки (подсказки) по режиму использования функции, на которой установлен курсор в поле редактирования. Этот пункт не отображается, если поле редактирования пустое или если курсор не установлен на правильное имя. Отображение подсказки автоматически отменяется через несколько секунд. На рисунке показана функция, управляющая несколькими форматами:

prtool[]



- **Подробная справка**: открывает страницу справки по функции, на которой установлен курсор в поле редактирования.
- **Список переменных**: открывает окно со списком программных переменных. При задании переменных <r>, переменные в списке только - <o> и <v>.



На рисунке выбрана страница отображения переменных <r>. Каждая заполненная строка соответствует заданной переменной:

- значок в первом столбце каждой строки указывает тип переменной. На рисунке r0 - целого типа 00, r1 и r3 - двойного типа 0.0, r2 - типа строки ab.

- затем приводится автоматическое имя переменной (r0, r1,..) и, если задано, в фигурных скобках приводится символическое имя (“r\tool”,..)
- в последнем столбце отображается значение переменной.

При выборе строки в зоне под списком приводится описание переменной (если она задана). На рисунке описание первой строки - "номер инструмента" (tool number).

- **Технология**: открывает окно технологии. Этот пункт не отображается, если не считывается никакая технология или если контроль технологии не управляет требуемой процедурой или при задании переменной или параметра операции типа строки. Выполнить выбор информации (ячейки в таблице или прочее) и подтвердить для ввода вызова технологической функции в поле редактирования, в положение курсора.

- **Список глобальных переменных**: открывает окно со списком *Глобальных переменных* среды TpaCAD. Этот пункт не отображается, если управление переменными не подключено или не задана ни одна переменная.

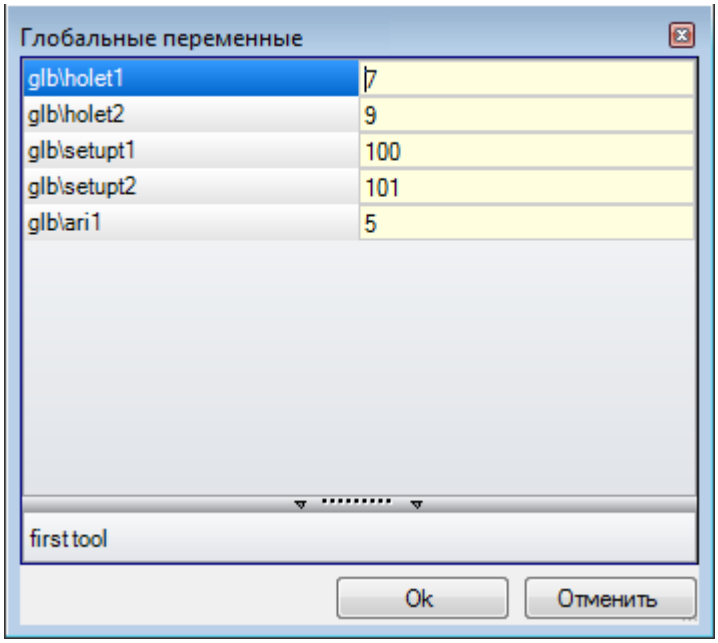


Таблица может представлять максимум 300 переменных числового типа, которые могут вызываться только по символическому имени с форматом: “glb\xxx”, где “xxx”=символическое имя. Переменные заданы в конфигурации TpaCAD и могут использоваться для любого задания.

Восстановление переменных "r" из существующей программы

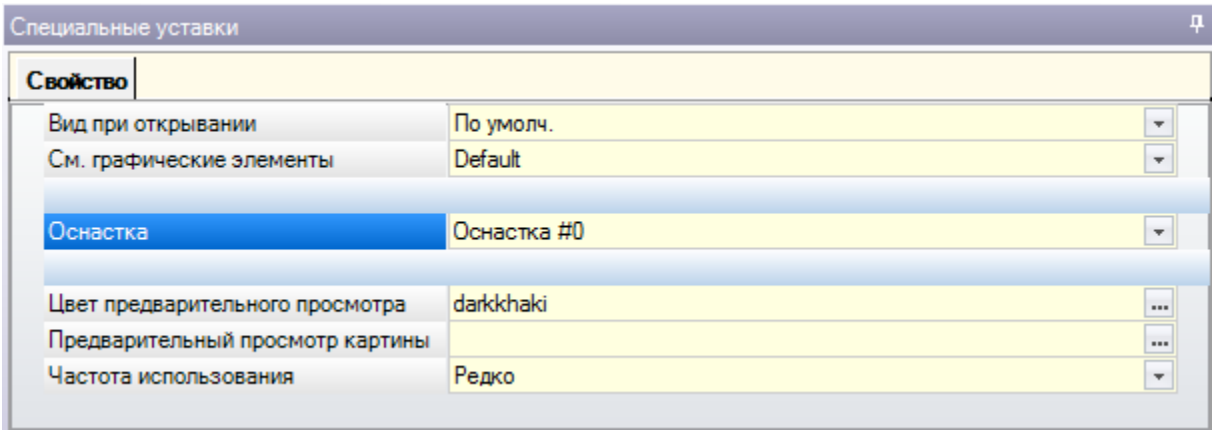
При использовании команды **Импортировать из файла** контекстуального меню можно импортировать весь список переменных "r" другой программы.
Если необходимо восстановить только некоторые переменные существующей программы, использовать команды **Копировать** и **Вставить** по приведенной ниже процедуре:

- открыть программу, из которой вы желаете скопировать переменные.
- открыть страницу переменных "r" и выбрать переменные, которые вы желаете импортировать. Например, от r5 до r9.
- выбрать команду **Копировать** из контекстуального меню для копирования выбранных переменных в Локальные примечания.
- открыть программу, в которую вы желаете импортировать переменные.
- открыть страницу переменных "r".
- выбрать команду **Вставить** из контекстуального меню для вставки скопированных ранее переменных. В частности, в нашем примере перезаписываются переменные с r5 по r9.

Специальные уставки

Это необязательная страница.
В этот раздел входит значимая информация, для которой TpaCAD должен подключить особые распознавания и процедуры, помимо информации исключительно клиентского типа.

Клиентские задания имеют значение, не известное приложению, и конфигурируются на этапе индивидуализации приложения производителем станка.
Даже заголовок раздела (здесь - "Специальные уставки"), может быть другим, т. к. может перезадаваться на клиентском уровне.



Может определяться максимум десять вкладок для группирования необходимых полей:
- девять вкладок, организующих информацию в виде списка, как показано на рисунке. Каждое поле имеет заданный тип редактирования (прямой, выбор в списке и т.д.) и формата (целый, двойной, строка);
- как опция - вкладка (все также в последнем положении), организующая информацию в таблицу: каждый столбец имеет свой тип редактирования и формата с возможностью дублирования информации на определенном числе строк.

Один пункт (информация) назначается в поле определенного типа:
- прямого редактирования числового значения двойного типа (например, "100.5")
- прямого редактирования числового значения целого типа (например, "12")
- прямого редактирования числового значения, задаваемого в параметрической форме (например, "l-100")
- выбора в списке
- списка упорядочиваемых значений
- цвета, выбираемого из палитры цветов
- прямого редактирования общей строки
- поиска файла (открывается окно открытия файла)
- выбора папки (открывается окно выбора папки)

Для каждого пункта может задаваться текст справки, отображаемый в специальной зоне под списком. Дополнительную информацию по значению каждого пункта в этом окне можно получить, обращаясь к Производителю станка.

Команды, доступные для изменения секции, перечислены в контекстуальном меню, которое открывается при нажатии правой клавиши мыши в зоне окна Специальные уставки. В частности:

 импортирует задания секции из выбранной программы

 инициализирует раздел для уставок по умолчанию

 инициализирует только страницу раздела, связанную с заданиями по умолчанию.

Задания по умолчанию считываются программой прототипой.
Среди Специальных уставок могут фигурировать также пункты, имеющие значимое значение:
- Направление зерна: задает зерна на панели
- Края (слева, справа, сверху, снизу): задает края панели, дифференцированные по сторонам.
- Оснастка: выбирает технологическую оснастку в числовом или строчном формате (имя маршрута или имя файла);
- Рисунок и Цвет панели;
- Просмотр графических элементов: это поле позволяет отключать представление стрелок, крайних точек и габаритов трехмерного отображения при загрузке программы. Эта опция удобна, например, при очень больших программах или кривых ISO;
- Вид предварительного просмотра: задает вид, назначаемый при открытии программы, с выбором из 6 граней виденных базовой детали.

Зона ошибок содержит перечень только ошибок, возникающих при задании секции. Недействительное задание сигнализируется также в соответствующем поле изображением  .
Сигнализация ошибки в контексте клиентского раздела может соответствовать полям в следующей форме:

- числовое значение двойного типа (например, "100.5")
- числовое значение целого типа (например, "12")
- числовое значение, задаваемое в параметрической форме (например, "l-100").

В случае параметрического поля: могут возникнуть все ошибки, связанные с неправильным параметрическим программированием.
Параметрическая форма может использовать:

- размеры детали
- переменные "o" и "v"
- переменные "r"

Ошибка может соответствовать также сигналу вследствие недействительного заданного или расчетного значения. Действительные значения заданы в конфигурации TpaCAD как интервал, заданный:

- между минимумом и максимумом; или,
- выше минимального значения; или,
- ниже максимального значения.

Сигналы соответствуют предупреждениям, поэтому они могут диагностироваться только на уровне TpaCAD, или ошибкам, назначенные во время настройки.

Дополнительная информация

Это необязательная страница .
В этот раздел входит информация исключительно клиентского характера. Задания представляются как сконфигурированные в специальной функции и имеют значение, не известное приложению.
Действительно приведенное для клиентской информации Специальных уставок.

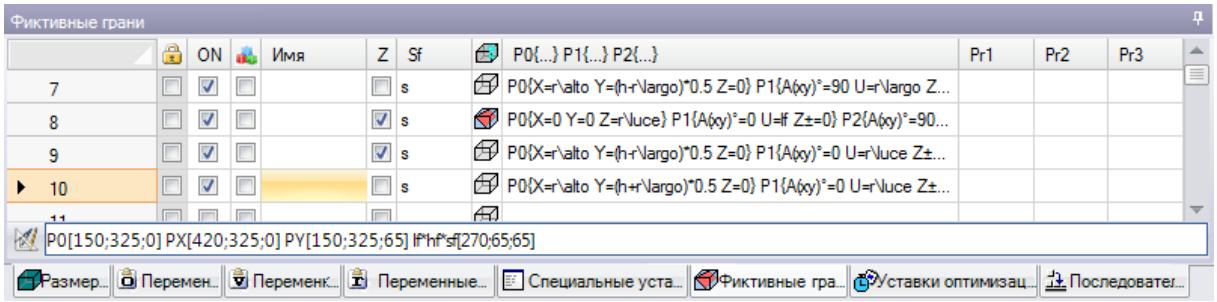
Моделирование

Это необязательная страница.
Описание ассоциированной функции см. в специальной документации (может вызываться со страницы кнопкой F1 или командой в меню .

Фиктивные грани

Это необязательная страница.
Фиктивная грань может задаваться для представления деталей сложной формы, например, паз для стекла в двери, или для облегчения программирования на ориентируемых плоскостях. Фиктивные грани нумеруются по порядку от 7 до 99.

Эта страница организует фиктивные грани в таблицу: каждая строка соответствует одной грани.

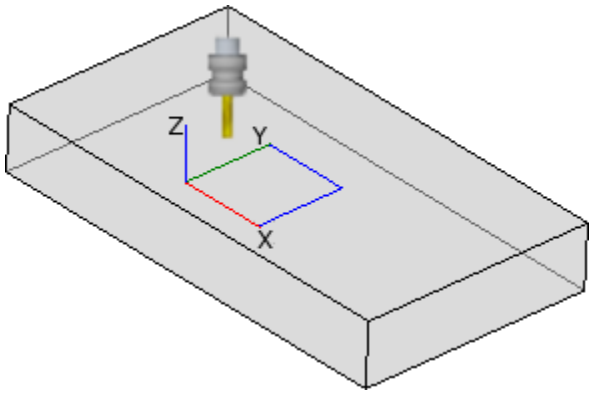


Страница фиктивных граней предлагает полный перечень задаваемых граней от грани 7 до грани 99, т. к. не требуется определять их последовательно.
Заданные в этом окне фиктивные грани графически представлены в зоне общего вида без запрограммированных операций. При выходе из окна задания фиктивных граней графическое представление программы обновляется.
Давайте посмотрим, какую информацию можно назначить для грани:

- **Заголовок** (например, "7") содержит номер грани.
- **ON:** при выборе активизирует задание грани.
- : при выборе указывает, что грань не может быть удалена, т. к. имеет запрограммированные операции (это поле задается автоматически).
- : при выборе активизирует использование грани только в качестве вспомогательной для построения других фиктивных граней. Вспомогательная грань построения не программируется и не может даже учитываться в геометрии детали. Этот столбец отображается только если подключено задание исходной грани (см. ниже).

- **Имя:** содержит имя грани.
- **XY:** открывает окно для задания режима представления плоскости XY грани в двухмерном представлении: ось X горизонтальна или вертикальна и направление для каждой из двух осей координат. Этот столбец отображается только если он активизирован.

- **Z:** выбирает направление оси z в воздухе относительно плоскости xy грани. **Z** - это ось глубины, перпендикулярная плоскости xy грани. При выборе поля задает три левые координаты xyz, в противном случае координаты будут правыми. Заданное здесь направление указывает, как работает инструмент (см. рисунок).



  - При оси Z, заданной положительной вверх, инструмент входит в деталь сверху: три координаты осей - правые (соответствует правилу правой руки, где: ось x соответствует большому пальцу, ось y - указательному пальцу, ось z - безымянному);
  - При оси Z, заданной противоположно показанному на рисунке (положительная вниз), инструмент входит в деталь снизу: теперь три координаты осей - правые (соответствует правилу левой руки, где: ось x соответствует большому пальцу, ось y - указательному пальцу, ось z - безымянному);

При использовании параметрического программирования ориентация оси возвращается универсальной функцией геометрической библиотеки geo[zface; nside]

- **Sf:** задает толщину грани. Если это поле не задано, используется значение по умолчанию = "s" (толщина детали).
  Задание толщины может параметризироваться на:
  - размеры детали
  - переменные "o" и "v"
  - переменные "r"

  При использовании параметрического программирования толщина грани возвращается универсальной функцией геометрической библиотеки geo[sface; nside].
- **:** отображает значок исходной грани использованной при задании фиктивной грани. Информация не может меняться и отображается только если она активизирована.
- **P0{...} P1{...} P2{...}** задает координаты трех точек грани в специальном окне (см. ниже). При этом поле в активном состоянии это окно запускается при нажатии любой буквенно-цифровой клавиши или функциональной клавиши F2.
- **Pr1,Pr2,Pr3,Pr4,Pr5** задает до 5 дополнительных параметров грани. Задание значений может параметризироваться по правилам, действительным для толщины **Sf**. Эти значения интерпретируются как координаты (типология: двойная; единица измерения соответствует единице измерения детали). Столбцы являются дополнительными. При использовании параметрического программирования параметры возвращаются универсальной функцией геометрической библиотеки geo[pr1; nside], geo[pr2; nside],..., geo[pr5; nside].

Команды, доступные для изменения фиктивных граней, содержатся в контекстуальном меню:

Импортировать из файла: импортирует задания переменных геометрий из другой программы. Открывается окно Файл Открыть с возможностью управления предварительным просмотром заданных граней. **ВНИМАНИЕ**: выполнение этой команды не удаляет фиктивные грани с запрограммированными операциями.

Копировать: копирует настройки выбранных граней (текущая грань, если нет выбранных линий) в Локальные примечания. Скопированные грани доступны для выполнения последующей команды "Вставить" в ту же или другую программу.
Чтобы выбрать или отменить выбор грани, щелкнуть на ячейке заголовка соответствующей строки, удерживая нажатой клавишу **[CTRL]**.

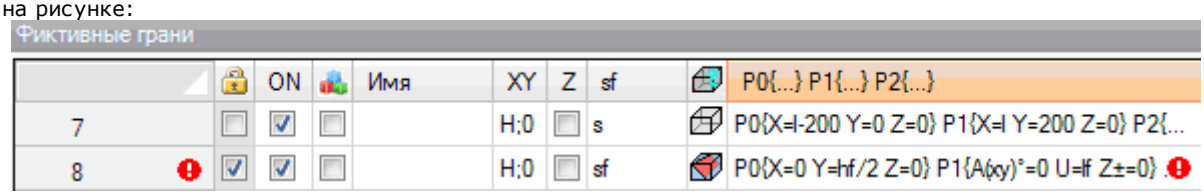
Вставить: присваивает настройки в соответствии с копией в Локальном буфере. Можно выделить два случая:
если Локальные примечания присваивают только одну грань: копия выполняется на текущей грани;
если Локальные примечания присваивают несколько граней: копия выполняется с учетом нумерации граней в примечания. При этом грань 7 копируется на грань 7 и т. д., до завершения элементов в буфере.

Удалить: удаляет задание выбранных или текущей грани, если нет выбранных линий.

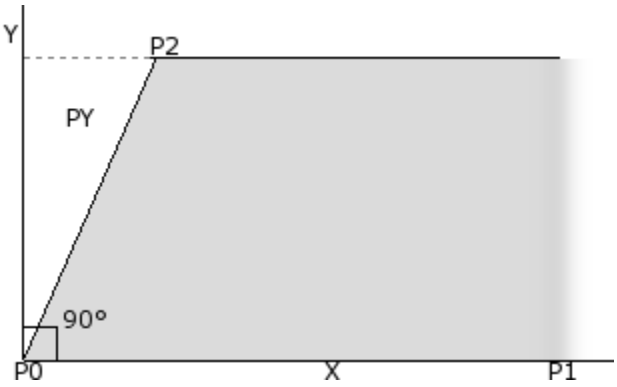


Удалить все: удаляет задание всех фиктивных граней, не имеющих запрограммированных операций и не являющихся исходными для граней, которые не могут быть удалены.

Зона ошибок содержит список только ошибок, которые возникли при задании фиктивных граней. Недействительное задание сигнализируется в строке, а также в соответствующих клетках, как показано на рисунке:



Недействительное задание может относиться к нескольким заданиям: геометрия, толщина, добавленные параметры.
Если геометрия фиктивной грани задана неправильно, плоскость XY грани совпадает с исходной плоскостью и ось Z ориентирована в соответствии с программированием.
Фиктивная грань всегда определяется тремя отдельными точками не на одной линии:



- P0 - это начало координат плоскости xy грани
- P1 - точка ориентации оси x+;
- P2 - третья точка на плоскости xy:
  - если прямая для P2-P0 перпендикулярна прямой для P0-P1: P2 - точка ориентации оси y+;
  - в противном случае: определяется точка, ориентирующая ось y+ в PY.

Расстояние P0-P1 задает длину грани.
Расстояние P0-PY задает высоту грани.

С оперативной точки зрения приведенные на рисунке точки известны не всегда. Не всегда известны значения координат трех точек, однако известно:

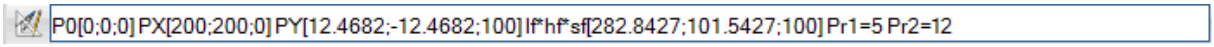
- на сколько грань наклонена относительно другой, или
- какова высота грани, или
- где заканчивается плоскость грани.

Режимы задания этих трех точек грани имеют целью обеспечить все эти три опции. В последующих параграфах приводятся примеры определения фиктивных граней.

Фиктивная грань может определяться подобно другой грани детали, если можно наложить три точки граней операциями переноса в любом направление и/или поворотом только на плоскости XY этой грани.
Плоскости двух граней должны быть параллельны, а полуоси Z - накладывающимися.
На фиктивной грани, подобной одной из 6 реальных граней детали, могут применяться операции и технология, обычно применяемые к реальной грани.
Задание подобной грани выполняется абсолютно независимо от задания исходной грани: подобие оценивает геометрические условия грани после ее определения.

Информация, относящаяся к фиктивным граням на локальной строке состояния

Локальная строка состояния с одной выбранной строкой грани имеет следующий вид:



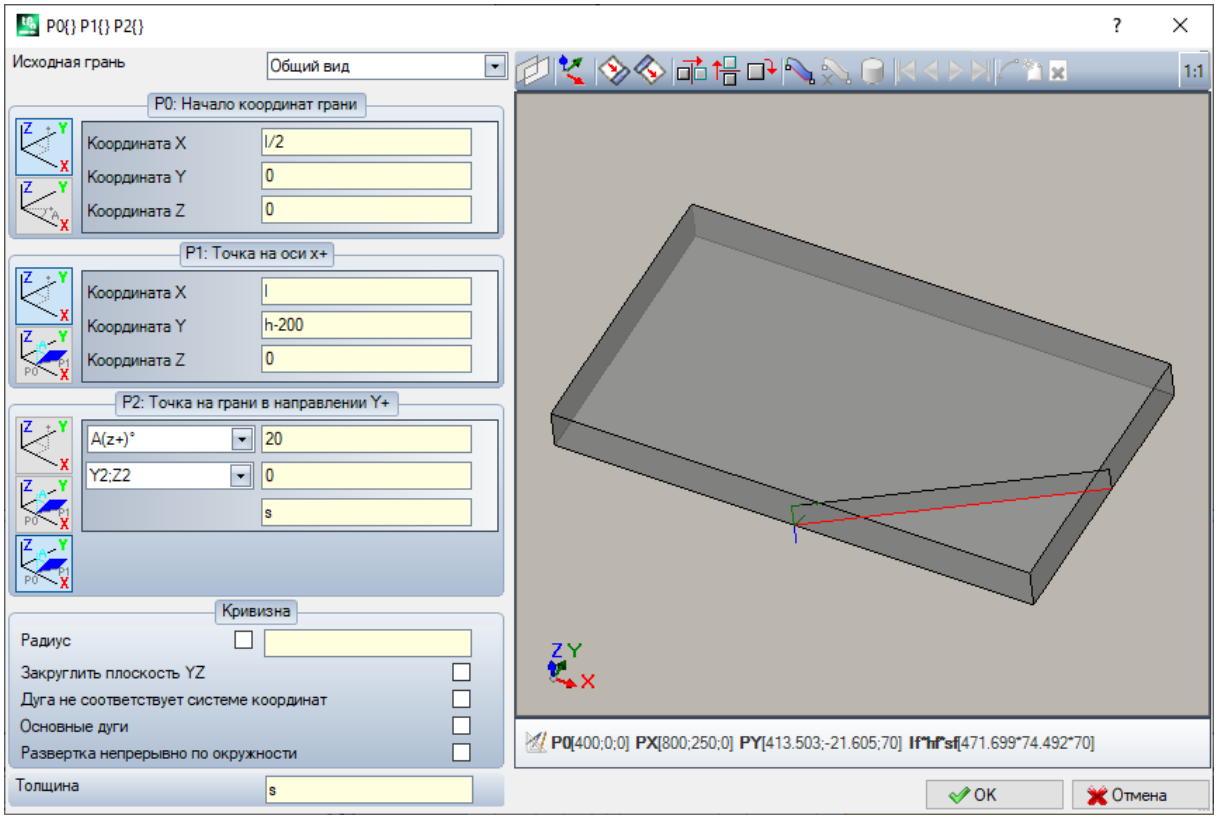
P0[...] начало координат грани
PX[...] крайняя точка по оси x+
PY[...] крайняя точка по оси y+ (расчетная)
lf*hf*sf=[...] размеры грани
Pr1=…Pr5=.. расчетные значения для дополнительных параметров, если таковые заданы.

Закругленные фиктивные грани, поверхности и моделирование

Задание фиктивных граней может управлять также функцией моделирования детали, задания закругленных граней или поверхностей, по которым просим см. специальную документацию (может вызываться командой

“Руководство по моделированию”, предусмотренной в меню).

Пример 1



- **Исходная грань**: выбирается система xyz для использования качестве системы координат для определяемой фиктивной грани. Это может быть абсолютная прямоугольная система координат на детали (выбор в списке: Общий вид) или система xyz другой реальной или фиктивной грани. Из списка исключаются закругленные фиктивные грани или заданные как поверхности.
-  : задает текущую грань так, чтобы она совпадала с выбранной исходной гранью. Эта опция дезактивирована в случае, если исходная грань - Общий вид.
-  : три инструмента применяют значительное преобразование на трех точках грани. В частности: горизонтальная симметрия, вертикальная симметрия и поворот на 90°. Эти инструменты меняют три точки с цифровыми настройками и в прямоугольной системе координат: параметрические назначения или выбор полярной системы теряются.
-  : в случае обнаружения сообщений об ошибках, поданных при определении фиктивной грани, текстовая область под графическим элементом управления переключается на область Ошибок

Задание точек грани:

- **P0: Начало координат грани:** задаются координаты x,y,z начала координат фиктивной грани (точка P0) в прямоугольной (первый выбранный bitmap) или полярной (второй выбранный bitmap) системе координат
- **P1: Точка на оси X+**: задаются координаты точки P1 в прямоугольной (первый выбранный bitmap) или полярной (второй выбранный bitmap) системе координат
- **P2: Точка на грани в направлении Y+:** задаются координаты точки P2 в прямоугольной (первый выбранный bitmap) или полярной (второй выбранный bitmap) системе координат или задается вращение сегмента p0-p2 относительно какой-либо оси.

При выборе последней указанной опции данные в левой клетке позволяют выбирать одну из 6 полуосей координат детали:

- A(z+)°, A(z-)°
- A(x+)°, (x-)°
- A(y+)°, A(y-)°.

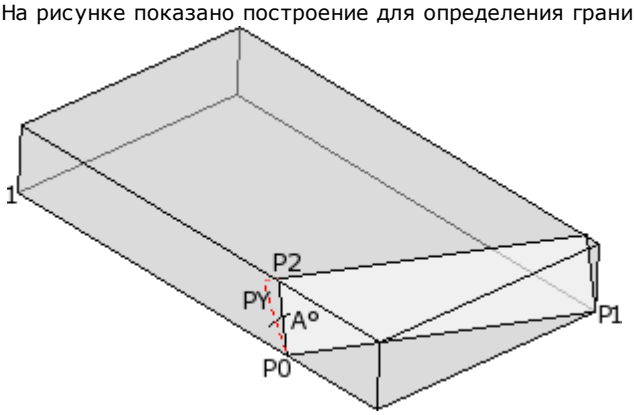
Выбранная полуось задает ось Y исходной осью грани (с началом в P0 и направлением y+ вдоль выбранной полуоси). Это значение задает угол вращения (в градусах) оси y+ грани вокруг своей оси x: ось вращается в положительном направлении к оси z+ грани (выбор как в: **Z в воздухе**)

В примере:
- исходная ось y+ грани задается как Z+ детали;
- угол вращения составляет 20°:
  - при Z в воздухе в правой системе координат: плоскость грани вращается наружу фигуры
  - при Z в воздухе в левой системе координат: плоскость грани вращается внутрь фигуры.

Пока мы задали плоскость грани, но остается позиционировать точку P2: полуось y+ фиксирована, но не положение P2.
Другая клетка выбора позволяет выбирать один из различных способов завершения задания P2:
- hf: задает высоту грани. Точка P2 оказывается на оси y+, на заданном расстоянии (поле справа от клетки выбора). Значение берется как абсолютное значение:
- X2;Y2: задает координаты (X, Y) P2, а координата z вычисляется с условием принадлежности точки плоскости грани
- X2;Z2: задает координаты (X, Z) P2, а координата y вычисляется с условием принадлежности точки плоскости грани
- Y2;Z2: задает координаты (Y, Z) P2, а координата x вычисляется с условием принадлежности точки плоскости грани

При задании двух координат точка P2 обычно оказывается за пределами оси y+. На рисунке выбор сделан на Y2;Z2:
- Y2 задана на поле справа клетки ("0");
- Z2 задана на поле справа, но под клеткой ("s").

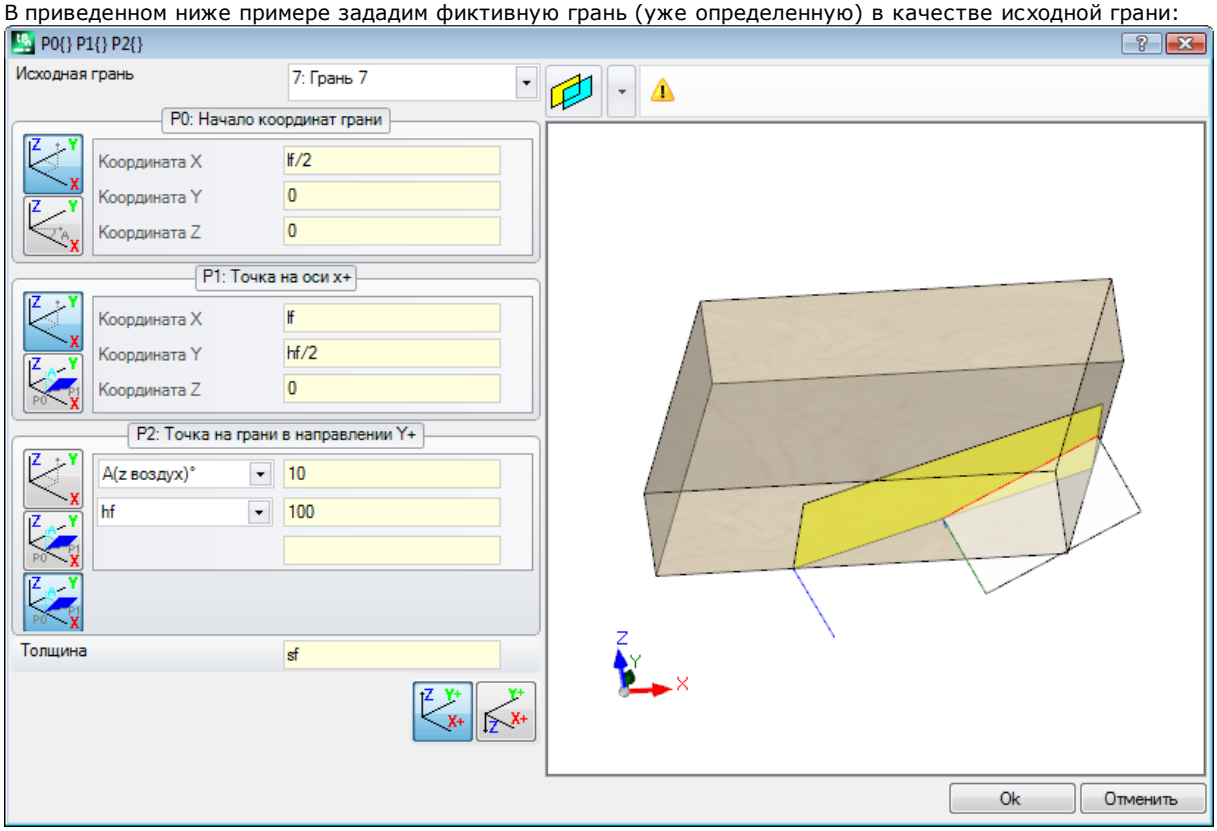
На рисунке показано построение для определения грани с заданными настройками:



- Деталь показана с осями, ориентированными как на чертеже трех картезианских координат, приведенном внизу слева (точка 1 - начало координат осей)
- начало координат грани - P0, а ось X+ находится между P0 и P1
- от P0 показана полуось, ориентированная как Z+ детали: это ось Y+ грани при нулевом вращении
- при выборе правой системы координат ось Z+ грани ориентируется наружу рисунка
- ось Y+ поворачивается на A° (20°-> положительное значение -> вращается к оси Z+ грани)
- координаты (Y, Z) фиксируют точку P2 на левой боковой грани детали: P2 образует с осью X грани (от P0 до P1) угол меньше 90°, поэтому пересчитывается проекция P2 на ось Y грани (PY). Линейные отрезки, соединяющие начало координат грани с P2 и P2 с PY, указывают несовпадение между P2 и PY.

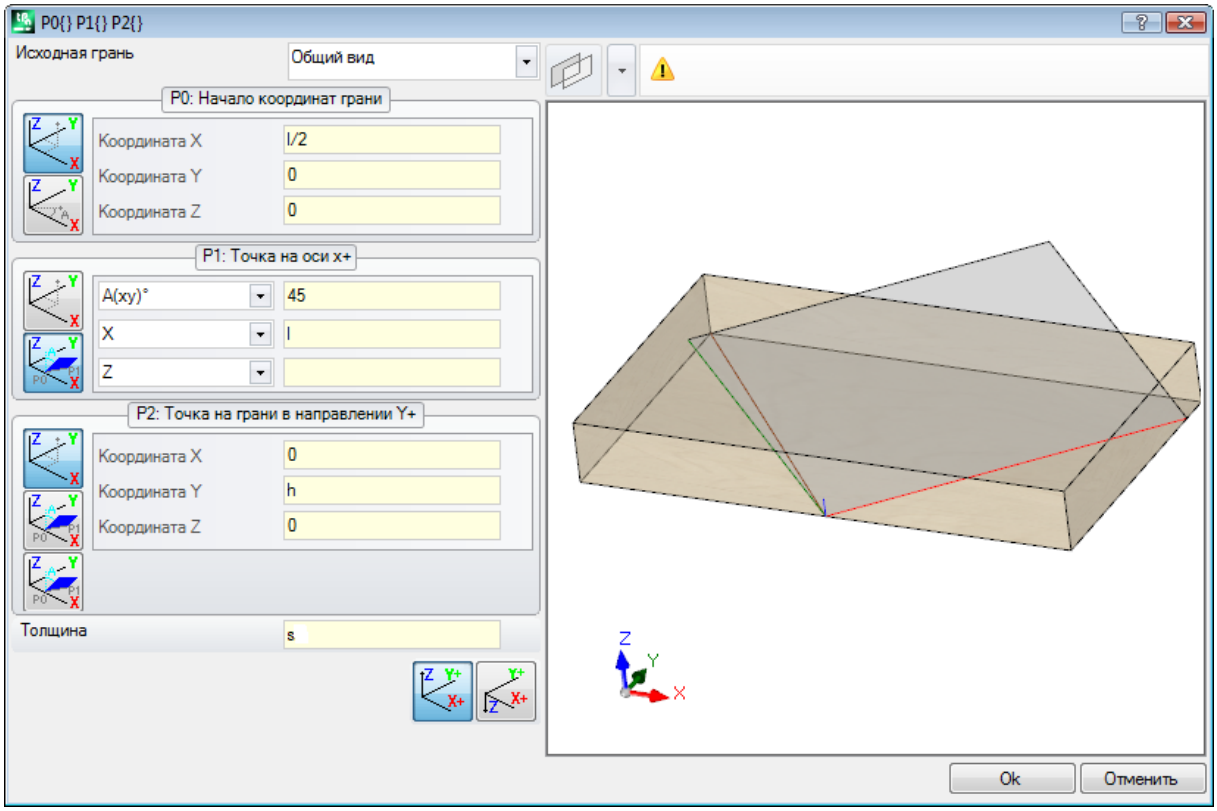
Пример 2

В приведенном ниже примере зададим фиктивную грань (уже определенную) в качестве исходной грани:



- **Исходная грань:** выбор показывает, что необходимо задать грань с использованием системы xyz другой фиктивной грани: грани 7. Для нас грань 7 - это грань, заданная в **Примере 1**. Клетка выбора исходной грани предлагает список граней:
  - все реальные грани детали (**ВНИМАНИЕ**: также неуправляемые реальные грани)
  - фиктивные грани, заданные на детали с номером меньше номера определяемой грани. Если, например, мы задаем грань 8, можно выбрать реальную исходную грань или, если фиктивную, - только грань 7
- **P0: Начало координат грани:** в рамке выбран первый bitmap (битовая карта). Выбор показывает, что для точки известны три координаты, но теперь это координаты, заданные на грани 7. Поля, относящиеся к координатам, позиционируют начало координат грани на половину оси x грани 7 (lf/2; 0; 0):
  - использование в параметрическом программировании переменных аргументов lf, hf, sf (длина, высота и толщина грани) обуславливает использование значений размеров грани 7
  - **ВНИМАНИЕ**: значение Координата Z, если отлично от 0, использует те же принятые знаки, которые используются для граней (отрицательный или положительный при операции на грани)
- **P1: Точка на оси X+:** и в этом случае в рамке выбран первый bitmap. Выбор показывает, что для точки известны три координаты, которые и здесь заданы на грани 7. Поля, относящиеся к координатам, позиционируют точку P1 на (lf; hf/2; 0);
- **P2: Точка на грани в направлении Y+:** теперь в рамке выбран третий bitmap: выбор показывает, что известен наклон грани относительно одной из осей координат исходной грани (грань 7). Задания аналогичны предыдущему примеру:
  - **A(z воздух)°:** соответствует уже рассмотренному выбору A(z+)°, только теперь сообщение показывает, что выбор находится на полуоси z в воздухе (аналогично: A(z детали)° соответствует уже рассмотренному выбору A(z-)°, только теперь сообщение показывает, что выбор находится на полуоси z в операции детали);
  - **hf:** точка P2 теперь определяется заданием высоты грани.

Пример 3



- **Исходная грань:** задается Общий вид. Выбор показывает, что необходимо задать фиктивную грань с использованием абсолютной системы xyz детали
- **P0: Начало координат грани:** в рамке выбран первый bitmap. Выбор показывает, что для точки известны три координаты. Поля, относящиеся к координатам, позиционируют начало координат грани на половину оси x детали (l/2; 0; 0):
- **P1: Точка на оси X+:** в рамке выбран второй bitmap. Выбор показывает, что для точки известны полярные координаты на одной из трех картезианских плоскостей детали (если бы была задана исходная грань, мы бы сказали: "…на одной из трех картезианских плоскостей исходной грани"). Приведенные поля задания отличаются от предыдущих случаев:
  - **A(xy)° 45:** левая клетка выбора позволяет выбирать одну из 3 картезианских плоскостей:
    - A(xy)°: плоскость вращения xy
    - A(xz)°: плоскость вращения xz
    - A(yz)°: плоскость вращения yz

  Это значение задает угол вращения в градусах на плоскости и полюс (центр) полярной системы - точка P0: ось, выходящая из P0 на плоскости и с заданным углом определяет ось X+ грани.
  На трех плоскостях угол вращается положительно:
    - с осью X+, замыкающейся к Y+, при плоскости вращения xy
    - с осью X+, замыкающейся к Z+, при плоскости вращения xz
    - с осью Y+, замыкающейся к Z+, при плоскости вращения yz

  Полуось X+ фиксирована, но не положение P1 на ней.
  - **X l:** клетка выбора слева позволяет выбирать один из 3 различных способов задания P1 на плоскости полярной системы:
    - **U:** задает модуль поляной системы (расстояние P1 от P0 на плоскости вращения). Значение берется как абсолютное значение
    - **X:** задает координату x P1, а координата y вычисляется при условии принадлежности оси x+ грани
    - **Y:** задает координату y P1, а координата y вычисляется при условии принадлежности оси x+ грани

  Координаты, приведенные в списке, соответствуют плоскости вращения:
  - (X, Y), если плоскость вращения xy
  - (X, Z), если плоскость вращения xz
  - (Y, Z), если плоскость вращения yz

  На рисунке выбор сделан на X1=l:

Таким образом, мы задали положение P1 на выбранной плоскости полярной системы, остается определить положение на оси, перпендикулярной плоскости.

- **Z 0:** клетка выбора слева позволяет выбирать один из 3 различных способов задания P1 на третьей оси (в примере - оси Z):
  - Z: непосредственно задает положение;
  - Z±: задает изменение положения относительно значения, заданного в точке P0;
  - AZ°: задает изменение угла относительно значения, заданного в точке P0. Заданное значение приводится к интервалу значений в пределах от -90° до +90°: значение считается правильным, если оно содержится в этом интервале, за исключением пределов (разница между значениями значима, если больше чем эпсилон°=0.001°). Положительные значения угла определяют увеличение координаты, отрицательные значения определяются уменьшение координаты.

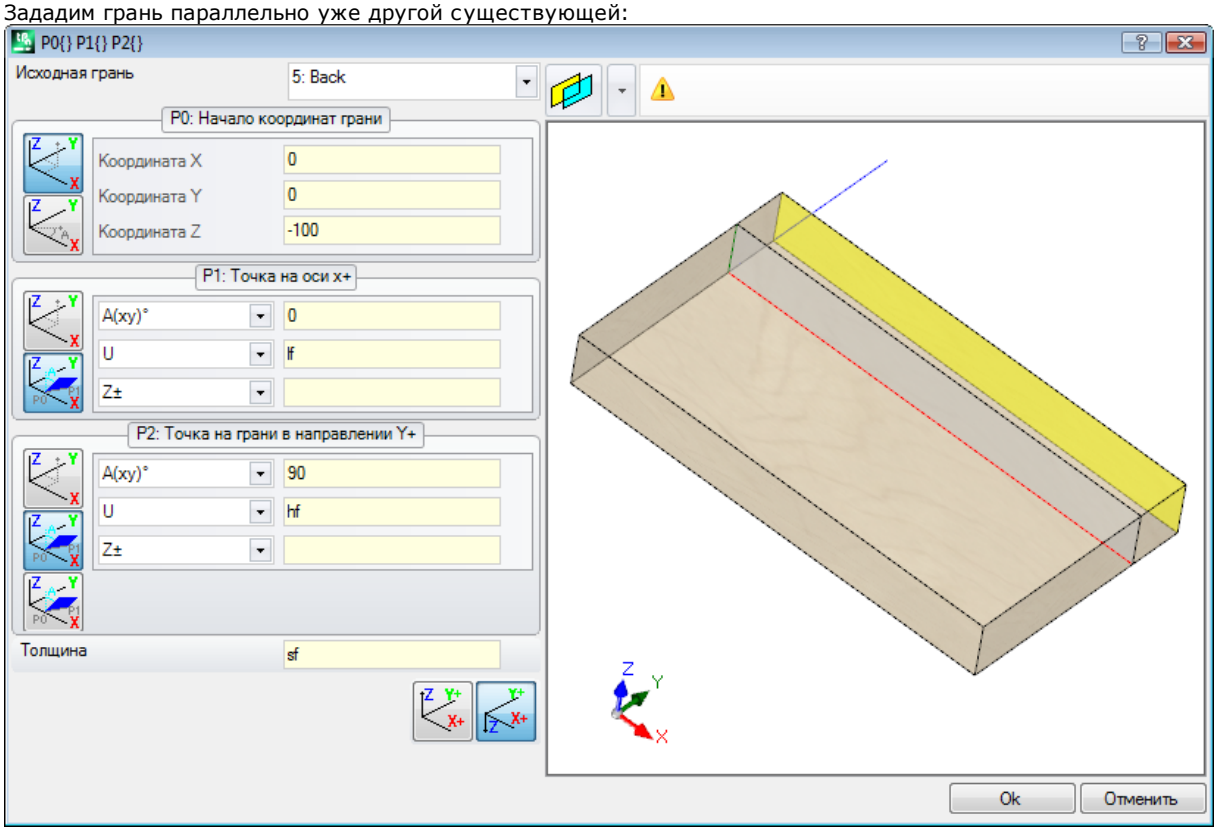
Задаваемая здесь координата соответствует оси, перпендикулярной плоскости вращения:

- z, если плоскость вращения xy
- y, если плоскость вращения xz
- x, если плоскость вращения yz

- **P2: Точка на грани в направлении y+:** в рамке выбран первый bitmap. Выбор показывает, что для точки известны три координаты. Поля, относящиеся к координатам, позиционируют точку P2 на (0; h; 0); P2 образует с осью x грани (от P0 до P1) угол меньше 90°, поэтому пересчитывается проекция P2 на ось y грани (PY). Линейные отрезки, которые в графическом представлении соединяют начало координат грани с P2 и P2 с PY, указывают как раз несовпадение между P2 и PY.

Пример 4

Зададим грань параллельно уже другой существующей:



Исходная грань: Грань 5: затем выберем кнопку  . Зададим координату Z точки P0 на координату –100 для перемещения грани по оси Z- исходной грани и получим грань, как показано на рисунке. Если необходимо сместить точку P0 по координатам x и/или y: задать значения, отличные от 0. Если необходимо задать другие размеры длины и/или высоты: заменить значения lf и/или hf.

Раздел привязок

Это необязательная страница  .
В этот раздел входит информация исключительно клиентского характера. Задания представляются как сконфигурированные в специальной функции и имеют значение, не известное приложению.

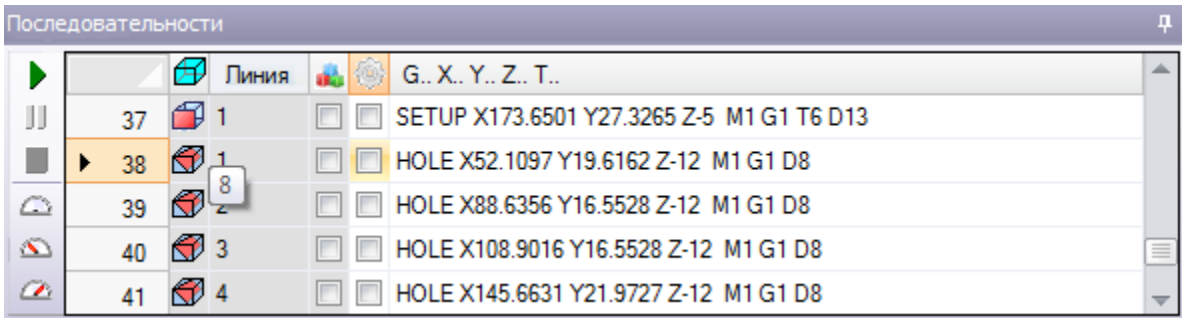
Действительно приведенное для клиентской информации Специальных уставок.

Оптимизации

Это необязательная страница .
В этот раздел входит информация исключительно клиентского характера. Задания представляются как сконфигурированные в специальной функции и имеют значение, не известное приложению.
Действительно приведенное для клиентской информации Специальных уставок.

Последовательности

Это необязательная страница.
Позволяет задавать особый порядок выполнения операций, заданных в целом для детали.
Эта страница организует операции в таблицу: каждая строка соответствует одной операции.



- **заголовок:** содержит номер последовательности по порядку
- : графически отображает грань, на которой запрограммирована операция. При проведении курсора мыши по значку отображается подсказка с номером грани
- **Линия:** номер операции по порядку в программе грани
- : клетка выбирается, если операция является конструкцией
- : эта ячейка указывает основной присвоенный операции цвет в зависимости от типа операции (точечная, подготовка, отрезок профиля) или рабочего кода
- флаг оптимизации операции (необязательный). Интерпретация флага зависит от каждого отдельного применения. Можно подключить или отключить отдельную клетку или, при использовании команд контекстуального меню, можно подключить или отключить несколько клеток. Это поле - единственное изменяемое в таблице поле.
- **Операция:** описательный текст операции
- **G..X..Y..Z:** отображает имя ASCII операции, точку применения и задания технологии (станок, группа, головка, шпиндель, диаметр, технологический приоритет).

Если по меньшей мере одна из фигурирующих в списке операций имеет заданное описание или Имя, добавляется столбец для их отображения.
Команды, предусмотренные для изменения списка последовательностей выполнения, содержатся в контекстуальном меню, которое может вызываться нажатием правой кнопки мыши в зоне окна.

Инициализировать деталь для процедуры программирования: инициализирует список по автоматическому порядку граней (верхняя грань первой, затем: нижняя, передняя, боковая правая, задняя, боковая левая, фиктивные с номерами с 7 по 99) и программирование каждой грани. Эта команда аннулирует все изменения, внесенные вручную при открытии сеанса (Вырезать, Вставить, Задание флага оптимизации).

Инициализировать грань для процедуры программирования: вставляет список операций грани, к которой относится текущая строка. Например, при выборе этой команды на одной строке, которая приводит операцию на грань 4, вставляются все операции грани 4 начиная со той же строки с сохранением порядка программирования этой грани.

Активировать флаг оптимизации: активирует флаг оптимизации выбранных строк. Эта команда необязательна.

Дезактивировать флаг оптимизации: отключает флаг оптимизации выбранных линий. Эта команда необязательна.

Вырезать: удаляет выбранные строки (текущую строку, если нет выбранных строк) из таблицы и вводит их в локальные примечания. Эта команда доступна только если локальные примечания

пусты. Чтобы выбрать или отменить выбор строки, щелкнуть на ячейке заголовка соответствующей строки, удерживая нажатой клавишу **[CTRL]**. Для отмены выбора для всего списка, щелкнуть в любом месте таблицы.



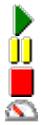
Вставить: вставляет содержимое локального буфера в текущую строку и освобождает локальный буфер. Эта команда доступна только если одна или более строк были внесены в локальный буфер строки вводятся до или после текущей операции в зависимости от подключения флага ввода на строке состояния.



Вставить снизу: выбирает точку ввода сверху/снизу относительно текущей линии.

Можно изменить порядок операций также путем буксировки в пределах зоны таблицы. Эта буксировка применяется к выбранным строкам (текущей строке, если нет выбранных строк) и подключается путем удерживания нажатой правой кнопки мыши до установки в нужное положение в списке. Изменение списка путем буксировки не зависит от состояния Локального буфера.

Строка кнопок слева таблицы приводит команды полезные для моделирования списка:





: запускает графическое моделирование. Текущая операция перемещается с первой на последнюю строку списка с сохранением постоянного периода. Моделирование можно приостановить, выбирая кнопку , и возобновить, выбирая заново . Кнопка завершает моделирование.

Скорость моделирования может меняться при помощи следующих кнопок:



: задает скорость моделирования по умолчанию



: уменьшает скорость моделирования



: увеличивает скорость моделирования

На основе заданного в индивидуализации TpaCAD, графика может дополняться всеми запрограммированными операциями или же только операциями, для которых можно задать последовательность. В этом случае не отображаются операции, соответствующие:

- операциям на грани-детали
- открытым профилям
- вложенным вызовам подпрограммы или макропрограммы
- операциям, для которых дезактивировано использование последовательности

В графическом представлении показывается операция, соответствующая выбранной в таблице строке. Представление может быть в трехмерном изображении (3D), каркасном представлении, отдельной грани (2D).

Можно определить операцию или выполнить выбор также непосредственно в графической зоне:

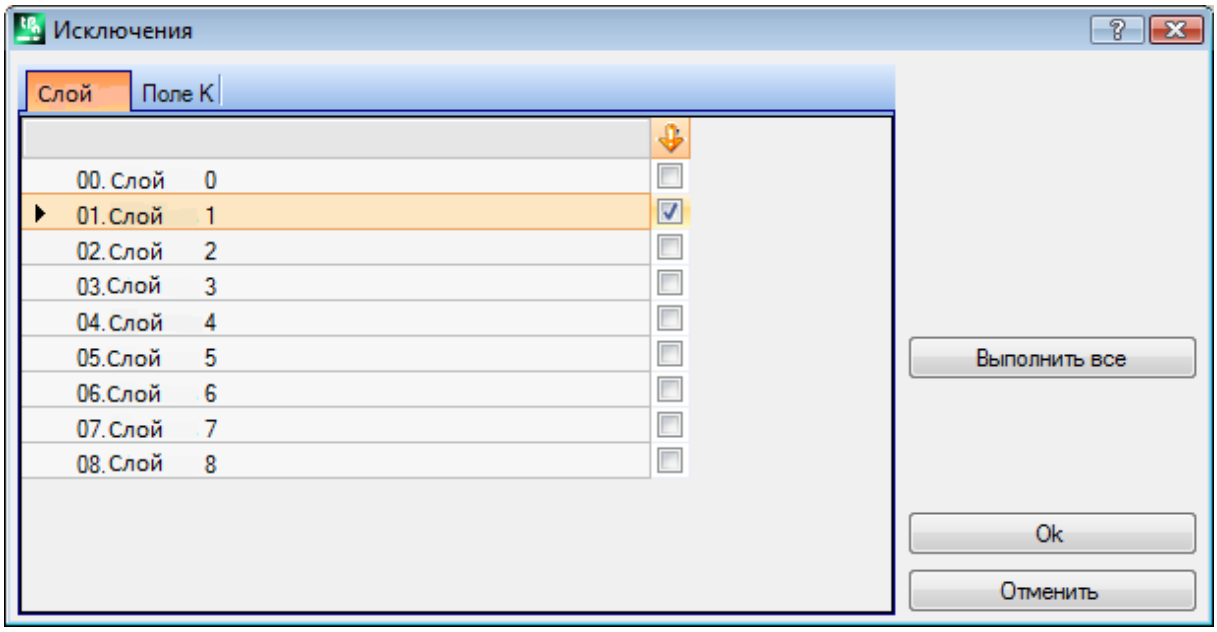
- **[Shift+(левая кнопка мыши)]:** запускает выбор зоны. Заключенные в указанном окне операции добавляются к текущему выбору в таблице. При нажатой также клавише [CTRL] сохраняются предыдущие выборы, в противном случае они обнуляются.
- **[CTRL+(щелчок левой кнопки мыши)]**: переключает состояние выбора операции, ближайшей к положению мыши
- в противном случае: **(щелчок левой кнопкой мыши)** перемещает текущую строку к операции, ближайшей к положению мыши, обнуляя все выборы.

При выходе из страницы "Последовательности" происходит полное обновление графического представления программы.

6.4 Продвинутые задания

Исключения

Эта команда необязательна и доступна в группе **Задать** вкладки **Редактирование**  .
TpaCAD позволяет исключить из выполнения группу операций, определенных общим значением свойства: L (Слой), K, K1 и K2. Исключение эквивалентно дополнительному логическому условию с той большой разницей, что оно не остается записанным в память и архивированным в программе. Заданные здесь исключения применяются к программе с выбором специального вида *Логических условий,* предусмотренного в группе **Виды** вкладки **Отобразить**. Возможность задания исключений в станке определяется приложением управления списками выполнения.
Каждая страница окна доступна только если подключено ее управление.



- **Слой** для задания исключений для значений свойства "L" для каждого слоя приводятся номер и имя присвоенные слою, максимум 16 значений (для последующих значений нельзя задать исключения). Имена слоев можно изменить на странице, которая открывается из меню Применение **Индивидуализировать->Цвета->Слой**). Состояние слоя указано в столбце, заголовком которому служит значок : флажок на ячейке указывает, что Слой отключен (на рисунке - Слой 1)

Первая строка соответствует слою 0 (незаданный слой).
Кнопка **[Выполнить все]** сбрасывает все заданные на странице исключения.
Выбрать кнопку **[Ok]** для задания исключений активной программе так, как они заданы, и прямого подключения вида на логических условиях.

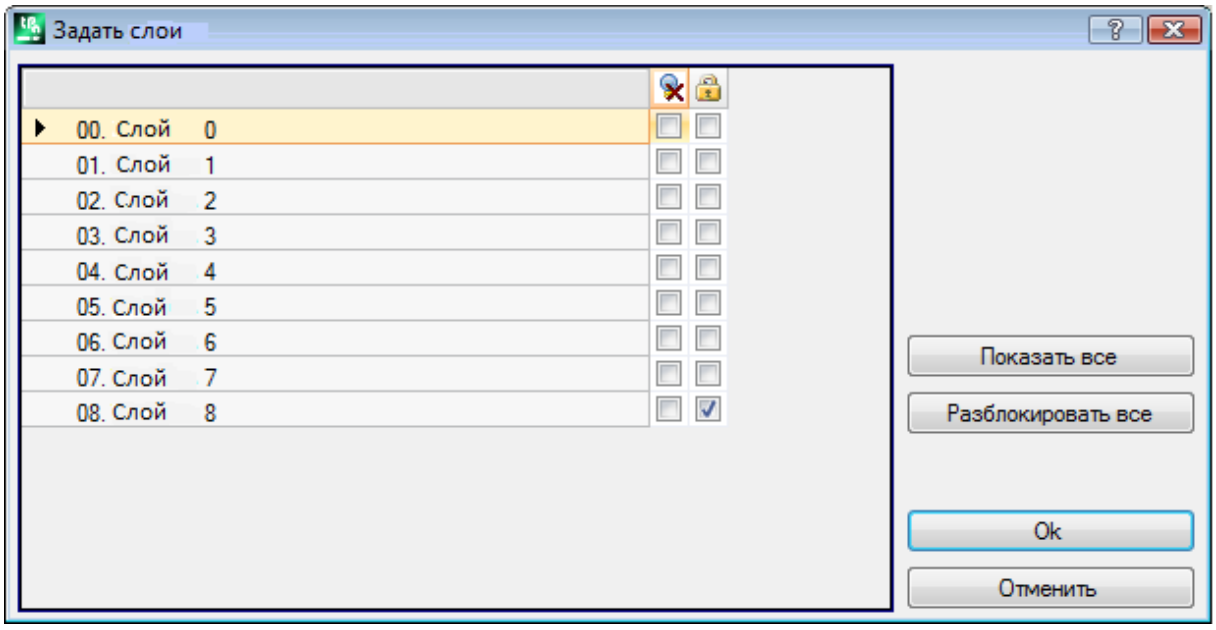
- **Поле K:** для задания исключений для значений свойства "K"
- **Поле K1:** для задания исключений для значений свойства "K1"
- **Поле K2:** для задания исключений для значений свойства "K2"

Также для полей (K, K1, K2) можно назначить исключения максимум для 16 значений.

Слои

Окно задания вызывается из вкладки **Редактирование** в группе **Задать** .

Приводятся слои, управляемые в конфигурации TpaCAD (в примере: до слоя 8), в любом случае, максимум 16 значений.
В этом окне можно задать фильтры изображения и изменения каждого слоя.
Заданные здесь фильтры изображения применяются к программе выбором специального вида *Слои*, предусмотренного в группе **Виды** вкладки **Отобразить**.
Фильтры изменения применяются напрямую: нельзя изменить операции с заблокированным заданным значением слоя.



- **Столбец заголовка:** для каждого слоя приводятся номер и имя, присвоенные слою (может меняться на странице, которая открывается из меню Применение **Индивидуализировать->Цвета->Слой**)
- **:** состояние вида слоя: флажок на клетке указывает, что слой отключен от отображения
- **:** свободное или заблокированное состояние слоя: **:** флажок на ячейке указывает, что слой заблокирован

Первая строка соответствует слою 0 (незаданный слой).
Кнопка **[Показать все]** переключает все уровни в отображенное состояние**.** Кнопка **[Разблокировать все]** переключает все уровни в свободное состояние.

Специальные фильтры

Окно задания вызывается из вкладки **Редактирование** в группе **Задать**

Это необязательная команда.

Приводятся значения Конструкции, Полей (O, K, K1 и K2), Технологии.
Каждая страница окна доступна только если подключено ее управление.
В этом окне можно задать фильтры состояния вида и изменения в порядке полностью аналогичном окну Слоев. Для вкладки Технологии можно задать только фильтры изображения.
Фильтры вида, как они заданы здесь, применяются к программе выбором специального вида Специальные виды, предусмотренного в группе **Виды** вкладки **Отобразить**.
Фильтры изменения применяются напрямую: нельзя, например, изменить операции с заблокированным заданным значением Поля O или конструкции.
Для каждой страницы свойств:

- столбец : состояние вида свойства. Флажок на ячейке указывает, что свойство отключено от отображения
- столбец : состояние свойства свободно/заблокировано. Флажок на ячейке указывает, что свойство заблокировано.

Кнопки **[Показать все]** и **[Разблокировать все]** дезактивируют специальные фильтры, заданные для активной страницы.

Конструкция
Приводятся значения конструкции, сконфигурированные в TpaCAD, и, в любом случае, - максимум для 16 значений. Для каждой конструкции приводятся соответствующий номер и имя.

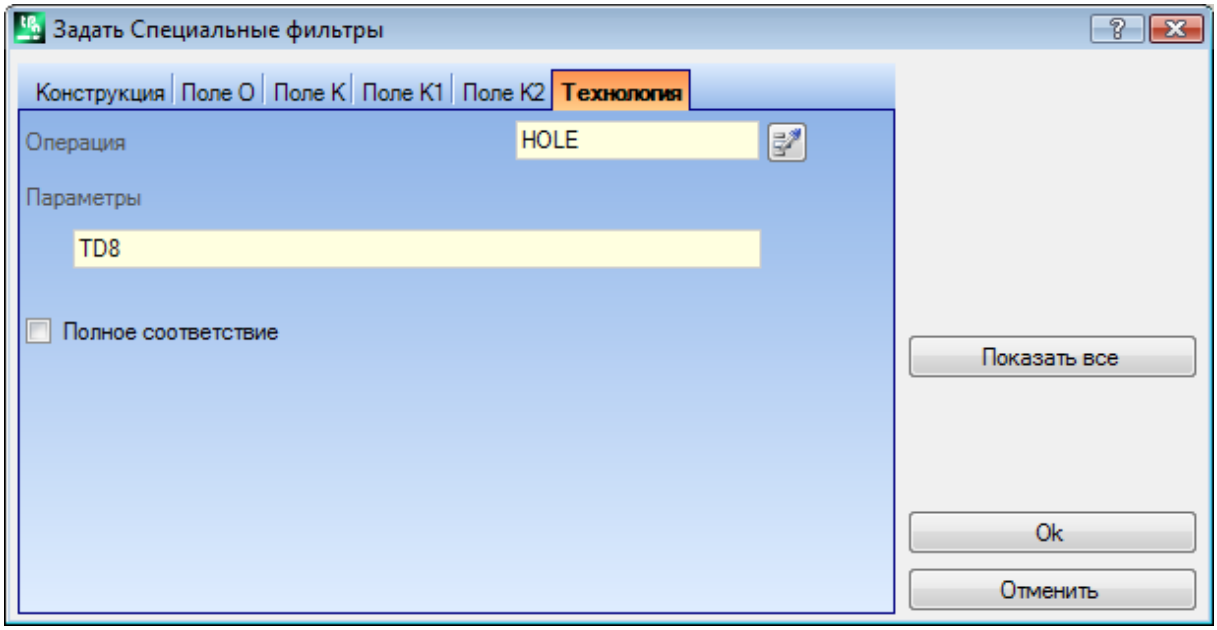
Поле O
Приводятся значения поля O, сконфигурированные в TpaCAD и, в любом случае, - максимум 16 значений. Для каждой конструкции приводятся соответствующий номер и имя. Эта страница отсутствует, если подключено задание поля "O" на отдельных отрезках профиля.

Поле K

Поле K1

Поле K2
Приводятся значения поля (K, K1, K2) сконфигурированные в TpaCAD, и, в любом случае, максимум для 16 значений.

Технология
Позволяет выбирать операции для отображения при задании кода ASCII операции и/или серии параметров этой операции. Можно задать только один фильтр отображения.



В зависимости от данных, заданных в окне в примере, отображаются только операции с кодом ASCII "HOLE" и с параметром TD со значением 8.

В виде грани кнопка  рядом с полем **Операция** позволяет задавать поле на код текущей операции. Интерпретируются только **параметры** с точки зрения технологической валентности (например, станок, группа, инструмент), поэтому в случае операции, относящейся к профилю, оцениваются параметры, относящиеся к операции начала профиля (подготовка или отрезок профиля).

Пункт **Полное соответствие** определяет критерии поиска операций, проверяющих задания. При выборе проверка выполняется также на имеющихся расширенных списках, т. е. на операциях, которые задаются подпрограммами или макропрограммами. При невыборе проверка выполняется только на запрограммированных операциях (список, приведенный в тексте ASCII).
В примере на рисунке:
- если эта опция не выбрана, проверка выполняется на операциях (HOLE, TD8), запрограммированных напрямую
- если эта опция выбрана, проверка выполняется также на операциях (HOLE, TD8), производных из программирования подпрограммы.

Не требуется задавать оба поля. Таким образом, все также на основе примера на рисунке:
- при незаданном поле "Параметры" отображаются только операции с кодом ASCII "HOLE";
- при незаданном поле "Операции" отображаются только операции с параметром TD со значением 8.

Для поля параметров можно определить параметрические задания:
- Параметры ="TD=r27" отображаются только операции с заданием параметра TD на "r27"
- Параметры ="TM2 TD=r27" отображаются только операции с заданием параметра TM на значении 2 и задания параметра TD на "r27"

Сравнение выполняется со строкой задания параметра.

Можно также задать логические условия. Примеры:

- Параметры ="TMR<=3": отображаются только операции со значением параметра TMR меньше или равным 3;
- Параметры="TMR#3", "TMR<>3": отображаются только операции со значением параметра TMR, отличным от 3;
- Параметры ="TMR>3": отображаются только операции со значением параметра TMR больше 3
- Параметры="TMR>3 GR=r4": отображаются только операции со значением параметра TMR больше 3 и заданием параметра GR на "r4".

При задании логических условий (т.е. не условия равенства) **рекомендуется** задавать цифровые настройки. Действительно:

- при числовом задании сравнение осуществляется со значением параметра
- при параметрическом задании сравнение выполняется со строкой задания параметра, с возможностью оценки разницы только между заданными строками.

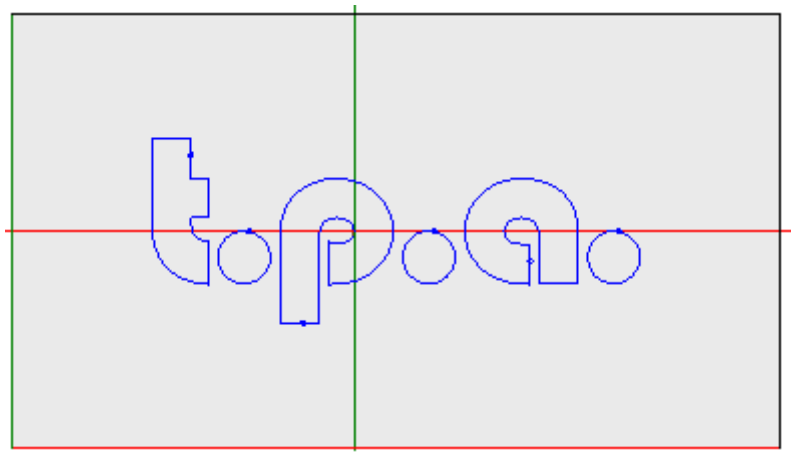
Изменение поля Параметры может вызывать автоматические изменения, обуславливаемые автоматическими проверками. В частности, части, которые распознаются как имена параметров, задаются прописными буквами, а нераспознанные задания отбрасываются.
Если соответствие удовлетворяется на подготовке открытия профиля, оно применяется ко всему профилю.
Кнопка **[Показать все]** дезактивирует специальные фильтры, заданные для Технологии.

7 Грань

7.1 Графическое представление Вида грани

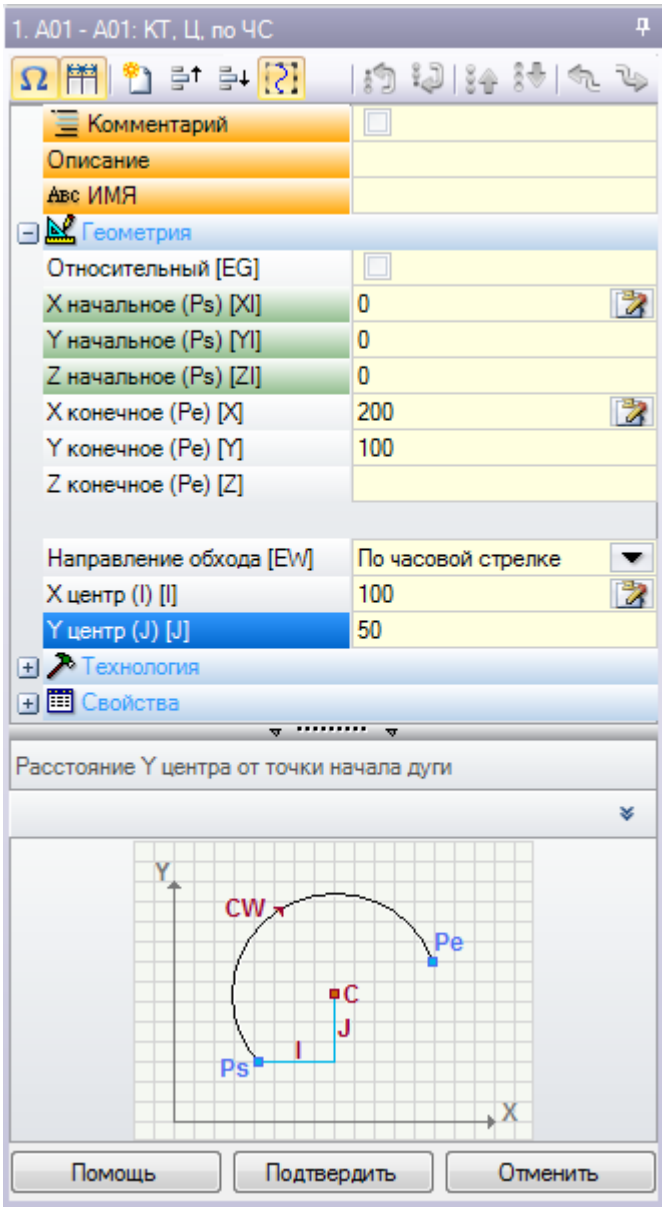
Зона графического представления Вида грани
Зона вида грани может быть представлена

- в трехмерном изображении. Текущая грань и операции грани выделяются другим цветом по сравнению с другими гранями и операциями. Операции других граней представлены серым или их отображение отключено.
- каркасное представление: текущая грань и операции грани выделяются другим цветом по сравнению с другими гранями и операциями
- двухмерное графическое представление на плоскости xy грани. Отображается текущая грань и ее операции.



На рисунке представлен пример графики грани. На всем поле на виде отображается также крестообразный курсор. Указаны начало системы координат и оси грани.

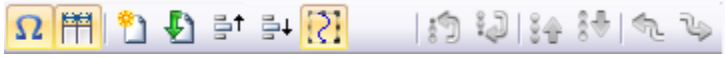
Зона задания операций
Если в программе грани были определены операции, в окне **задания операции** приводятся данные активированной операции.



- **Заголовок:** отображается в форме: Имя ASCII - Наименование операции. Как показано на рисунке "A01 - A01: КТ, Ц, по ЧС"
- **Зона задания параметров и свойств операции:** пункты организованы в список как прямые пункты (Имя, Описание, Комментарий и т.д.) или сгруппированы в пункты (Геометрия, Технология, Свойства). Свойства имеют одинаковое значение для всех операций и приводятся с равномерным расположением для всех операций: некоторые в начале, как прямые пункты (как показано на рисунке: Имя, Описание, Комментарий), другие сгруппированы в последнем узле. Параметры, как правило, могут отличаться для разных операций как по значению, так и по организации. Рядом с описаниями параметров в квадратных скобках могут присутствовать имена ASCII, заданные для этих параметров. Для указания того, что параметр является параметром типа строки, добавляется значок ab

Каждый приведенный в списке узел изначально предлагается закрытым или открытым, как задано для представления каждой операции; последующие представления операции сохраняют для каждого узла последнее выбранное для отображения состояние.

- **Зона "Текстовая помощь":** описание редактируемого параметра
- **Зона "Графическая помощь":** графическая помощь для задания геометрических характеристик операции.



Строка кнопок под названием операции содержит следующие команды:

при подключении окно, помимо описательных сообщений, содержит имена ASCII параметров. Например, как видно на рисунке, параметр **Относительный** имеет имя ASCII, равное [EG]

при активизации это окно делит два столбца заголовка пунктов и программирования на равные части. В противном случае столбец заголовка имеет такие размеры, чтобы полностью отобразить самый длинный текст.

выбор соответствует настройке **Всегда показывать начальную точку на отрезках профиля** в **Индивидуализировать->Среда>Редактирование операции**

дублирует текущую операцию и вставляет дублированную операцию после текущей. Если текущая операция находится на этапе ввода или изменения, вначале задает изменения, а затем выполняет ее копию. Эта команда доступна при подключении в конфигурации TpaCAD. Эта команда не имеет никакого действия, если текущая операция имеет заблокированное изменение.

Перезадать операцию: задает текущую операцию состоянию, соответствующему вводу при прямом выборе из палитры операций. Эта команда не имеет никакого действия, если текущая операция имеет заблокированное изменение.

: перемещают текущую операцию в положение предыдущей или последующей строки. Эти команды доступны при подключении в конфигурации TpaCAD. Эти команды не имеют никакого действия, если текущая операция имеет заблокированное изменение. Функционирование команд зависит от состояния показанной сбоку кнопки:

- если она не выбрана, предыдущие команды касаются только текущей операции, даже если она относится к профилю. Кроме того: текущая операция перемещается на одну позицию в списке, даже если при этом она прерывает профиль до или после;
- при выборе: если текущая операция относится к профилю, предыдущие команды перемещают весь профиль. Кроме того: если до или после имеется профиль, он рассматривается как одно целое и не прерывается в результате перемещения в списке;

: перемещают текущую операцию в первое или последнее положение программного списка

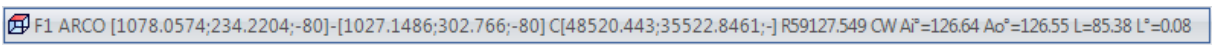
: перемещают текущую операцию к предыдущей или последующей

: перемещают текущую операцию на строку открытия или закрытия текущего профиля (если текущая операция относится к профилю)

: эта команда доступна только если TpaCAD реализует грань-деталь и автоматические грани и подключена только на грани-детали. При выборе перемещение операции, создающей автоматическую грань, перемещает также операции, применимые к этой грани.

: эта команда **Редактировать подпрограмму** доступна только если обработка вызывает подпрограмму. В этом случае подпрограмма открывается в другом экземпляре TpaCAD, гле ее также можно редактировать.

Строка состояния
В строке состояния приводится геометрическая и технологическая информация по текущей операции. Ниже приводится пример составления для дуги:



Информация, приведенная в строке состояния, относится к данной конкретной операции.
Как и в рассмотренном здесь случае дуги, приводится полная геометрия геометрического элемента: начальная и конечная точки, центр, радиус, направление вращения, углы касания на крайних точках, длина развертки дуги (линейная и угловая).

7.2 Порядок открытия

Вид грани для отображения выбирается в строке выбора граней, которая всегда отображается и содержит общий вид, грань-деталь (если реализована), реальные грани и фиктивные грани.



Реальные грани - это только грани, фактически активизированные при конфигурации производителем станка. Фиктивные грани - это грани, заданные на Общем виде, за исключением граней, заданных как вспомогательные построения.

Каждый bitmap строки соответствует одной грани

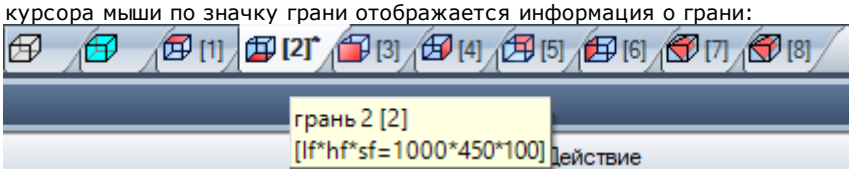
 общий вид

 грань-деталь

 от грани 1 до грани 6

 фиктивная грань. При задании нескольких фиктивных граней bitmap приводится только для первой. Это сделано для снижения до минимума необходимости прокрутки строки для выбора из всех граней в случае задания множества фиктивных граней.

При проведении курсора мыши по значку грани отображается информация о грани:

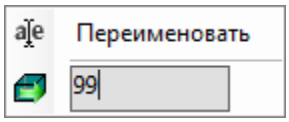


- имя (на рисунке - "грань 2")
- число программных строк грани, при наличии (на рисунке - "(2)")
- размеры (на рисунке: "[lf*hf*sf=1000*450*100]").

Нумерация реальных граней может меняться в разных случаях применения. Можно также задать персонализированную нумерацию.

 Если грань имеет запрограммированные операции, вверху, рядом с номером грани, имеется звездочка.

При щелчке правой кнопкой мыши на значке текущей грани открывается локальное меню:



• **Переименовать**: выбрать для изменения имени грани. Имена граней не записываются в файле языка и поэтому не переводятся. Этот пункт отсутствует в Общем виде;

• **Перейти к грани**: следующая строка меню отображается только если на *строке выбора* граней активны кнопки прокрутки. Это может наблюдаться в случае задания многочисленных фиктивных граней или при очень малом размере по горизонтали зоны экрана, отведенной под строку. На строке представлено:

- поле редактирования, в котором задается номер подключаемой грани (на рисунке: 99);
- кнопка, соответствующая команде **Перейти к грани**, для перехода к виду соответствующей грани. Для выбора *Общего вида*: задать нецифровой символ (или отрицательное значение).

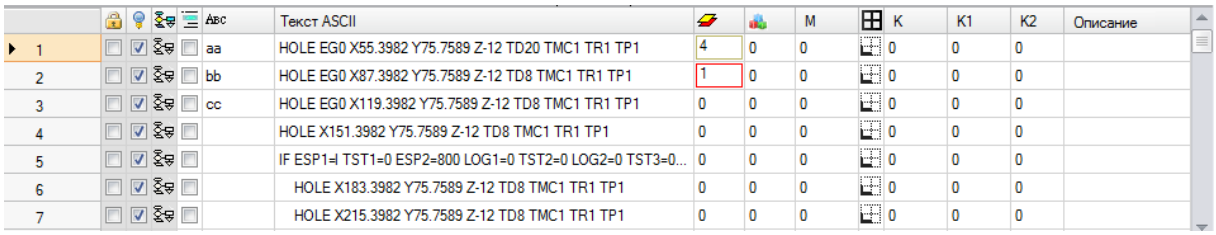
Можно подключить определенную грань также в интерактивном режиме непосредственно из общего вида или из вида грани (отличного от режима "грань-деталь"):
- **выбрать кнопку ALT и щелкнуть левой кнопкой мыши на соответствующей зоне грани; или**
- **двойным щелчком мыши на соответствующей зоне грани.**

В случае графического наложения нескольких граней повторить выбор (кнопка ALT + щелчок или двойной щелчок) до активизации требуемой грани.

7.3 Зона текста ASCII

В зоне текста ASCII отображается программа грани в формате ASCII. Прямые изменения в таблице можно вносить только для следующих заданий:
- **Описание**: без специального подключения, если программная строка изменяема;
- **Свойство "C"** (комментарий): с подключением из конфигурации TpaCAD и только если программная строка изменяема;
- **Свойство "N"** (имя): с подключением из конфигурации TpaCAD и только если программная строка изменяема.

На странице приводится таблица с таким количеством строк, сколько имеется операций на грани. Список упорядочен на основе исходного порядка программирования.



	Текст ASCII			M	K	K1	K2	Описание
1	aa	HOLE EG0 X55.3982 Y75.7589 Z-12 TD20 TMC1 TR1 TP1	4	0	0	0	0	0
2	bb	HOLE EG0 X87.3982 Y75.7589 Z-12 TD8 TMC1 TR1 TP1	1	0	0	0	0	0
3	cc	HOLE EG0 X119.3982 Y75.7589 Z-12 TD8 TMC1 TR1 TP1	0	0	0	0	0	0
4		HOLE X151.3982 Y75.7589 Z-12 TD8 TMC1 TR1 TP1	0	0	0	0	0	0
5		IF ESP1=I TST1=0 ESP2=800 LOG1=0 TST2=0 LOG2=0 TST3=0...	0	0	0	0	0	0
6		HOLE X183.3982 Y75.7589 Z-12 TD8 TMC1 TR1 TP1	0	0	0	0	0	0
7		HOLE X215.3982 Y75.7589 Z-12 TD8 TMC1 TR1 TP1	0	0	0	0	0	0

Заголовок: номер операции по порядку (начиная с 1).

свободное / заблокированное состояние для операции: если ячейка активна (установлен флажок), это означает, что в операции заблокировано поле слоя ("L") или конструкции ("B") или свойств O ("O"). Блокировка слоя (или конструкции или поля O) не позволяет выполнять изменения операций с полем, заданным на заблокированное значение. Фон ячейки другого цвета указывает, что она не соответствует напрямую запрограммированному состоянию, а что она вытекает от определенных оценок. Состояние отображения столбца может меняться в Индивидуализации TpaCAD.

активированный вид: ячейка активирована (установлен флажок), если операция представлена графически. Операция не представлена графически в одной из следующих ситуациях:

- если у нее активирован флажок комментария (свойство "C")
- относится к логическому типу
- активен фильтр отображения свойств (поля: "L", "B", "O") или технологии (рабочий код и/или технологические параметры)
- подключен специальный вид логических условий

И в этом случае фон ячейки другого цвета указывает, что она не соответствует напрямую запрограммированному состоянию. Состояние отображения столбца может меняться в Индивидуализации TpaCAD.
ВНИМАНИЕ: состояние ячейки не зависит от задания поля отображения программы (в строке состояния).

логическое состояние: этот столбец значим при подключении вида логических ограничений. В этом случае ячейка содержит:

- перечеркнутую желтую стрелку, если эта операция не удовлетворяет логические ограничения
- зеленую стрелку, если эта операция удовлетворяет логические ограничения и не является особой логической командой
- знак остановки, если эта операция удовлетворяет логические ограничения и является особой логической командой (ERROR, EXIT). В этом случае: логические условия могут определяться внешними условиями (циклы IF – ELSE – ENDIF), суммированными с запрограммированными условиями непосредственно на операции.
- сигнал предупреждения, в случае логической команды WARNING, удовлетворяющей внешние логические условия (циклы IF – ELSE – ENDIF) и условия, запрограммированные непосредственно на операции.

И в этом случае фон ячейки другого цвета указывает, что она не соответствует напрямую запрограммированному состоянию. Состояние отображения столбца может меняться в Индивидуализации TpaCAD.

Свойство "C" (комментарий)**:** этот столбец не отображается, если это свойство не реализовано. Если эта ячейка выбрана, это означает, что операция находится в списке, но не влияет на программу. В этой связи, когда ссылаются на операцию до или после какой-либо операции, *операции комментария должны считаться исключенными*. Если ячейка выбрана, в окне **задания операции** все остальные поля, свойства и параметры находятся в отключенном состоянии, т. е. не могут быть изменены. Возможности изменения операции возвращаются в нормальное состояние при отключении выбора ячейки. **ВНИМАНИЕ**: изменение возможно в окне **задания операции.** Поле Комментарий может задаваться во всех операциях без исключения. Как уже говорилось, при подключении ячейка может меняться напрямую.

свойство "N" (Имя): это необязательный столбец. Это имя, заданное операции. Это нечисловое поле длиной максимум 16 символов, действительны буквенно-цифровые символы, а первый символ должен быть алфавитным. Например, это свойство используется, например, для применения сложных кодов трансформаций для прямого применения к запрограммированным операциям. Поле Имя может задаваться во всех операциях без исключения.

Текст ASCII: отображает рабочий код (интерпретирует первое поле текста ASCII. Примеры: "G89", "L01", "A01") и параметры в формате ASCII в соответствии с определенным для операции. Этот столбец может содержать отступы, сразу указывающие логическую структуру программы, которая оценивается в циклах IF (IF, ELSE, ENDIF) и FOR (FOR, ENDFOR, если это текст макропрограммы).

При текущей операции, не являющейся комментарием, при щелчке правой клавиши мыши на клетке кнопки ASCII можно открыть контекстуальное меню, которое может помочь, например, в перемещении внутри программы.

В этом списке могут отображаться следующие пункты:
- **Открытие ветви**: перемещает текущую операцию на программную строку вверх, начинающую текущий логический цикл (циклы IF, FOR)
- **Закрытие ветви**: перемещает текущую операцию на программную строку вниз, закрывающую текущий логический цикл (циклы IF, FOR)
- **Выбрать текущую ветвь:**выбирает блок операций, относящихся к той же логической ветви, что и текущая операция.
- **Операция начала/конца профиля**: перемещает текущую операцию на строку начала или конца профиля, к которому относится текущая операция.
- **Выбрать отсюда до начала/конца профиля**: выбирает часть профиля между текущей операцией и началом/ концом профиля, к которому относится текущая операция.

- **Распространить операцию**: текущая операция является сложной (вызов подпрограммы или макропрограммы) или это множественный отрезок, команда открывает окно с определением ее развертки. Каждая строка этого расширенного списка соответствует одной операции, о которой приводится информация о геометрии, технологии и заданные свойства таким же образом, как и информация, приведенная на строке состояния для текущей операции.



цвет операции: содержит цвет, присвоенный этой операции в зависимости от типа (точечная, подготовка, отрезок профиля) или рабочего кода. Задание цвета в зависимости от типа выполняется в Индивидуализации TpaCAD или в базе данных операций. Приведенный здесь цвет не зависит от задания свойств (например, уровня, конструкции). Этот столбец приводится также в расширенном списке операций.
Состояние отображения столбца может меняться в Индивидуализации TpaCAD.



свойство "L" (слой): это необязательный столбец. Отображает значение слоя операции. Если значение слоя 0, означает, что операции не было присвоено никакого уровня. В частности: если задано значение больше 0, операция может быть представлена заданным цветом (как показано цветной рамкой в соответствующей ячейке). **ВНИМАНИЕ**: здесь приводится значение слоя, которое может быть получено также из решения параметрического задания.
Поле "L" не может задаваться на операциях:
- профиля (линии и дуги), для всего профиля действительно значение подготовки
- логические команды (циклы IF, задание переменных...)
- клиентские (точечные, подготовки, логические) или сложные операции, для которых отключено управление ими при конфигурации.



свойство "B" (конструкция): это необязательный столбец. Отображает значение поля Конструкция операции. В частности: если задано значение больше 0, операция может быть представлена заданным цветом (как показано цветной рамкой в соответствующей ячейке). Если операция указана как конструкция, она заполняется, но не исполняется.
Поле "B" не может задаваться на операциях, см. поле "L".



свойство "M": это необязательный столбец. Отображает значение поля M операции.
Поле "M" не может задаваться на операциях, см. поле "L", но есть возможность подключить его управление на отрезках профиля.



свойство "O": это необязательный столбец. Если задано значение больше 0, операция может быть представлена заданным цветом (как показано цветной рамкой в соответствующей ячейке). При максимальном реализованном значении не более 4 TpaCAD может отображать точку отсчета (стороны или ребра) для операции. **ВНИМАНИЕ**: отображение значков, в любом случае, зависит от подключения определенных параметров в конфигурации TpaCAD. Поле "O" не может задаваться на операциях, см. поле "L", но есть возможность подключить его управление на отрезках профиля.



свойство "K": это необязательный столбец. Отображает значение поля K операции. Поле "K" не может задаваться на операциях: см. поле "L".

K1 **свойство "K1"**: это необязательный столбец. Отображает значение поля K1 операции. Поле “K1” не может задаваться на операциях: см. поле “L”.

K2 **свойство "K2"**: это необязательный столбец. Отображает значение поля K2 операции. Поле “K2” не может задаваться на операциях: см. поле “L”.

Описание: описательный текст, который может задаваться в дополнение или в помощь задания операции. Текст может изменяться непосредственно в таблице.

Столбцы свойств, фактически представленные в таблице, зависят от конфигурации TpaCAD.

8 Грань-деталь

8.1 Что это

Грань-деталь - это грань, не имеющая своей геометрической идентификации. Можно сказать, что она представляет деталь в комплексе, включая все характеризующие ее грани.
Традиционно, для грани-детали задаются

- абсолютная система координат детали
- размеры детали (l,h,s)
- идентификационный номер 0.

Программа грани-детали позволяет задавать операции непосредственно на различных гранях, в одном программном списке. Задание операции сохраняет ссылку на свою грань применения, которая задана в дополнительном поле (см. ниже: поле F) в окне задания операции.
Программа, написанная в режиме "грань-деталь", **не может использоваться как подпрограмма.** Поэтому грань-деталь доступна только в случае детали с типом программы.
Программа, написанная на грани-детали, не является суммой программ, написанных отдельно на других гранях, а добавляется к ним.

Если конечный пользователь TpaCAD должен писать подпрограммы, для их последующего применения он должен программировать операции на видах активизированных граней.
Подпрограмму можно вызвать на грани-детали программы:

- поле F, заданное в вызове подпрограммы, указывает, на какой грани подпрограммы она применяется
- выбор грани подпрограммы для вызова осуществляется в уже рассмотренном порядке (пункт "Грани", который имеется, например, среди параметров операции SUB).

Производитель станка может сконфигурировать TpaCAD так, чтобы могла программироваться только грань-деталь. Это действительно только в случае деталей типа программы. В случае деталей типа подпрограммы или макропрограммы всегда реализованы сконфигурированные реальные грани и все запрограммированные фиктивные грани, а грань-деталь отключена.

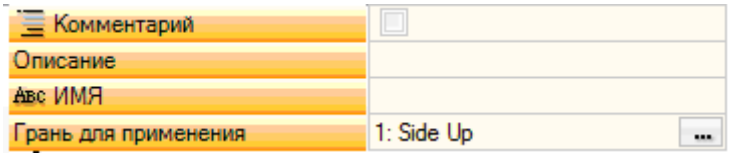
8.2 Порядок открытия

Грань-деталь выбирается в строке выбора грани.



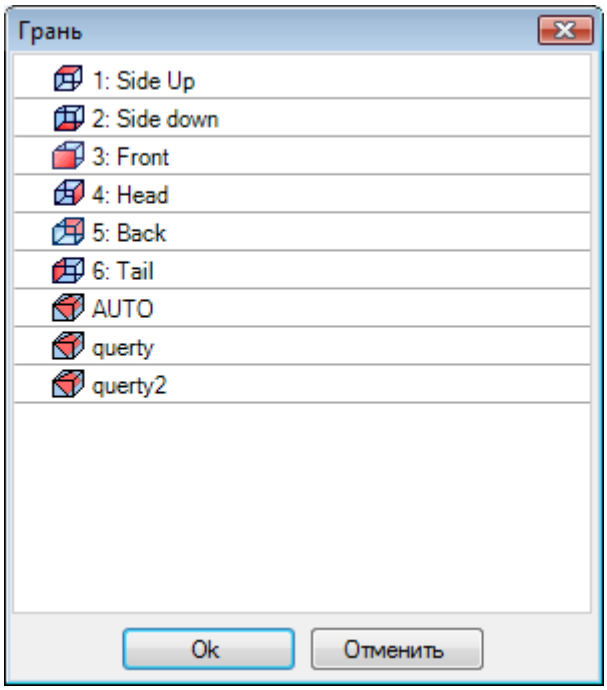
В графической зоне представляется деталь в трехмерном отображении. Не выделена ни одна грань.

8.3 Зона задания операций



Отображаемая информация - такая же, как уже рассматривалось для всех граней, с добавлением граней для применения или **Поле F** для каждой операции программы.

Задание этого поля выполняется только в специальном окне: нажать кнопку  для отображения списка всех реализованных в детали граней, как реальных, так и фиктивных, которые могут выбираться. Не допускается параметрическое программирование. Поле F значимо для всех операций за исключением логических инструкций (IF, ELSE, ENDIF, задание переменных J...).
Список граней может меняться в зависимости от выбранной операции. Например, операция прорези применима, как правило, только на грани 1 (сторона вверху) и 2 (сторона внизу), поэтому в списке отображаются только грани 1 и 2. Если до операции заданы автоматические грани, список выбора для поля F, включает также пункт AUTO (АВТ.), соответствующий применению на автоматической грани.
Более подробно см. код создания автоматической грани.
Рассмотрим пример окна для выбора **Поля F**:



По порядку в нем представлены:
- 6 граней базовой детали (в этом примере они все могут выбираться)
- две фиктивные грани (7 и 8)
- пункт АВТ., соответствующий последней заданной автоматической грани до текущей строки текущей программы
- две строки (последние), соответствующие прямому выбору одной из автоматических граней, заданных ранее с именем.

Из списка исключаются вспомогательные грани (фиктивные или автоматические).

8.4 Зона Текст ASCII

Отображаемая информация - такая же, как уже рассматривалось для программ грани, с добавлением грани для применения или **Поля F** для каждой программной строки.

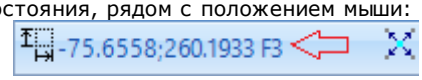
8.5 Поле F

Команда **Поле F** находится в группе **Задать свойство**  вкладки **Редактирование**.

8.6 Представление

В режиме грань-деталь можно подключить любой из видов грани: трехмерный, каркасное представление или двухмерный.
В отличие от вида на других гранях, здесь текущая грань меняется при изменении текущей операции. Кроме того, текущая грань - это также выбранная грань для работы, например, ввода геометрического элемента, точки, линии, дуги в интерактивном режиме. В частности:
- для работы на грани, уже имеющей заданные операции, достаточно щелкнуть рядом с любой операцией грани
- для работы на грани, не имеющей заданных операций: нажать кнопку ALT и щелкнуть левой кнопкой мыши в соответствующей грани зоне, или два раза щелкнуть мышью в зоне грани. В случае графического наложения нескольких граней повторить выбор (кнопка ALT + щелчок или двойной щелчок) до активизации требуемой грани.

Рабочая грань указана в строке состояния, рядом с положением мыши:



8.7 Последовательности исполнения

Одной из характеристик программы грани-детали является непосредственное определение последовательностей выполнения. В функции задание последовательностей операции, запрограммированные на грани-детали, не упорядочиваемые.
Операции грани-детали выполняются до любого другого списка операций на грани, в порядке, как они были запрограммированы. Одним преимуществом грани-детали является возможность надежного группирования операций программы, требующих соблюдения определенной последовательности выполнения.
Типовой пример - создание фиктивных граней в ходе выполнения: необходимо гарантировать, чтобы создание грани выполнялось до обработки грани. В этом случае может оказаться полезным задать операцию резки грани в грани-детали: в любом случае, как бы не изменялась программа, при этом есть гарантия того, что грань будет создана сразу.

9 Операции

9.1 Типы операций

Простые и сложные операции

Ввод операции осуществляется путем выбора операции в группах вкладки **Операции**.

Простые операции включают: одиночные сверления, одиночные подготовки, отрезки линии и/или дуги, логические инструкции.

Точечные и подготовительные операции имеют прямое задание технологии и геометрии.
Основное использование операций подготовки - открытие профиля. В этом случае подготовка предоставляет технологическую информацию, необходимую для выполнения профиля. Подготовка может также использоваться отдельно, т. е. без последующего профиля.
Точечная операция используется, в любом случае, отдельно. Примерами точечных операций являются сверления.

Логические операции характеризуются тем, что они отвечают также определенным требованиям индивидуализации.
Могут выполняться логические операции для:
- измерения детали в реальном времени;
- запрограммированных остановок при обработке детали;
- размещения связей.

Логическим операциям могут присваиваться геометрические поля и технологические поля, которые не интерпретируются TpaCAD. Логические операции не имеют графического представления, не учитываются в расчете габаритов и последующих позиционирований в режиме относительного размещения.
Обычно всегда доступна группа логических операций, помимо операций, заданных на основе конкретных потребностей персонализации. Это операции, которые мы назовем логическими инструкциями: циклы If (IF – ELSE – ENDIF), ERROR, EXIT, задания переменных J.

Сложные операции определяются путем агрегации простых и/или сложных операций. Они включают, например, циклы сверления (подгонка, повторить), многоугольники, прорези.
Как уже говорилось, операция задает параметры и свойства.
Параметр операции может всегда использовать все параметризации, реализованные в параметрическом программировании. В частности:
- размеры детали и/или грани (l, h, s, lf, hf, sf)
- программные переменные (o, v, r)

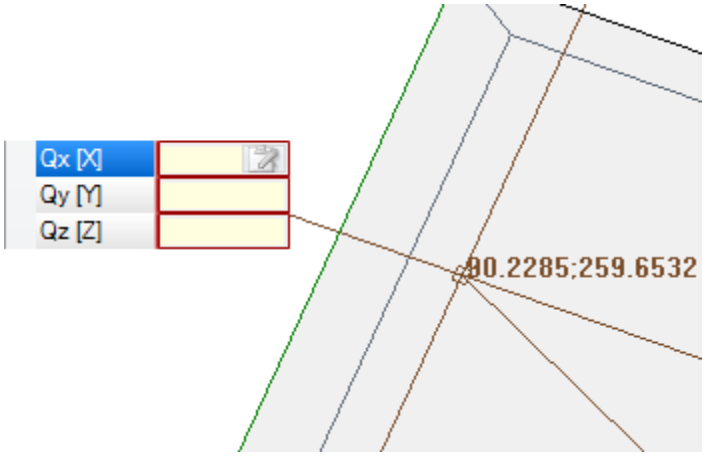
Максимальная длина параметра - 100 символов.
Свойства операции допускают, как правило, только числовое программирование. Для некоторых свойств может быть также подключено параметрическое программирование в таком же порядке, как и для параметров. При параметрическом программировании: ячейка, соответствующая зоне Текста ASCII, содержит его соответствующее числовое значение.

После подтверждения данных введение операции в программу грани осуществляется только если не сигнализируются ситуации ошибки программирования этой операции. В этом случае необходимо устранить неправильные ситуации или отменить ввод. Только при задании операции в тексте макропрограммы можно подтвердить ввод также при ошибке, т. е. предусмотреть вероятность ложной ошибки. Типовым случаем является, например, программирование дуги с заданием геометрии с использованием переменных, заданных локально в цикле FOR. После выполнения ввода программа грани обновляется как в плане графического представления, так и в списке текста ASCII, и введенная операция становится активной операцией.

Направляемое редактирование
Оператору предоставляется помощь при программировании параметров операции. Абсолютно аналогично уже рассмотренному для задания переменных, при нажатии правой клавиши мыши в зоне задания редактируемого параметра открывается контекстуальное меню со следующими пунктами:
- **Подсказать слово:** открывает меню, в котором имеются все функции и аргументы параметрического программирования, сгруппированные в узлы.
- **Быстрая справка**: подсказка аргументов, необходимых для используемой функции
- **Подробная справка**: открывает страницу справки по введенной функции
- **Список переменных**: открывает окно со списком заданных переменных (" r", "o" и "v").

Для параметров геометрического задания точки может также отображаться значок  . Выбрать для считывания соответствующего положения, щелкая мышью непосредственно в графической зоне.
Доступность режима прямого считывания определяется конфигурацией TpaCAD и заданием поля отображения программы (в строке состояния). Считывание в графической зоне всегда подразумевается на двухмерной плоскости xy грани и дополнительно может расширяться на координату глубины. Выделяются некоторые особые случаи:

- значимое задание точки (координаты x и y): в зоне операции выделяются поля, соответствующие двумя координатам. Для принудительного считывания только одной из двух координат подключить блок другой координаты в локальном меню, реализованному в графической зоне (правая клавиша мыши)



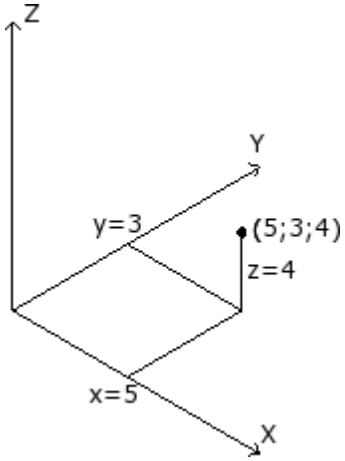
- значимое задание только одной координаты (абсциссы или ординаты): в зоне операции показано только соответствующее поле и горизонтальный или вертикальный курсор показывает его значение.

Локальное меню, реализованное в графической зоне, обеспечивает наличие также команд подключения привязок (к сетке или к объекту), которые обычно реализованы в интерактивных процедурах считывания.

Точка применения

Точка применения операции определяется координатами, заданными в плоскости XY, и координатой Z, перпендикулярной плоскости грани.
Координаты X и Y могут задаваться в прямоугольной или полярной системе координат.

Задание координат в картезианской системе:



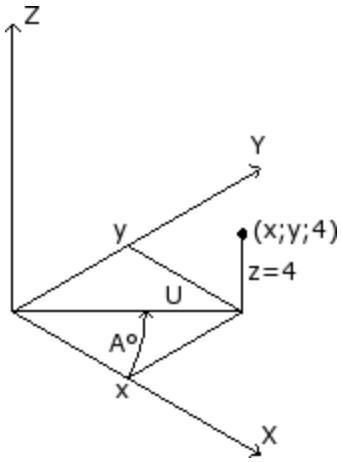
Qx [X]	If/2+50
Qy [Y]	20+32
Qz [Z]	a;4

В прямоугольной системе задаваемые непосредственно координаты могут быть:

- абсолютными из начала координат грани, если не выбрана клетка **Относительный**
- относительными из предыдущей последней запрограммированной позиции, если выбрана клетка **Относительный**

Если точка задана, как показано на рисунке, с координатами (x=5;y=3;z=4), но в относительном режиме с последней запрограммированной позицией в (x=2;y=2;z=2), операция будет иметь свою точку применения в (x=7;y=5;z=6).
Выбор абсолютный/относительный применим ко всем координатам.
При выбранном режиме **Относительный** можно принудительно задать абсолютный режим на одной отдельной координате, указывая перед координатой надпись "a;". Если координата не задана (поле пустое), распространяется значение, заданное для предыдущей операции.

Задание полярных координат:



Геометрия	
Относительный [EG]	
Центр X [I]	0
Центр Y [J]	0
Kz [Z]	-5
Угол (A°) [A]	45
Модуль (U) [U]	100
Технология	
Диаметр [TD]	8
Станок [TMC]	1
Блок [TR]	1

На рисунке показаны полярные координаты. Координата Z остается задаваемой напрямую, как в случае задания координат в картезианской системе.
Положение точки на плоскости XY указывается с указанием ее расстояния от центра и ее угла (в градусах) на плоскости XY относительно оси X.
На рисунке:

- центром является начало координат грани (0;0);
- расстояние U от центра составляет 100;
- угол: 45°.

Выбор абсолютный/относительный теперь применим к координате Z и к координатам (x;y) центра полярной системы.
При выбранном режиме **Относительный** можно принудительно задать абсолютный режим на одной отдельной координате, указывая перед координатой надпись "a;".

Технология

Точечная или подготовительная операция имеет задание технологии, что обуславливает оценку архитектуры установки. Действительно, технологические данные должны определяться в зависимости от инструмента, который будет выполнять операцию, и группы и станка, к которому он относится.

На установке определяются один или несколько станков, на каждом из которых могут работать одна или несколько групп (или головок), которые, в свою очередь, могут раскладываться на устройства: инструменты, электрошпиндели, устройства для смены инструментов. При задании технологии точечной или подготовительной операции используется оснащенный инструмент в положении (шпиндель/электрошпиндель) группы головки станка.
Для каждой группы головки задается максимальная конфигурация устройств, которая зависит от конкретного применения. Каждый станок может иметь каталог инструментов и каталог держателей, каждый держатель может содержать определенное максимальное число инструментов, которое все также зависит от применения.

Общие критерии оценки
Ниже рассматриваются критерии оценивания для программирования инструмента, принятые в TpaCAD, по схеме возможных случаев на основе **приоритета** их оценивания:

Программирование по шпинделю (или электрошпинделю) **и инструменту**

Технология	
Станок [TMC]	1
Блок [TR]	1
Электрошпиндель [EM]	100
Инструмент [T]	2
Тип инструмента [TP]	100

На рисунке заданы поля **Электрошпиндель** со значением 100 и **Инструмент** со значением 2.
Для **Станка** и/или **Группы** обычно необходимо задать значение. Если это поле не задано, по умолчанию используется значение 1.
При щелчке на значке можно выбрать технологические параметры непосредственно из окна представления технологии.

Значение, заданное в поле **Электрошпиндель,** задает положение устройства в группе, а значение, заданное в поле **Инструмент,** определяет инструмент (или держатель) для установки на **Электрошпиндель,** с возможной интерпретацией выбора в списке устройств или типов устройства.
В зависимости от технологии станка значение, заданное в поле **Инструмент**, может определять также держатель инструмента, а в случае держателя, оснащенного несколькими инструментами, может определять использованное положение.
В примере, приведенном на рисунке:

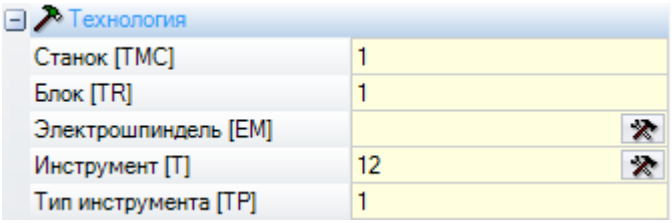
- если с **Электрошпинделем** 100 группы 1 связано устройство смены инструмента, оно оснащается инструментом номер 2;
- в противном случае **Электрошпиндель** 100 должен обязательно оснащаться инструментом номер 2.

Если задано только одно из двух полей (**Инструмент** или **Электрошпиндель**) см. случай программирования по Электрошпинделю, описанный далее.

Можно задать также **Тип инструмента**, позволяющий задать больше характеристик для выбора инструмента. В соответствии с рисунком, операция будет считаться правильной и выполняться только если **Инструмент**=2 **Станка**=1 и **Группа**=1 сконфигурирована с **Типом инструмента**=100.

Поле **Электрошпиндель** может задаваться по умолчанию и не отображаться в окне задания данных операции. Эта ситуация соответствует случаю группы только с одним сконфигурированным электрошпинделем или если выбор позиции в группе не может оснащаться другим образом.

Программирование по шпинделю (или электрошпинделю)



Могут возникнуть следующие случаи:

- имеется и задано только поле **Инструмент**: (на рисунке со значением 12);
- имеется и задано только поле **Электрошпиндель**;
- имеется оба поля, но задано только одно (на рисунке - инструмент со значением 12).

Для **Станка** и/или **Группы** обычно необходимо задать значение. Если это поле не задано, по умолчанию используется значение 1.
Выбор шпинделя осуществляется непосредственно в поле **Инструмент** (или электрошпиндель) с использованием текущего оснащения.
Если шпиндель не оснащен в технологических параметрах, могут возникнуть следующие случаи:

- выбор инструмента по умолчанию
- ситуация ошибки.

Как и в предыдущем случае, обычно можно задать также **Тип инструмента.**

Программирование по диаметру



Задается поле **Диаметр**: здесь - со значением 8, и не задаются значения **Электрошпиндель** и **Инструмент**.
Для полей **Станок**, **Группа** и **Тип инструмента** обычно можно задать одно значение. Если значение не задается, принимается значение по умолчанию.
Критерии выбора инструмента при выполнении программы остаются специфическими для каждого конкретного применения.

При установленном поле **Диаметр** программирование диаметра распознается, если значение, установленное для **Инструмента**, равно нулю (= 0) и если значение, установленное для **Электрошпинделя**, равно нулю или отрицательно (<= 0).
ВНИМАНИЕ: в версиях до [2.4.7] для распознавания программирования по диаметру требовалось, чтобы поле **Инструмент** не устанавливалось.

Программирование по диаметру типично для операций сверления и может определять выполнение нескольких отверстий за один раз в зависимости от доступности заявленных инструментов.

Инструмент по умолчанию
Не задано ни одно из полей **Электрошпиндель, Инструмент**, **Диаметр**.
Можно принудительно задать выбор в полях **Станок** и/или **Группа** и/или **Тип инструмента.**
Критерии выбора инструмента при выполнении программы остаются специфическими для каждого конкретного применения. Программирование по **Инструменту по умолчанию** не всегда является реально рабочим. В этом случае на этапе оптимизации программы подается сигнал ошибки.

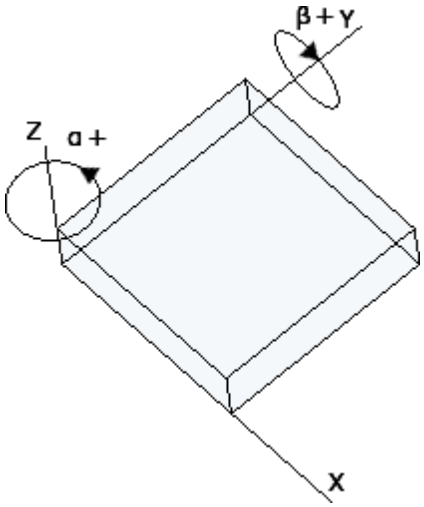
Автоматический инструмент
Выбор автоматического инструмента превалирует над заданиями в полях **Электрошпиндель**, **Инструмент** и **Диаметр**.
В любом случае, можно принудительно задать выбор в полях **Станок** и/или **Группа** и/или **Тип инструмента.**
Критерии выбора инструмента при выполнении программы остаются специфическими для каждого конкретного применения.

Ориентированные геометрии

В нормальных рабочих условиях предусматривается установка инструмента перпендикулярно плоскости xy рабочей грани. Операция setup может также задать ориентацию инструмента относительно плоскости грани. В этом случае говорят об ориентированной подготовке. Ориентацию инструмента определяют следующие поля:

- угол вращения (альфа),
- угол горизонтального вращения (бета).

Эти две оси вращения имеют **абсолютное** программирование **на детали**.
Поля вращения инструмента, если они заданы для операции, в любом случае, являются значимыми, даже если не заданы (в этом случае они принимают значение 0).



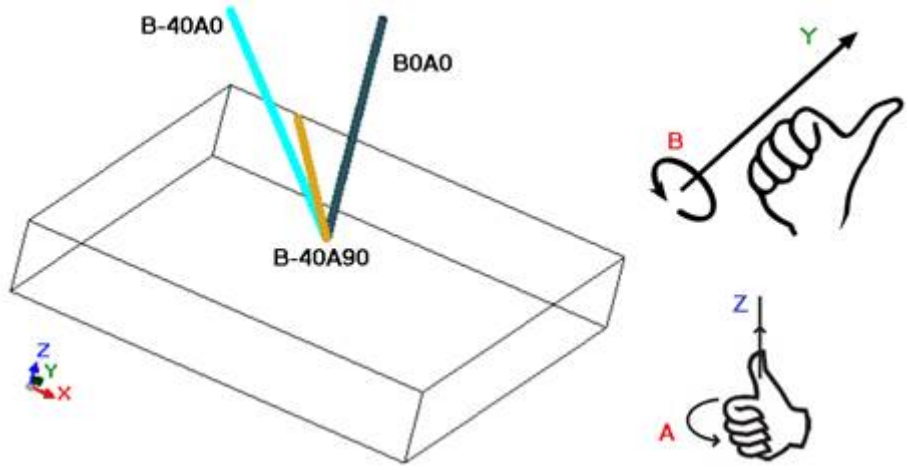
На рисунке представлен общий вид детали и **абсолютная система трех прямоугольных координат**:
- бета вращается вокруг оси Y
- альфа вращается вокруг оси Z.

Рисунок показывает ситуацию по умолчанию. В зависимости от конфигурации TpaCAD:

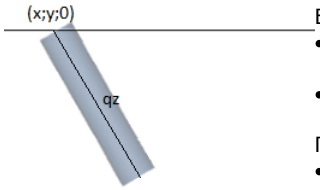
- вращение, уазанное дла оси *бета* может иметь противоположный знак; или
- ось бета может вращаться вокруг оси X.

Рисунок показывает операции setup запрограммированные на грани 1:

a. первая подготовка с (B0;A0)=вертикальное направление
b. вторая подготовка вращает вокруг B против часовой стрелки
c. третья подготовка добавляет вращение по часовой стрелки по A



В операции setup, задающей ориентацию инструмента, координаты XYZ принимают значимое значение по сравнению с неориентируемой подготовкой.
Как правило, можно изменить режим программирования точки применения (координаты на плоскости XY и Z для оси глубины) с заданием параметра **Исх. коорд. Z ортогональна плоскости [DZ].**



Если параметр **Исх. коорд. Z ортогональна плоскости [DZ]** не выбран:
- координаты X и Y программируют точку врезки инструмента на плоскости грани
- координата Z, запрограммированная для глубины, измеряется по ориентированной оси инструмента.

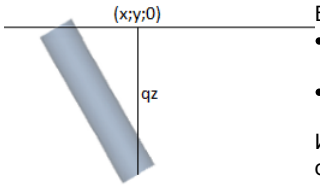
Глубина значима со знаком:
- положительное значение перемещает острие инструмента из запрограммированного положения XY по противоположной результирующей углов (альфа; бета);
- отрицательное значение перемещает острие инструмента из запрограммированного положения XY по результирующей углов (альфа; бета);

При правильно заданных углах (альфа; бета) для позиционирования инструмента на входе в грань:
- положительное значение перемещает инструмент в воздух;
- отрицательное значение перемещает инструмент на операцию в грани.

инструмент входит на плоскость грани на запрограммированных координатах XY с направлением, заданным углами вращения и горизонтального вращения, на величину, равную заданной глубине.

На рисунке показана врезка инструмента в деталь с боковым видом относительно (верхней) плоскости грани.
Дальнейший профиль соблюдает ориентацию, заданную на подготовке.

Этот случай соответствует ситуации по умолчанию, которая применяется к программированию ориентированной подготовки даже если параметр **Исх. коорд. Z ортогональна плоскости [DZ]** не задан для подготовки.



Если параметр **Исх. коорд. Z ортогональна плоскости [DZ]** выбран:
- координаты X и Y программируют точку на плоскости грани, соответствующую конечному положению кончика инструмента;
- Z, запрограммированная для глубины, измеряется вдоль плоскости, ортогональной грани (ось глубины грани).

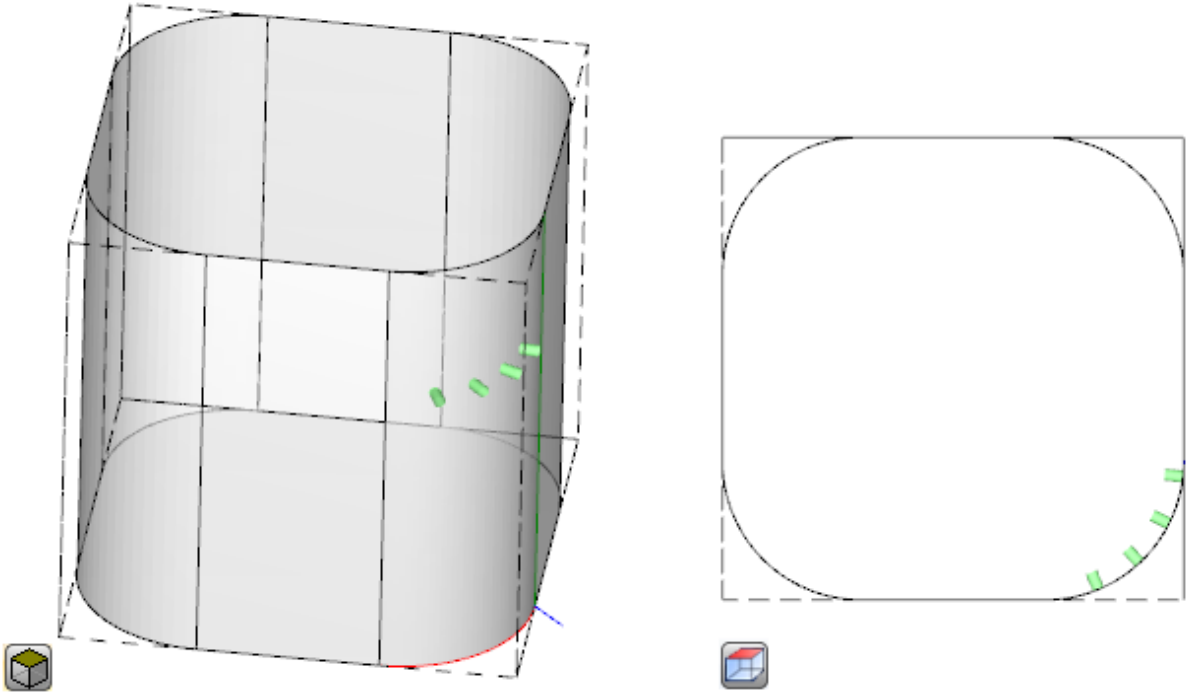
Инструмент входит на плоскость грани, с заданным направлением, в точке P', обеспечивающей соблюдение положения, заданного на трех запрограммированных координатах. Точка P' вычисляется автоматически, а программирование касается кончика инструмента.

Этот режим программирования используется по умолчанию, если подготовка выполняется на закругленной грани или поверхности.

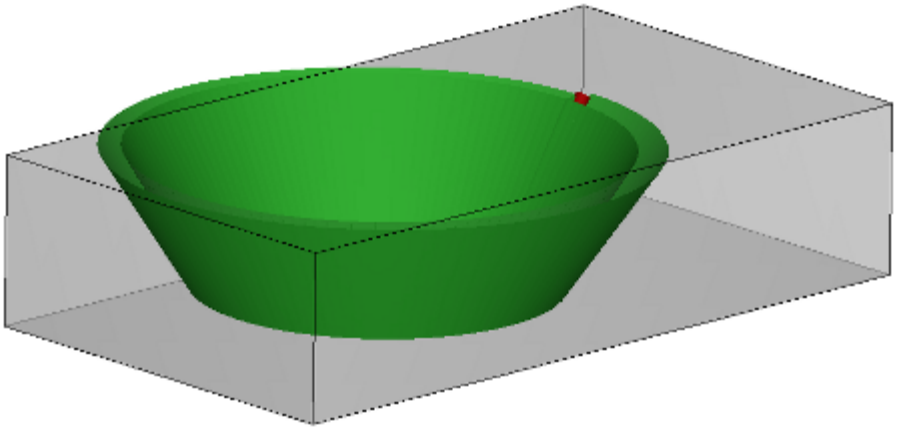
Программирование углов (альфа, бета) может облегчаться за счет использования следующего:

- параметрические формы (**geo[alfa;x;y;z]; geo[betaalfa;x;y;z]**), возвращающие углы (альфа, бета), соответствующие вертикальному направлению грани, но только если она плоская. В случае закругленной грани или поверхности функции вычисляют направление грани с нулевым радиусом закругления;
- параметрические формы (**geo[alfa]; geo[beta]**), возвращающие углы (альфа, бета), соответствующие вертикальному направлению, в определенной точке грани, включая случаи закругленной грани или поверхности;
- выбор **Вертикальное направление**, выбрать для задания вертикального направления грани независимо от ее задания: плоской, закругленной или поверхность. Значения, вычисленные для углов (альфа, бета), не задаются автоматически в соответствующих полях, а определяются для графического представления операции.

На рисунке приводится пример изолированных подготовок, запрограммированных на закругленной грани: слева - трехмерное представление детали, справа - вид сверху. Очевидно, что перпендикулярное направление грани меняется при изменении пропорции по оси закругления.



Программирование ориентированной подготовки может подключать режим **Преследование касательной**, соответствующий запросу сохранения оси инструмента перпендикулярной при выполнении профиля. Этот режим используется для выполнения невертикальных фрезерований (ось инструмента параллельна плоскости XY грани или наклонена, но не перпендикулярна плоскости XY) с необходимостью поддержания оси вращения инструмента перпендикулярной выполняемому профилю.
На рисунке приводится пример фрезерования на верхней грани детали с инструментом, корректирующим наклон вдоль профиля.



Если выбор **Преследование касательной** выявляет ориентацию инструмента перпендикулярно грани, он соответствует запросу сохранения оси инструмента перпендикулярно при выполнении профиля, даже при изменении плоскости грани и аналогично указанному ниже для случая вертикальной подготовки.

Выбор **Преследование касательной** может быть связан с программированием вертикальной подготовки при выполнении на закругленной грани или поверхности. В этих случаях выбор соответствует запросу того, чтобы направление инструмента всегда сохранялось перпендикулярным плоскости XY грани, плоскости, которая теперь, как правило, является переменной. Работая на переменной геометрии грани, можно сказать, что подготовка все равно должна считаться запрограммированной с ориентированной геометрией.
При рассмотрении случая закругленной грани:
• при активном выборе: вдоль профиля, инструмент меняет направление для сохранения перпендикулярности к плоскости грани
• при неактивном выборе: подготовка выполняется с инструментом, перпендикулярным плоскости грани, и это направление сохраняется по всему профилю.

В зависимости от конфигурации TpaCAD и/или выбора в операции setup, интерполяция профиля в **Преследование касательной** может выполняться с интерполяцией по 4 или 5 осям:
- режим интерполяции по 5 осям предполагает, что обе вращающиеся оси могут перемещаться при выполнении операции
- режим интерполяции по 4 осям позиционирует вращение головки на две поворотные оси, но последующие перемещения могут затрагивать только ось (альфа), а положение (бета) остается без изменений.

Возможности интерполяции, помимо инсталлированных функций, зависят от физической конфигурации машины.

Технологический приоритет

Задание может быть доступно для каждого операции, которое интерпретирует технологическое задание, следовательно, пунктуального типа или типа настройки.
Информация интерпретирует положительное целое значение (значение по умолчанию: 0) и доступна для использования в оптимизации, с целью выполнения программы.

Детали интерпретации и использования информации могут различаться в зависимости от конкретного приложения, но общие термины соответствуют необходимости установления критериев взаимного приоритета выполнения между аналогичными операциями.
Основной случай использования касается упорядочения выполнения запрограммированных профилей, чтобы оптимизировать операции смены инструмента. Возможно, что несколько профилей используют одну и ту же технологию, но некоторые должны обрабатываться после всех остальных, независимо от технологии, назначенной для каждого профиля.

Программирование **технологического Приоритета** позволяет решить поставленную задачу, разместив в порядке исполнения:
- сначала все профили с *технологическим Приоритетом = 0*, с или без групп, оптимизированных для инструмента
- ниже профилей с *технологическим Приоритетом = 1* , с или без групп, оптимизированных для инструмента
- ..
- до исчерпания профилей.

Графическое представление

Точечные и подготовительные операции представлены на двухмерном изображении грани кружком диаметром, равным диаметру запрограммированного инструмента; в трехмерном изображении они представлены цилиндром диаметром, равным диаметру запрограммированного инструмента и высотой, равной габариту инструмента по глубине на грани.
В трехмерном изображении в подготовке с ориентированным инструментом инструмент представляется ориентированным по углам вращения и горизонтального вращения.
Операция, запрограммированная с инструментом с несколькими сверлами, отображается одним кружком диаметра, равного диаметру первого сверла головки.

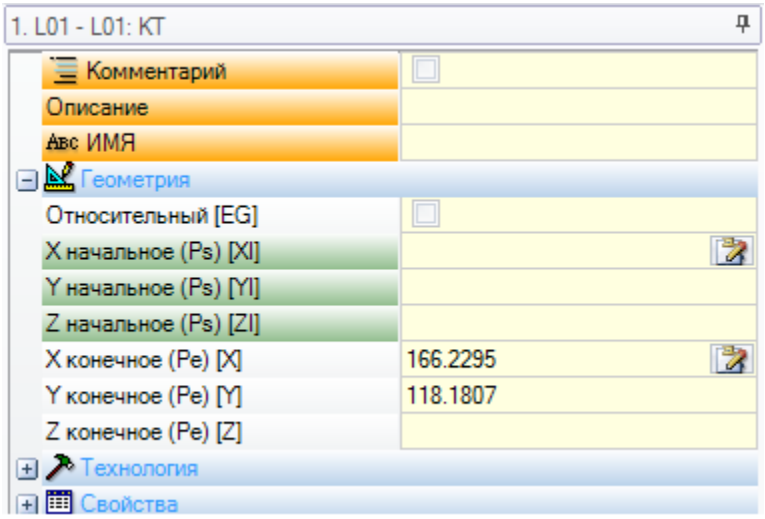
9.2 Профиль

Операции профиля

Элементарные операции, которые могут участвовать в построении профиля, находятся во вкладке **Операции** в различных группах:
- Отдельные отрезки линий: решают линейный сегмент;

- Отдельные дуги: решают дугу в плоскости XY грани;
- Фаска и скругление: решают два линейных сегмента или один линейный сегмент и дугу;
- Многократные дуги: решают одну или несколько дуг;
- Круги: решают круг в плоскости xy грани;
- Дуги в плоскостях xz, yz, xyz: решают дугу в плоскостях xz, yz, xyz;
- Многоугольники: решают, как правило, замкнутую фигуру, соответствующую многоугольнику (прямоугольнику, треугольнику, шестиугольнику) или конусу (овал, эллипс)
- Разметка (см. следующий параграф)

Все доступные в этих группах операции имеют интерпретацию, значимую для программы TpaCAD: каждая решает определению интерпретацию запрограммированной геометрической информации. Как правило, говорят, что каждая из этих операций определяет *отрезок профиля* с общей характеристикой *линейного* отрезка или *дуги*.
На рисунке приводятся параметры геометрического задания отрезка самого простого линейного профиля (L01):



Рассмотрим задание двух крайних точек отрезка:
- Ps (XI;YI;ZI): начальная точка отрезка. Эта точка обычно имеется в каждой операции профиля, но программируется только для задания открытого профиля (т. е. профиля, начинающегося без подготовки). Обычно начальная точка выполнения отрезка профиля определяется автоматически конечной точкой выполнения предыдущего отрезка
- Pe (X;Y;Z): конечная точка отрезка. Эта точка может не программироваться напрямую (на некоторых координатах или только частично), исходя из геометрии программирования каждой операции профиля, и в этом случае определяется автоматически.

Построение профиля

Как правило, профиль определяется непрерывной последовательностью линейных и/или кольцевых отрезков, не обязательно начинающейся операцией setup.
При выполнении программы выбранный инструмент остается занятым от начальной точки до конечной точки без прерывания. TpaCAD распознает задание профиля:
- как единую геометрическую развертку между одним или более отрезками профиля,
- за счет зацепления частей профиля, определенных отдельно (зацепления между отрезками профиля с подпрограммами или макропрограммами),
- как развертка применения подпрограммы или макропрограммы.

Профиль может начинаться:
- операцией setup, задающей комплекс технологии, связанной с выполнением этого профиля
- отрезком профиля и в этом случае он называется также **открытым** профилем. Технология профиля может задаваться позднее, все также на этапе редактора, или непосредственно на этапе обработки программы для ее выполнения.

Профиль считается открытым в одной из следующих двух ситуаций:
- отрезок профиля (дуга или линия) имеет задание даже всего в одном из параметров задания начальной точки отрезка.
- до данного отрезка профиля не задана подготовка или другой отрезок профиля.

Точка применения

Операции профиля имеют точку применения в конечной точке. В случае множественного отрезка точка применения - конечная точка последнего решенного отрезка.

Каждый код профиля решает определенную геометрию на плоскости.
Рассмотрим несколько примеров из возможных ситуаций:

- **L2 [xy(polo, U, A), Zf]:** решает линейный отрезок в пространстве, заданный в двух геометрических компонентах:
  - плоскость XY: линейный отрезок, определенный в полярной системе
  - направление Z: с одним компонентом, перпендикулярным плоскости грани

  Операция решает линейный отрезок в пространстве, в котором каждая ось имеет линейное движение.
  Ось глубины - Z.

- **A4 [xy(P1,Xf,Yf),Zf]:** решает спиральный отрезок, заданный в двух геометрических компонентах:
  - плоскость XY: часть окружности, определенная в прямоугольной системе координат как дуга, заданная тремя точками
  - направление Z: с одним компонентом, перпендикулярным плоскости грани.

  Операция решает спиральный отрезок в пространстве, с осью спирали, перпендикулярной плоскости грани и кольцевой разверткой, заданной на плоскости (XY) грани.
  Ось глубины - Z.

- **A5 [xz(Xf,Zf,centro,rot),Yf]:** решает спиральный отрезок, заданный в двух геометрических компонентах:
  - плоскость XZ: круговой отрезок, определенный в картезианской системе координат как дуга, заданная тремя точками
  - направление Y: с одним компонентом, перпендикулярным плоскости грани XZ.

  Операция решает спиральный отрезок в пространстве, с осью спирали, параллельной плоскости Y грани и кольцевой разверткой, заданной на плоскости (XZ) грани.
  Ось глубины - Y.

- **A9 [xyz(Xf,Yf,Zf,centro,rot)]:** решает дугу в плоскости, с общей ориентацией в пространстве:
  - не задана ось глубины.

Как видно, каждый отрезок глубины может, как правило, напрямую задавать также точку начала отрезка. При этом отрезок напрямую открывает профиль. Если отрезок профиля не задает точку начала, она позиционируется на точку применения операции, заданной ранее.

Программирование углов

Коды профиля часто используют задание углов:
- программирование углов осуществляется в градусах и дробных долях градуса (x.xx °)
- используемое вращение показано на рисунке: от 0° до 360° с вращением против часовой стрелки.

Отрицательные углы проходят плоскость XY, начиная от оси X по часовой стрелке.



Касательные и секущие

Это геометрические элементы, используемые в кодах профиля.

Касательная обозначает прямую, которая задает условие касания запрограммированному отрезку профиля (линия или дуга). Различают:
- касательная в начале: задает касание на начальной точке отрезка
- касательная в конце: задает касание на конечной точке отрезка

Секущая обозначает прямую, которая задает условие принадлежности точки применения (конечная точка отрезка) к этой прямой.
Секущая может также задавать условие касания на конечной точке отрезка.

В случае одного линейного отрезка нет различия между касательной в начале и касательной в конце.

Касательная в начале определяется как:
- **касательная по умолчанию:** если она задана одинаковой с касательной в конце отрезка предыдущего профиля
- **запрограммированная касательная:** требует программирования:



  - угла наклона прямой (a1); или
  - двух точек (P1 и P2) на прямой. Ориентация прямой определяется от P1 к P2.

Программирование угла превалирует над программированием точек.

Касательная в конце определяется как:
- **касательная по умолчанию:** только в случае кругового отрезка, который завершается в точке подготовки профиля: задается одинаковой с касательной запуска первого отрезка профиля;
- **запрограммированная касательная:** требует программирования:



  - угла наклона прямой (a2); или
  - двух точек (P3 и P4) на прямой. Ориентация прямой определяется от P3 к P4.

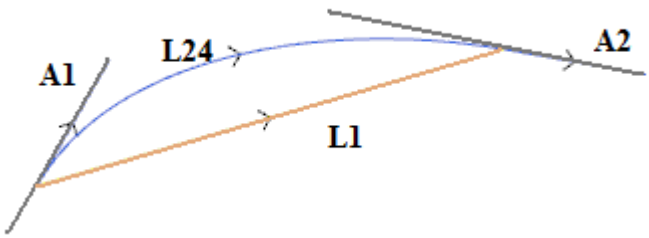
Программирование угла превалирует над программированием точек.

Секущая всегда программируется напрямую. Программирование может выполняться при помощи:
угла (a2) и точки на прямой (P3); или
двух точек (P3 и P4) на прямой. Ориентация прямой определяется от P3 к P4.

Разметка

Термином *Разметка* можно определять:
- определенную операцию, которая может применяться наряду с другими для определения профиля. Код операции - L24;
- профиль, состоящий из элементов типа разметки.

В частности, термин *Разметка* присвоен одной из опций кривых, выбираемых при помощи инструмента Создание кривых Сплайн.
На рисунке показана интерполяция, связанная с кодом L24:



L1 - линейный сегмент, объединяющий крайние точки отрезка (отрезок представлен как член сравнения с полученной кривой)

L24 - кривая, полученная на сегменте L1:
- **A1** - касательная начала кривой L24
- **A2** - касательная конца кривой L24.

Операция L24 задает геометрическую информацию по следующим параметрам:
- начальная точка отрезка (если не продолжает расположенный ранее маршрут)
- конечная точка отрезка
- исходная касательная
- конечная касательная,

Если не задана исходная касательная, принимает:
- предыдущую конечную касательную, на профиле;
- направление между крайними точками отрезка, если конечная касательная не является значимой.

Если не задана конечная касательная, принимает:
- направление между крайними точками отрезка.

Два поля задания касательных могут меняться интерактивно при выборе связанной с полем кнопки.
Операция L24 решает:
- единственный линейный отрезок (L1 на рисунке), если начальное и конечное направления совпадают;
- непрерывную кривую (L24 на рисунке), выборка которой выполнялась последовательностью линейных микроотрезков. Длина отрезков оценивается автоматически и число отрезков выборки обычно большое: в теоретической кривой постоянно меняется закругление и выборка по микроотрезкам, хотя и частая, все равно дает приблизительное решение.

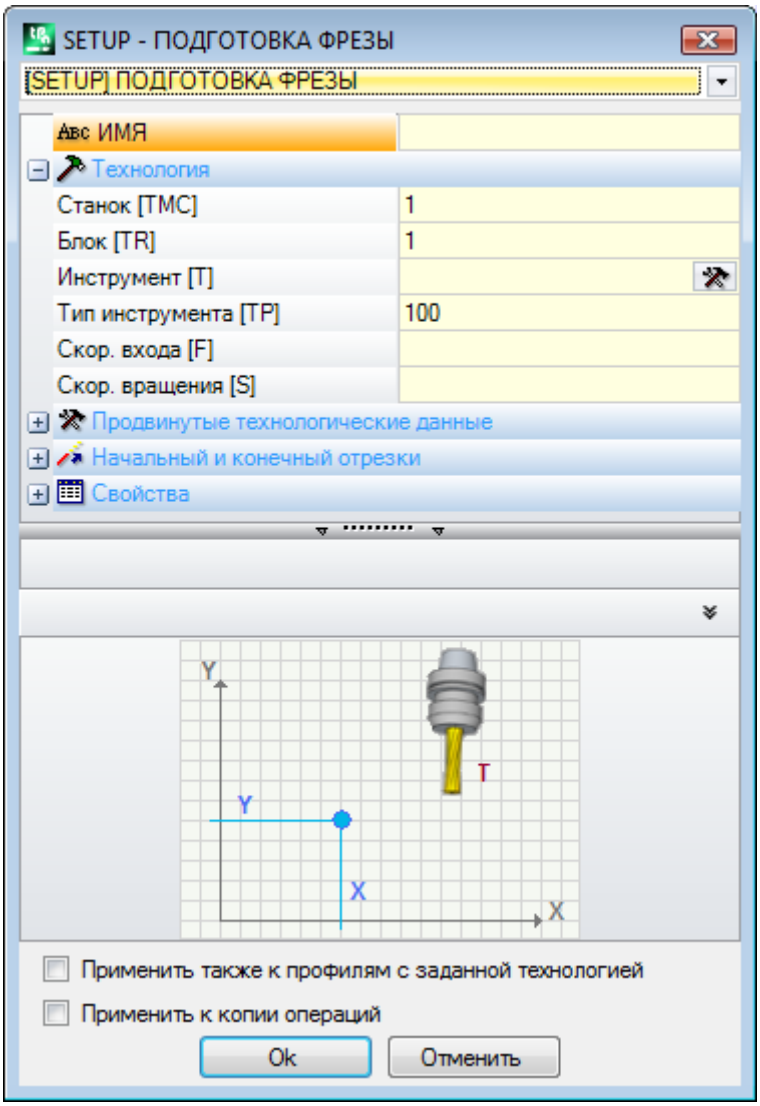
Операция L24 разбивается в списке микроотрезков, которые решают ее только в решении некоторых специальных инструментов.
Применение передовых инструментов к операции L24, кроме того, может ограничиваться, учитывая специфику этой операции, и может отсутствовать для выбора в палитре операций: в этом случае может определяться только применением инструмента *Создание кривых сплайн.*
Несомненно, можно сказать, что не считается нормальным правилом применение продвинутых инструментов манипулирования профилями к кривым *Разметки.*

Задание технологии

Можно задать технологию для профиля, вводя операцию setup в начале этого профиля. Setup профиля не обязательно видима: если, например, профиль полностью или частично определяется в применении подпрограммы (или макропрограммы), подготовка может применяться к созданию подпрограммы внутри. Профиль без подготовки начала называется **открытым** и не имеет четкого задания технологии. В любом случае, на этапе выполнения профиль всегда начинается setup начала и соответствующей заданной технологией. В этом случае речь идет о технологии по умолчанию, как задано в диалоговом окне, открываемом из меню Применение **Индивидуализировать->Технология->Коды по умолчанию**.
Таким образом, возможность управления открытыми профилями является только способом упрощения программирования, но всегда должно быть ясно, какую технологию должна использовать программа на этапе исполнения. Если для выполнения профиля требуется технология, отличная от умолчания, программист должен задать ее напрямую.

Для выполнения открытых профилей, в любом случае, можно выбирать из различных опций, заданных в Конфигурации TpaCAD производителем станка:
- профили, запрограммированные открытыми, могут всегда исключаться из исполнения (например, таким же образом, как операции конструкции).
- Профили, запрограммированные открытыми, могут выполняться обычным образом при задании технологии по умолчанию.
- программирование открытых профилей создает условие ошибки и программа не может выполняться. В этом случае оператор должен напрямую задавать технологию каждого профиля.

Порядок задания технологии профилю
Можно задать данные технологии, вручную изменяя профиль (вставляя и/или изменяя его подготовку) или вызывая команду **Применить подготовку** группы **Изменить профили** во вкладке **Инструменты**  .



В этом окне можно выбрать, прежде всего, задаваемый код подготовки из имеющихся в списке (на рисунке: [ПОДГОТОВКА] ПОДГОТОВКА ФРЕЗЫ) и список параметров обновляется данными, относящимися к выбранной операции. Задать затем технологические параметры и свойства операции и подтвердить кнопкой **[OK]** для применения задания.
Как показано на рисунке, здесь нельзя задать геометрические параметры операции setup.
Технологические параметры операции setup касаются выбора не только Станка, Группы, Электрошпинделя, Инструмента, но и групп параметров, определяющих, в частности:

- режимы Коррекции на инструмент
- режимы Открытия и закрытия профиля

эти аспекты рассматриваются в последующих параграфах и, как это станет очевидным, могут значительно изменять конечную развертку профиля.

В окне инструмента имеется две опции:

- **Применить также к профилям с заданной технологией**: применяет инструмент также к профилям, уже открытым операцией setup. Если эта опция не выбрана, инструмент применяется только к открытым профилям или профилям, которые начинаются с операции ГЕОМЕТРИЧЕСКОЙ ПОДГОТОВКИ (эти профили могут определяться преобразованием формата);
- **Применить к копии операций**: применяет этот инструмент к копии операций и не меняет исходные линии.

Это задание применяется к профилям, имеющим как минимум один выбранный элемент, или к текущему профилю (если нет выборов). В любом случае, применение ограничивается профилями, соблюдающими активированные фильтры изображения: варианты выбора, логические условия, уровни, специальные фильтры. Если этот инструмент применяется непосредственно к исходным профилям (выбранным или

текущим), изменение не может быть применено к операциям в заблокированном состоянии (наличие заблокированного уровня, конструкции или поля O).

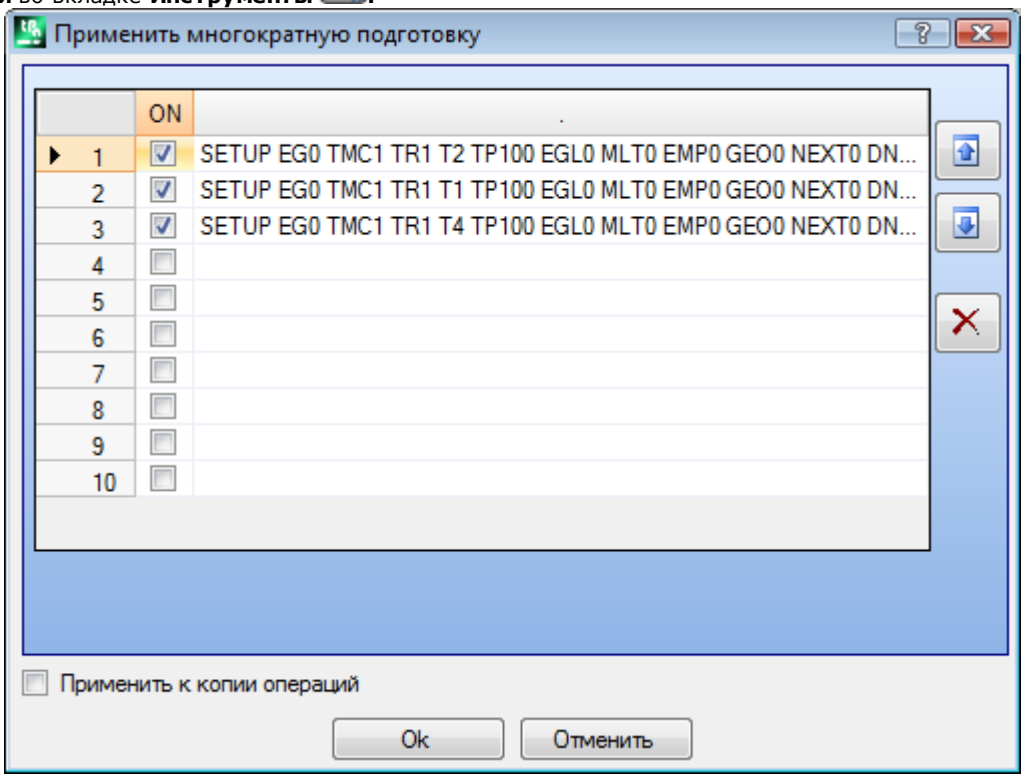
Многократная подготовка

Многократной подготовкой или мультипрофилями называются профили, которым соответствующим образом задается более одной операции подготовки.
При выполнении профиль повторяется столько раз, сколько задано подготовок:
- профиль выполняется первый раз с первой подготовкой и технологией согласно заданному в ней
- профиль выполняется второй раз со второй подготовкой и технологией согласно заданному в ней
- и так далее для всех остальных заданных подготовок.

При этом можно дублировать выполнение профиля без необходимости программировать его несколько раз, в т.ч. при очень разных технологических заданиях. Еще раз подчеркиваем, что технология каждой подготовки определяет в общей сложности:
- выбор (Станок, Группа, Электрошпиндель, Инструмент),
- режимы Коррекции на инструмент
- режимы Открытия и закрытия профиля.

В среде TpaCAD для мультипрофиля видна только первая подготовка, а для последующих применяется узнавание **Зацепление точки**, что делает подготовки прозрачными при выполнении самого профиля. Если, например, запрашивается применение коррекции на инструмент, профиль корректируется на основе заданного в первой подготовке.

Можно задать данные технологии многократных подготовок, вручную изменяя профиль (вставляя и/или изменяя его подготовки) или вызывая команду **Применить многократную подготовку** группы **Изменить профили** во вкладке **Инструменты**  .



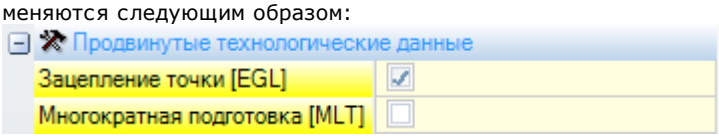
В окне представлена таблица из 10 строк. Каждая строка может задавать подготовку по той же процедуре, которая рассматривалась по применению одиночной подготовки. Для подключения подготовки выбрать соответствующую клетку в столбце ВКЛ: открывается окно выбора (операции setup, технологии). Для изменения уже заданной подготовки два раза щелкнуть мышью (или нажать кнопку F2) на правой клетке соответствующей строки для открытия окна задания; для отключения уже заданной подготовки убрать флажок выбора в соответствующей клетке в столбце ВКЛ.

Порядок строк в таблице отражает порядок, в котором заданы подготовки профиля. Для перемещения подготовки использовать кнопки в правой части таблицы.

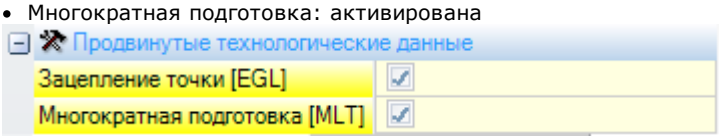
Доступна опция **Применить к копии операций,** которая применяет этот инструмент к копии операций и не меняет исходные линии.

Критерии для выявления профилей, задействованных в применении инструментов, - те же команды **Применить подготовку.**
Теперь задания будут всегда применяться также к профилям с уже заданной технологией: применяет этот инструмент также к профилям, уже открытым с операцией setup или многократными setup.

При задании мультипрофиля страницы **Продвинутые технологические данные** каждой подготовки меняются следующим образом:



Первая подготовка:
- Зацепление точки: не активировано
- Многократная подготовка: активирована



Последующие подготовки:
- Зацепление точки: активировано
- Многократная подготовка: активирована

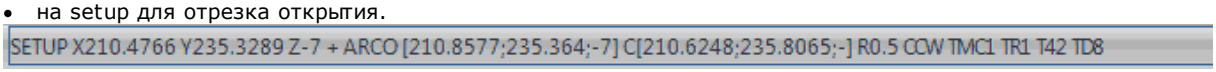
Открыть и закрыть профиль

В операции setup профиля можно задать режим открытия и закрытия профиля.

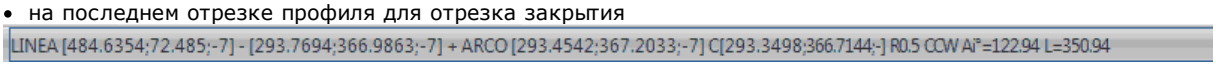
Можно добавить отрезок открытия и закрытия, и указать его тип (линейный сегмент или дуга), длину и изменение глубины отрезка.
- отрезок открытия смещает точку подготовки относительно запрограммированного положения.
- отрезок закрытия выполняется после последнего запрограммированного отрезка профиля.

Отрезки открытия и закрытия не генерируются при наличии изолированных подготовок, всегда отображаются в виде коррекции на инструмент, а отображение без этого активированного вида факультативно. При отображении геометрическая информация о профилях открытия и закрытия приводится в строке состояния:

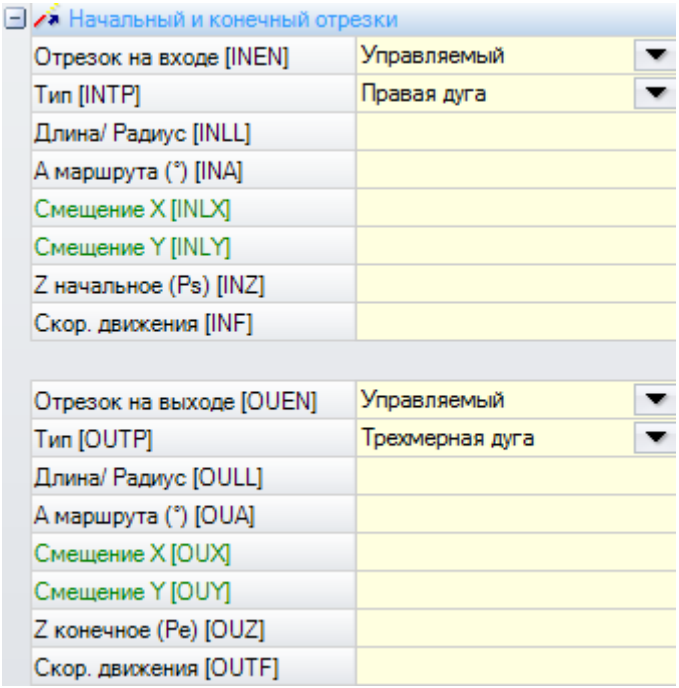
- на setup для отрезка открытия.



- на последнем отрезке профиля для отрезка закрытия



Параметры, задающие отрезки открытия и закрытия, сгруппированы в узле setup:



Для отрезка открытия выбор осуществляется из пяти **типов**:
- **линия:** линейный отрезок, вычисленный на непрерывной касательной
- **левая дуга:** дуга в плоскости xy слева от профиля, вычисленная на непрерывной касательной
- **правая дуга:** дуга в плоскости xy справа от профиля, вычисленная на непрерывной касательной
- **трехмерная дуга:** дуга на ориентированной плоскости, вычисленная на непрерывной касательной
- **приближение:** два линейных отрезка, на которых размещаются перемещение по оси глубины и перемещение на плоскости грани. Вначале выполняется перемещение по оси глубины, а затем - перемещение на плоскости XY грани. Не гарантируется создание на непрерывной касательной: если это условие не соблюдается, выбор типа не применяется, если требуется коррекция на инструмент.

Для отрезка закрытия могут выбираться шесть **типов**:
- **линия, левая дуга, правая дуга, трехмерная дуга:** созданная аналогично отрезку открытия
- **удаление:** два линейных отрезка, на которых размещаются перемещение по оси глубины и перемещение на плоскости грани. Вначале выполняется перемещение на плоскости XY грани, а затем - по оси глубины. Не гарантируется создание на непрерывной касательной: если это условие не соблюдается, выбор типа не применяется, если требуется коррекция на инструмент.
- **перекрытие:** может использоваться только в случае профиля, который завершается в точке подготовки (замкнутый профиль), покрывает часть первого отрезка профиля. Не гарантируется создание на непрерывной касательной: если это условие не соблюдается, выбор типа не применяется, если требуется коррекция на инструмент.

Для типа **Линия** могут задаваться следующие значения:
- **Длина/Радиус:** длина отрезка в плоскости грани. Минимальное задаваемое значение - 50*эпсилон координат. Если активированы как отрезок на входе, так и отрезок на выходе, но для последнего не было определено значение, заданное значение распространяется от входного отрезка до выходного.

Для типов **Дуга (левая, правая, трехмерная)** могут задаваться следующие значения:
- **Длина/Радиус:** радиус дуги. Минимальное задаваемое значение - 50*эпсилон координат. Заданное значение, если активизированы как начальный, так и конечный отрезок, но для конечного отрезка не было задано никакого значения, распространяется с начального отрезка до конечного.
- **У пройденный (°):** угловая амплитуда дуги. Если значение не задается, принимает значение по умолчанию 45°. Минимальное значение равно 1°, максимальное - 270°, если дуга покоится на плоскости xy, в противном случае максимальное значение - 90°. Если активированы как отрезок на входе, так и отрезок на выходе, но для последнего не было определено значение, заданное значение распространяется от входного отрезка до выходного.

Для типов **Приближение / Удаление** могут задаваться следующие значения:
- **Смещение X, Смещение Y:** задают перемещение для двух осей координат. Заданные значения суммируются с соответствующими координатами подготовки или конечной точки
- **Длина/Радиус:** длина отрезка на плоскости грани, используемая, если два предыдущих значения оказываются равными нулю (оба значения меньше 10*эпсилон координат). В этом случае отрезок вычисляется на непрерывной касательной. Минимальное задаваемое значение - 10*эпсилон координат.

Если активированы как отрезок на входе, так и отрезок на выходе, но для последнего не было определено значение, заданное значение распространяется от входного отрезка до выходного.

Для типа **Перекрытие** могут задаваться следующие значения:

- **Длина/Радиус:** длина отрезка в плоскости грани. Если значение не задано: использует длину начального отрезка профиля.
- **Скорость движения:** задает скорость интерполяции на отрезках. Если на входном отрезке значение не задано, используется скорость, заданная на первом отрезке профиля. Если на выходном отрезке значение не задано, используется скорость, заданная на последнем отрезке профиля.

Для входного отрезка:

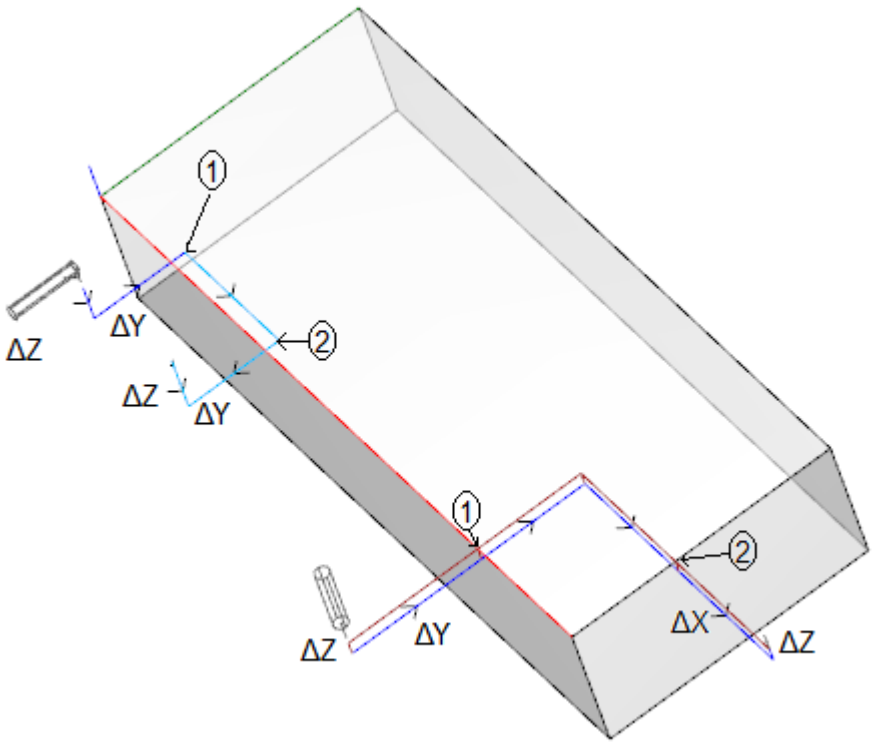
- **Z начальная:** задает начальную глубину отрезка. Конечная глубина отрезка - это глубина, заданная для подготовки. Она имеет абсолютное программирование и если значение не задано, принимается значение, заданное в поле **Qz** (глубина, заданная при подготовке). Если выбранный тип - **Трехмерная дуга,** требуются некоторые уточнения. Прежде всего, геометрия отрезка зависит от начального отрезка профиля. Если начальный отрезок представляет собой:
  - дугу в плоскости xy выполняется дуга на плоскости xyz
  - дугу в плоскости xz выполняется дуга на плоскости xz
  - дугу в плоскости yz выполняется дуга на плоскости yz
  - линейный отрезок выполняется дуга на плоскости xyz

  Значение, заданное для начальной Z, не может, как правило, применяться к начальной точке дуги, так как оно определяется заданной угловой амплитудой.
  Начальная Z значима только если радиус дуги не задан и поэтому выводится из значения изменения между начальной Z и глубиной, заданной для подготовки. Знак колебания между начальной Z и глубиной, заданной для подготовки, определяет решение результирующей дуги так, чтобы войти с заданного направления. Если начальная Z не задана, можно выделить следующие случаи:
  1. если профиль начинается с дуги, для входной дуги задается направление вращение, противоположное направлению первой дуги профиля.
  2. если профиль начинается с линейного отрезка, для входная дуги задается направление входа координаты воздуха

Для выходного отрезка:

- **Z конечная:** задает конечную глубину отрезка. Начальная глубина выходного отрезка - это конечная глубина, заданная для профиля. Она имеет абсолютное программирование и если значение не задано, принимается значение, заданное конечной глубине профиля. Если выбранный тип - **Трехмерная дуга:** для определения геометрии отрезка и *Z конечная* действительны положения, аналогичные изложенным для входного отрезка.

Тип отрезка **Приближение/Удаление** используется в случаях, когда необходимо проверить движения, которые зацепляют/ приближают или отцепляют/удаляют инструмент от детали. На рисунке показаны две типовые ситуации:



Оба профиля программируются с верхней грани:

- в одном случае, левый профиль, инструмент ориентируется по горизонтали так, чтобы входить перпендикулярно боковой грани
- в другом случае, правый профиль, инструмент вертикален по отношению к грани.

Точки, обозначенные как **1**, соответствуют положению, запрограммированному для подготовки профилей.
Точки, обозначенные как **2**, соответствуют положению, запрограммированному для конца профиля.
В обоих случаях подготовка программирует входной отрезок с приближением, а выходной - с удалением.

Левый профиль может соответствовать операции простой выемки на боковой грани детали: начальный и конечный отрезки выполняют контролируемые зацепление и расцепление на габаритах детали и рабочей группы.
Правый профиль может соответствовать фрезерованию угла с его возможной резкой при выполнении с превышением глубины детали. Начальный и конечный отрезки позволяют выполнять правильное зацепление и расцепление инструмента за пределами габарита детали, в этом случае - также с контролем позиционирования по оси глубины.

Зацепить профили

Особым аспектом при определении профиля является возможность его зацепления. Речь идет об опции **Зацепление точки,** доступной как параметр операций подготовки и сложных кодов.
Зацепление к точке означает **постоянное** применение относительного программирования нулевых смещений. Кроме того:
- если ранее задан зацепляемый элемент профиля (подготовка, дуга, линия, подпрограмма, завершающая свою развертку элементом профиля)
- и если текущая операция - это операция подготовки или сложный код, начинающий свою развертку элементом профиля, предыдущий профиль продолжает последующий профиль без выполнения какой-либо промежуточной подготовки. В этом случае говорят о **зацеплении между профилями.**

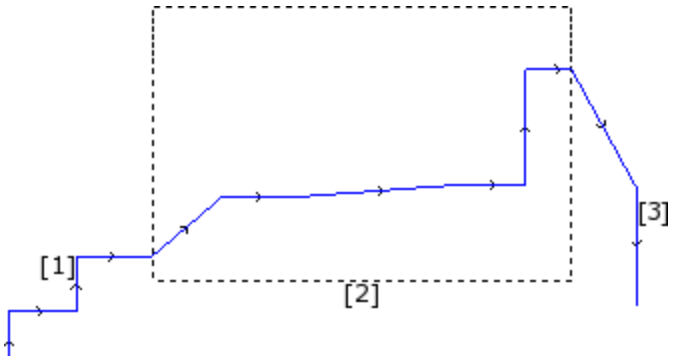
Профиль, построенный с зацеплениями, - это полноценный единый профиль, технология профиля задается операцией setup, расположенной на открытии, до всех частей профиля, расположенных в зацеплении. Если начальная seup не задана, и в этом случае мы имеем открытый профиль.

Постараемся получше объяснить значение выражения "зацепляемый профиль" или, скорее, в каких ситуациях сложный код (подпрограмма или макропрограмма) становится незацепляемым.
Первый случай соответствует сложному коду, для которого выражено исключение в базе данных операций. Типовой пример касается кодов LAMA, выполнение которых позволяет интерпретировать не любой профиль, а только линейный отрезок.
Программирование конечной точки применения (см. параграф: Операции->Подпрограмма->Позиционировать подпрограмму->Конечная точка применения) исключает возможность зацепления после подпрограммы.

Простые профили

При помощи механизма зацепления точки можно продолжить профиль частями, заданными применением подпрограмм или макрокоманд. Однако это не значит, что это всегда позволит получать профиль без какого-либо прерывания, где выбранный для выполнения инструмент остается занятым от начальной точки профиля до конечной точки профиля.

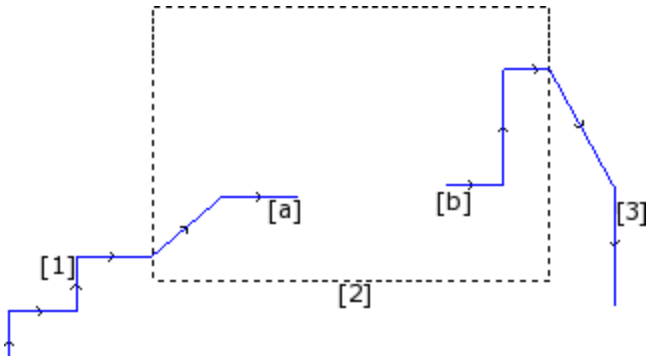
Рассмотрим **первый пример**:



Показанный на рисунке профиль состоит из 3 частей:
- **[1]** первая начальная (слева) состоит из линейных отрезков (не важно, открытый это профиль или нет)
- **[2]** центральная часть показана заключенной в прямоугольник и в этом случае она получена применением подпрограммы (с зацеплением точки)
- **[3]** конечная часть (справа) состоит из линейных отрезков и завершает профиль.

Можно утверждать, что был построен профиль. Инструмент выполнения остается врезанным от точки начала профиля до конечной точки без прерывания.

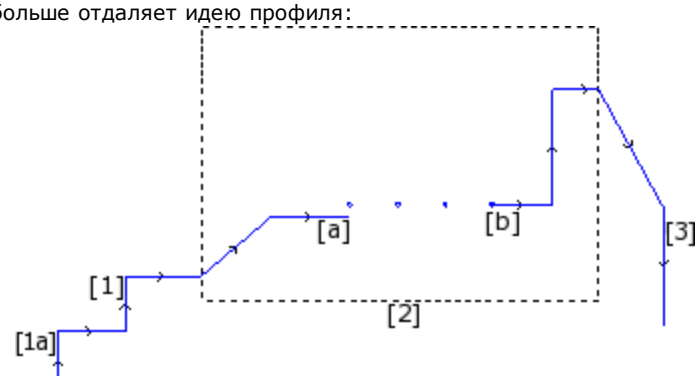
Теперь рассмотрим **второй пример**:



представление подобно предыдущему, различие состоит в том, что центральная часть профиля имеет прерывание.
Еще можно утверждать, что был построен профиль?
на этапе выполнения имеется два отдельных профиля:

- первый профиль выполняет первую часть **[1]** и продолжает до точки **(a)**, показанной на рисунке
- второй профиль начинается в точке **(b)**, показанной на рисунке, и продолжается до завершения конечной части **[3]**.

Третий пример еще больше отдаляет идею профиля:



подпрограмма, обозначенная **[2]**, теперь выполняет:

- в начальной части: профиль (который предварительно зацепляется с **[1]**)
- в центральной части: четыре операции сверления
- в конечной части: профиль (который затем зацепляется с **[3]**)

Следуя логике, определение профиля должно было применяться только к первому из трех рассмотренных случаев.
Однако имеются специальные функции профиля, для которых три рассмотренные выше случая не имеют никакого различия. Если, например, нам необходимо применить инструмент профиля, задающий определенную технологию профилю, начинающемуся в (**1a**), может оказаться полезным, чтобы сам инструмент рассматривал совокупность операций как один профиль, не учитывая, как был определен блок **[2]**: в этом случае речь идет о профиле, который, в любом случае, определен или непростым, или расширенным.
В первом из трех рассмотренных случаев определяется простой профиль: действительно, блок [2] может быть уподоблен элементу профиля.
Таким образом, профиль является простым, если он образован из простых элементов профиля (линейных отрезков или дуг) и/или из сложных кодов (подпрограмм или макрокоманд), которые могут уподобляться простым элементам профиля.

Коррекция на инструмент

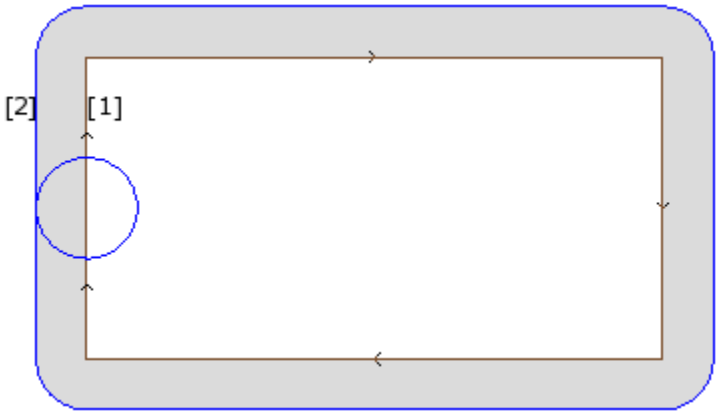
Запрос коррекции на инструмент подключает механизм автоматического смещения запрограммированных траекторий (профилей) так, чтобы учитывать реальный габарит инструмента, выполняющего эту траекторию.

Команда **Коррекция на инструмент** для подключения или отключения коррекции на инструмент находится в группе **Виды** вкладки **Показать**.
Коррекция на инструмент применяется в плоскости xy и не может применяться к дугам, заданным на плоскости, отличной от xy, если:

- первоначальная дуга представляет собой круг или дуга инвертирует направление оси x или y
- скорректированные дуги определяют решение скругления или пересечения внутри отрезка.

Коррекция на инструмент применяется также к профилям конструкции.

В примере ниже показано простое применение коррекции на инструмент:



(1) запрограммированный профиль:

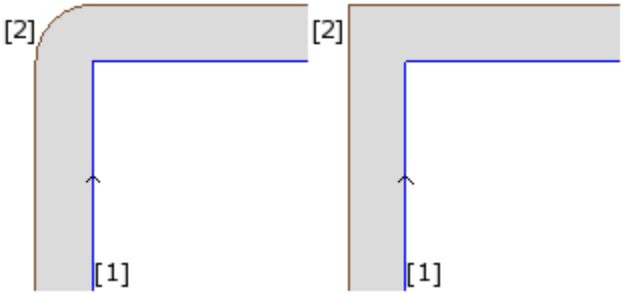
- прямоугольник, проходимый по часовой стрелке;
- кружок на левой вертикальной стороне прямоугольника указывает габарит рабочего инструмента;

(2) профиль, полученный при коррекции на инструмент:

- он находится снаружи запрограммированного профиля и проходится в том же направлении (по часовой стрелке);
- расстояние между двумя профилями равно радиусу инструмента.

При выполнении профиля на станке внутренний прямоугольник будет иметь размеры, с которыми он был вычерчен: исходя из требуемой коррекции, при этом инструмент работает за пределами запрограммированной траектории.
Если необходимо соблюдать наружные размеры прямоугольника, требуемая коррекция должна быть внутри прямоугольника.

Рассмотрим деталь ребра прямоугольника в этом примере:

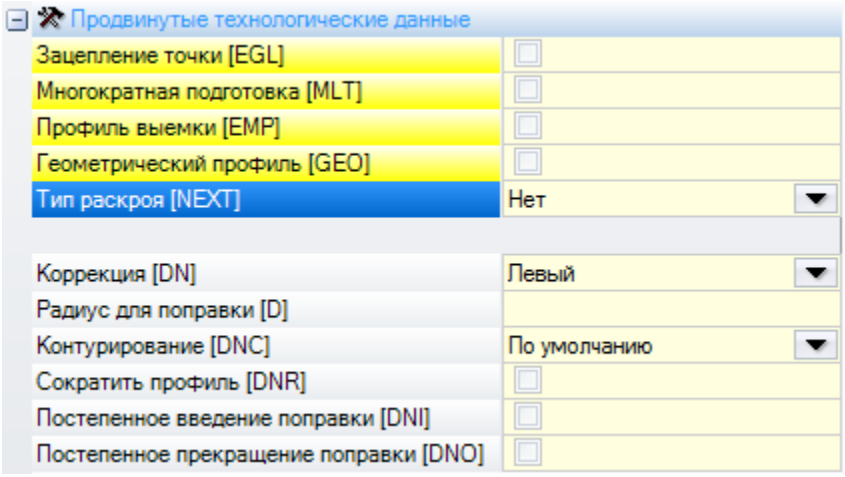


На рисунке слева скорректированный профиль перемещается вокруг изначального ребра с дугой радиусом, равным радиусу инструмента; на рисунке справа скорректированный профиль продолжается до точки пересечения за пределами скорректированных линейных отрезков.
Первый случай соответствует случаю применения режима коррекции с вставкой **скруглений**.
Второй случай соответствует случаю применения режима коррекции с вставкой **пересечений** (который называется также "коррекция с **контурированием**").
Сторона коррекции определяется в направлении прохождения запрограммированного профиля. В примере на рисунке:

- левая сторона соответствует коррекции за пределами прямоугольника
- правая сторона соответствует коррекции внутри прямоугольника.

Задание радиуса коррекции, отличного от радиуса инструмента, позволяет увеличить или уменьшить коррекцию по умолчанию. Минимальное признаваемое значение соответствует эпсилон разрешения, заданного в конфигурации производителем станка. Задание значения меньше эпсилон игнорируется.

Параметры для выполнения коррекции на инструмент задаются на уровне технологии профиля и подготовки и могут отображаться только частично, в зависимости от определенного в конфигурации TpaCAD.



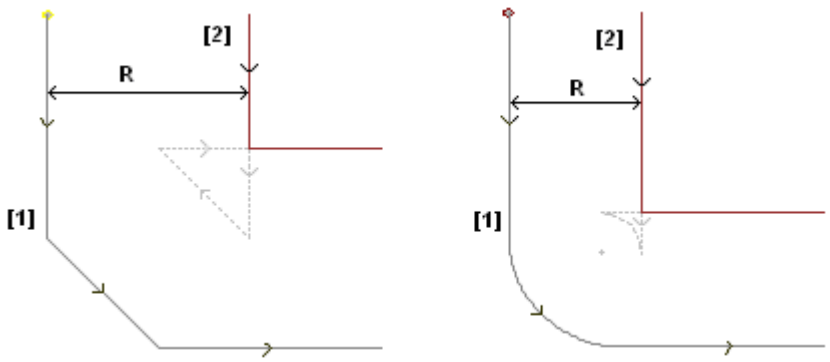
Эти параметры сгруппированы в пункте **Продвинутые технологические данные**:

- **Коррекция**: активизирует коррекцию с прямым выбором стороны коррекции. В списке имеется три пункта:
  - **выкл** отключает коррекцию
  - **левый** подключает коррекцию на левой стороне профиля
  - **правый** подключает коррекцию на правой стороне профиля
- **Радиус коррекции**: задает радиус коррекции, если требуется отличным от радиуса инструмента. В конфигурации TpaCAD можно определить другую интерпретацию значения.

- Типичная конфигурация распознает:

✓ задание радиуса коррекции, в случае программирования без начального знака. Примеры: “5”, “r4”, “prfi[12]/2”
✓ изменение к коррекции, которая будет добавлена к радиусу, объявленному для инструмента, в случае программирования, которое начинается со знака +/-. Примеры: “+2”, “-2”, “+r4”.

- **Контурирование**: активизирует режим коррекции на ребрах. В списке имеется три пункта:
  - **умолчание** активирует режим, заданный по умолчанию (в конфигурации TpaCAD);
  - **Скругления** активизирует коррекцию с введением скруглений
  - **Ребра** активизирует коррекцию на поиске пересечений
- **Сократить профиль:** активизирует удаление отрезков в исправленном профиле по сравнению с изначальным с учетом геометрических условий габарита, превышающих саму коррекцию На рисунке представлены две типовые ситуации, которые можно решить только при подключении сокращения профиля:



(1) запрограммированные профили,
(2) профили, полученные при коррекции на инструмент
R радиус коррекции

На рисунке слева показана часть профиля, заданная с фаской:
- применена коррекция на левой стороне профиля
- значение коррекции (R) превышает габарит фаски.

Если сокращение профиля неактивизировано, коррекция профиля не выполняется. Сигнализируется ошибка вследствие избытка коррекции на наклонном отрезке.

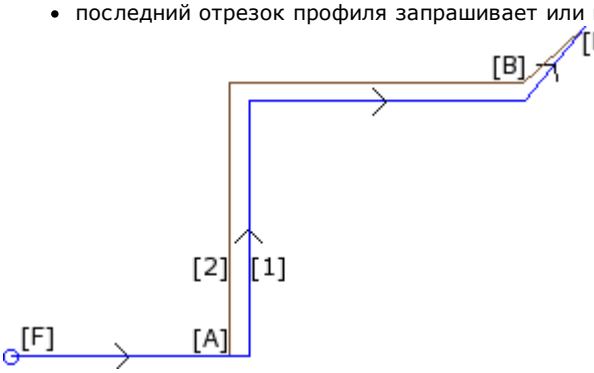
Исправленный профиль (2) получается при активизированном сокращении профиля: не отображается промежуточный отрезок, который был удален в проекции двух смежных отрезков для построения исправленного профиля.
Части пунктиром показывают, каким бы был исправленный профиль, если бы была признана действительной коррекция, примененная к промежуточному отрезку. Очевидно, что прохождение промежуточного отрезка было бы инвертировано с соответствующим изменением начальной геометрии.

На рисунке справа показана часть профиля, заданная со скруглением:
- применена коррекция на левой стороне профиля
- значение коррекции (R) превышает радиус скругления

Если сокращение профиля неактивизировано, коррекция профиля не выполняется. Сигнализируется ошибка вследствие избытка коррекции на дуге.
Исправленный профиль (2) получается при активизированном сокращении профиля: не отображается промежуточный отрезок, который был удален в проекции двух отрезков для построения правильного профиля.
Части пунктиром показывают, каким бы был исправленный профиль, если бы была признана действительной коррекция, примененная к промежуточному отрезку. И в этом случае прохождение промежуточного отрезка было бы инвертировано с соответствующим изменением начальной геометрии.

Сокращение профиля применяется только там, где требуется (где наблюдаются случаи, подобные приведенным выше), и может удалять и последовательные отрезки.
Необходимо подчеркнуть, как сокращение профиля никоим образом не оценивает профиль в комплексе: при удалении отрезка отыскивается решение пересечения отрезков, соответственно, до и после удаленного, без оценки того, будет ли это пересечение интерферировать с остальными частями профиля. Поэтому рекомендуется подключать сокращение профиля при необходимости и всегда проверять внесенное изменение, особенно при значениях коррекции, намного превышающих габариты исходного профиля.

- **Постепенное введение коррекции**: активизирует постепенное введение коррекции на первом отрезке профиля. Коррекция профиля вычисляется начиная со второго отрезка профиля и движение на первом отрезке линейно: от точки, запрограммированной для подготовки, до начальной скорректированной точки, вычисленной для второго отрезка. Постепенное введение коррекции не применяется при возникновении одного из следующих условий:
  - первый отрезок профиля не является линейным
  - профиль задан только с одним отрезком
  - первый отрезок профиля требует прерывания коррекции (см. ниже)
- **Постепенное прекращение коррекции:** активизирует постепенное прекращение коррекции на последнем отрезке профиля. Применяется, только если последний отрезок является линейным. Коррекция профиля вычисляется до предпоследнего отрезка и движение на последнем отрезке линейно: от конечной скорректированной точки предпоследнего отрезка до конечной точки, запрограммированной для профиля. Постепенное прекращение коррекции не применяется при возникновении одного из следующих условий:
  - последний отрезок профиля не является линейным,
  - профиль задан только с одним отрезком,
  - последний отрезок профиля запрашивает или продолжается прерыванием коррекции (см. ниже).



(1) запрограммированный профиль,
(2) профиль, полученный при коррекции на инструмент.

Профиль применяет:
- постепенный запуск коррекции ([F] - первый отрезок);
- постепенное прекращение коррекции ([L] - последний отрезок).
-

[A] - начальная скорректированная точка второго отрезка профиля;
[B] - конечная скорректированная точка предпоследнего отрезка профиля

- **Ввести коррекцию от подготовки:** активирует коррекцию с началом от точки, запрограммированной для подготовки. В списке имеется три пункта:
  - **умолчание** активирует режим, заданный по умолчанию (в конфигурации TpaCAD);
  - **выкл** отключает режим коррекции
  - **применить** активизирует режим коррекции;

При активизации этого пункта исправленный профиль начинается из точки, запрограммированной в подготовке, и доходит до точки коррекции на первом отрезке с линейным движением.
Введение коррекции от подготовки не применяется при возникновении одного из следующих условий:

- требуется и применяется постепенный запуск коррекции
- первый отрезок профиля требует прерывания коррекции (см. ниже).

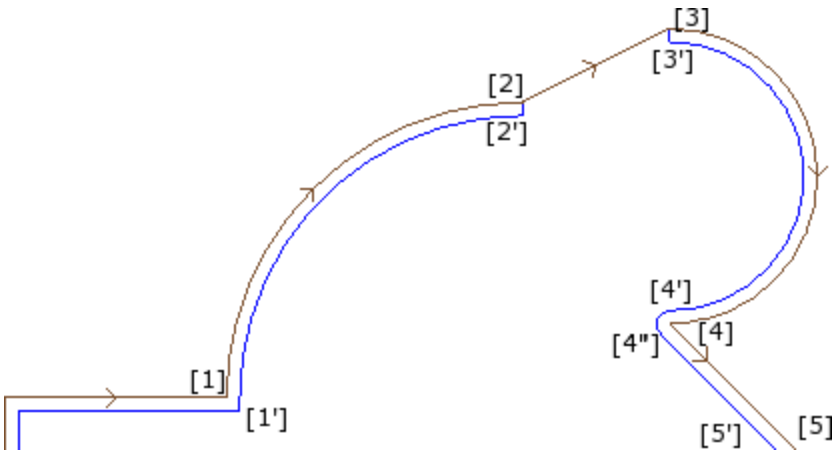
Запуск коррекции из подготовки обычно используется при обработке мрамора и других очень твердых материалов, когда на запрограммированных для подготовки позициях специальными инструментами выполняются *заходные отверстия*, с которых затем инструмент для выполнения профиля может легко начать работу без опасности поломки.

Изменение коррекции

Режимы применения коррекции на инструмент могут меняться также в ходе разработки профиля. В операции профиля можно задать параметр **Коррекция**, который может представлять до четырех пунктов в списке:

- **Без изменений**: коррекция продолжается без изменений по сравнению с предыдущим отрезком
- **Возобновить**: возобновляет коррекцию, если она была прервана или приостановлена
- **Прервать**: прерывает коррекцию начиная с текущего отрезка до последующего возобновления
- PROFESSIONAL **Приостановить**: приостанавливает коррекцию начиная с текущего отрезка до последующего возобновления. Эта опция доступна только в режиме **Профессиональный**.

Рассмотрим пример применения прерывания при коррекции:



Интересующая нас часть профиля запрограммирована на отрезках:

- ..
- [1] -> [2] (дуга)
- [2] -> [3] (линия)
- [3] -> [4] (дуга)
- ..

Запрограммированный профиль имеет стрелки направления. Коррекция находится с правой стороны профиля.
Рассмотрим профиль с примененной коррекцией на инструмент:

- ([1] -> [2]) скорректирован на дуге: [1'] -> [2']
- дополнительный линейный отрезок: [2'] -> [2]
- изначальный отрезок сегмента: [2] -> [3]
- дополнительный линейный отрезок: [3] -> [3']
- ([3] -> [4]) скорректирован на дуге: ([3'] -> [4']) и скругление ([4'] -> [4"]) добавлено до коррекции последующего отрезка.

Коррекция не была применена на отрезке (линейном): [2] -> [3]. В частности:

- отрезок [2] -> [3] снова проходится полностью и на исправленном профиле;
- коррекция отменена в конце предыдущего отрезка (дуга: [1] -> [2]), определяя точку [2'] так, как будто этот отрезок является последним профиля, с добавлением линейного отрезка от [2'] до точки [2];

- коррекция возобновлена на последующем отрезке (дуга: [3] -> [4]), определяя точку [3'] так, как будто этот отрезок является первым профиля, с добавлением линейного отрезка от [3] до точки [3'].

Коррекция как показана получена за счет использования параметра **Коррекция**, заданного на отрезках профиля.
В приведенном здесь примере исправленный профиль получается как требуется, если профиль задан:

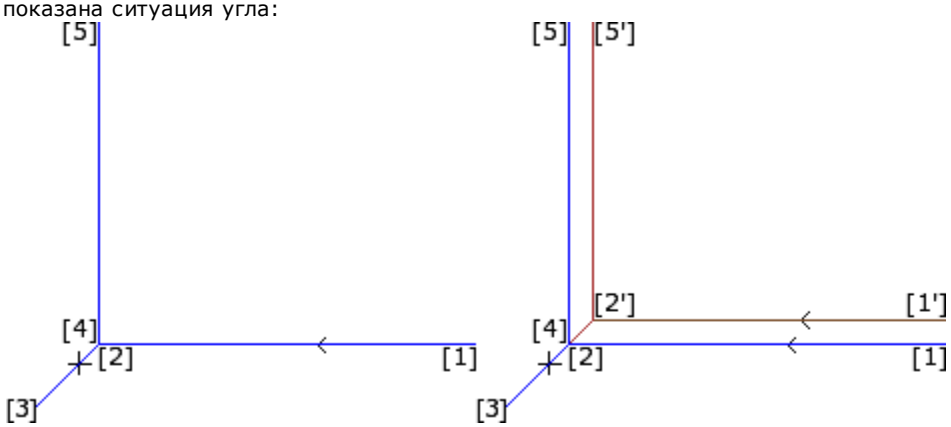
- [setup]: запрашивает сторону коррекции: Правая
- …
- [1] -> [2]: **Коррекция:** Без изменений
- [2] -> [3]: **Коррекция:** Прервать
- [3] -> [4]: **Коррекция:** Возобновить
- …

Можно:

- запросить прерывание коррекции в т. ч. на первом отрезке профиля;
- прерывание не обязательно должно аннулироваться возобновлением, оно может продолжаться до конца профиля.

Использованный пример позволяет выделить аспект, касающийся режима коррекции на ребрах. Подготовка профиля обязательно запросила решить коррекцию поиском пересечений (Контурирование: Ребра): решение пересечения в точке [1'] подчеркивает это.
Мы видели, что коррекция в точке [4] вставила скругление (дуга: [4'] -> [4]): это происходит потому, что коррекция двух отрезков, сходящихся в точке [4], не смогла найти пересечение и поэтому была вынуждена вставить соединительную дугу.

Пример применения приостановки при коррекции касается операции на углах рамок.
На рисунке показана ситуация угла:



Слева приводится профиль, как он запрограммирован, на отрезках:

- [1] -> [2]
- [2] -> [3]
- [3] -> [4]
- [4] -> [5]

Ребро находится на двух промежуточных отрезках (2 -> 3), (3 -> 4). **ВНИМАНИЕ**: точки [2] и [4] совпадают.
Коррекция находится с правой стороны профиля.

На рисунке справа показано, что необходимо получить, с примененной коррекцией на инструмент:

- первый скорректированный отрезок: [1'] -> [2']
- дополнительный линейный отрезок: [2'] -> [3]
- дополнительный линейный отрезок: [3] -> [2']
- последний скорректированный отрезок: [2'] -> [5'].

Точка [2'] определяется пересечением коррекций на двух изначальных отрезках, соответственно, до и после ребра: (1 -> 2) и (4 -> 5).

Коррекция как показана получена за счет использования параметра **Коррекция**, заданного на отрезках профиля.
В приведенном выше примере рамки исправленный профиль получается как требуется, если профиль задан:

- [setup]: запрашивает сторону коррекции: Правая
- …
- [1] -> [2]: **Коррекция:** Без изменений

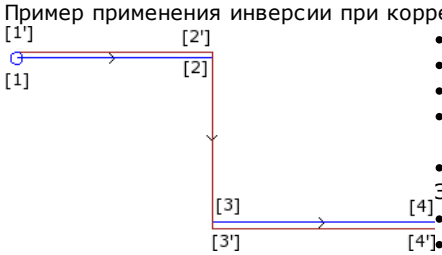
- [2] -> [3]: **Коррекция:** Приостановить
- [3] -> [4]: **Коррекция:** Приостановить
- [4] -> [5]: **Коррекция:** Возобновить
- …

Необходимо, чтобы:
- запрос на приостановку аннулировался возобновлением. Приостановка до конца профиля обуславливает сигнализацию ошибки применения коррекции на инструмент
- два отрезка до и после приостановки геометрически последовательны, т. е. первый заканчивается в точке начала второго. В противном случае подается сигнал ошибки применения коррекции на инструмент
- коррекция на двух отрезках (отрезки до и после приостановки) может определять условие пересечения (а не соединения). В противном случае подается сигнал ошибки применения коррекции на инструмент.

Изменение стороны коррекции

PROFESSIONAL **Изменение стороны коррекции (**Эта опция доступна только в режиме **Профессиональный**.)
В операции профиля можно выбрать параметр **Изменить сторону коррекции,** которая инвертирует сторону коррекции (слева направо или наоборот).
Подключение этой опции связано с некоторыми ограничениями:
- запрос может соответствовать возобновлению коррекции после прерывания или
- предыдущие отрезки, соответствующие запросу, могут решать пересечение исправленных отрезков; или
- предыдущие отрезки, соответствующие запросу, задают инвертированную геометрию.

Пример применения инверсии при коррекции



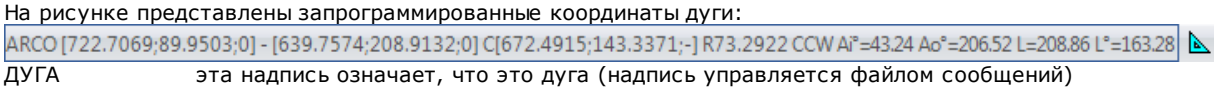
- [1] (подготовка) запрашивает левую коррекцию
- [1] -> [2] (линия): корректируется
- [2] -> [3] (линия): прерывает коррекцию
- [3] -> [4] (линия): возобновляет коррекцию и требует изменения стороны.
- ..

Этот же профиль с примененной коррекцией на инструмент:
- ([1] -> [2]) скорректирован слева на: [1’] -> [2’]
- дополнительный линейный отрезок: [2’] -> [2]
- изначальный отрезок: [2] -> [3]
- дополнительный линейный отрезок: [3] -> [3’]
- ([3] -> [4]) скорректирован на: ([3’] -> [4’]) на правой стороне.

Отображение

Команды **Габарит корректируемых профилей** и **Коррекция исходных профилей**, находящиеся в группе **Индивидуализировать виды** вкладки **Показать** меняют **Вид коррекции на инструмент**.
Габарит корректируемых профилей: исправленные профили и профили, не применяющие какой-либо коррекции, представляются толщиной, равной габариту инструмента. Для этих профилей не отображаются крайние точки и стрелки направления.
В любом случае, остаются представленными с унитарной толщиной:
- профили конструкции
- отрезки профиля в воздухе.

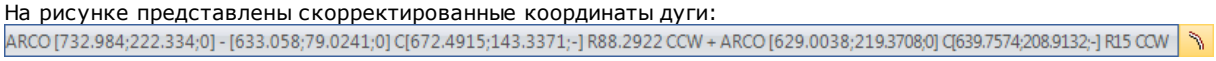
Если не активизируется, исправленные профили представляются с унитарной толщиной.
Исходные профили процесса коррекции при подключении: вид содержит также исходные профили (неисправленные профили), в противном случае вид содержит только исправленные профили и профили, не применяющие никакой коррекции (с применением, если требуется, стрелок направления на представленных отрезках).
строка состояния
При активированном Виде коррекции на инструмент в строке состояния можно отобразить координаты, относящиеся к запрограммированным или исправленным отрезкам. Переход осуществляется при щелчке на изображении справа от зоны координат.
На рисунке представлены запрограммированные координаты дуги:



ДУГА	эта надпись означает, что это дуга (надпись управляется файлом сообщений)
[722.7069;…]	положение начальной точки дуги
-	положение конечной точки дуги
[639.7574;208.91	

32;…]	
C[672.4915;…]	положение центра дуги
R=73.2922	радиус дуги
CCW (против часовой стрелки)	вращение против часовой стрелки (CW, если по часовой стрелке)
Ai°=43.24	начальный угол отрезка (в градусах)
Ao°=206.52	конечный угол отрезка (в градусах)
L=208.86	длина отрезка (в трехмерном виде)
L°=163.28	угловая амплитуда дуги (в градусах)

На рисунке представлены скорректированные координаты дуги:



ДУГА	эта надпись означает, что это дуга (надпись управляется файлом сообщений)
[732.984;…]	положение начальной точки скорректированной дуги
-[633.058;…]	положение конечной точки скорректированной дуги
C[672.4915;…]	положение центра дуги
R=88.2922	скорректированный радиус дуги
CCW (против часовой стрелки)	указывает вращение против часовой стрелки
+	указывает, что коррекция вставила отрезок после дуги
ДУГА	эта надпись означает, что это дуга скругления (надпись управляется файлом сообщений)
[629.0038;…]	положение конечной точки дуги скругления
C[639.7574;…]	положение центра дуги скругления
R15	радиус дуги скругления
CCW (против часовой стрелки)	указывает вращение дуги скругления против часовой стрелки

Для скорректированного отрезка не приводятся углы (начальный и конечный) и длина.

Выполнение профиля для насечки с острыми ребрами

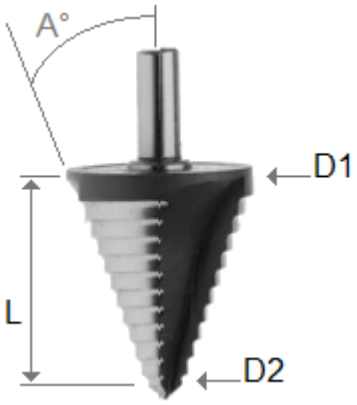
PROFESSIONAL

На подготовке профиля можно запросить изменить обычную логику исполнения профиля для получения насечки с острыми ребрами. Эта опция относится к выполнению ребер профиля, включенные между двумя отрезками профиля.

Подключение выполняется в узле **Продвинутые технологические данные**:

- **Насечки с острыми ребрами:** подключение работает только при выполнении некоторых условий
  1. инструмент, заданный на подготовке, - конический
  2. запрограммированная глубина на точке подготовки зацепляет инструмент в детали.

На рисунке приводится пример конического инструмента:



характеризуется двумя крайними диаметрами и углом конуса. Очевидно, что инструмент выполняет паз в материале в зависимости от задействованного сечения: чем больше запрограммированная глубина, тем больше диаметр, работающий на поверхности детали.

При удовлетворении условий на ребрах добавляются два линейных движения:

• одно подъема инструмента до нулевой координаты глубины к теоретическому ребру выемки инструмента, на наружной части ребра (определяется со стороны большего угла). Длина отрезка определяется формой инструмента, геометрией ребра и запрограммированной глубиной на вершине ребра.
• одно опускания инструмента для установки на запрограммированное ребро.
В результате получается более ярко выраженное ребро, до максимума, разрешенного инструментом.
В конфигурации определяются минимальный угол и максимальный угол ребра так, чтобы применить добавление линейных отрезков: можно конфигурировать интервал значений от 1° и 179°.
Из оценки исключаются ребра, образованные запрограммированными входными и выходными отрезками

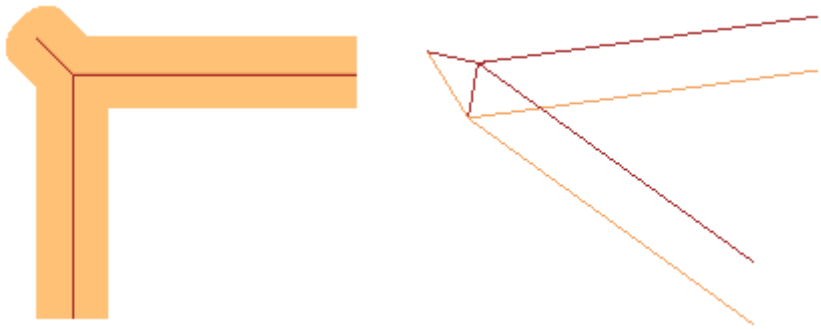
профиля.

Типовое применение - применение в операции рамок.

Если профиль требует Поправки на инструмент и все также предполагая, что выполняются установленные условия:

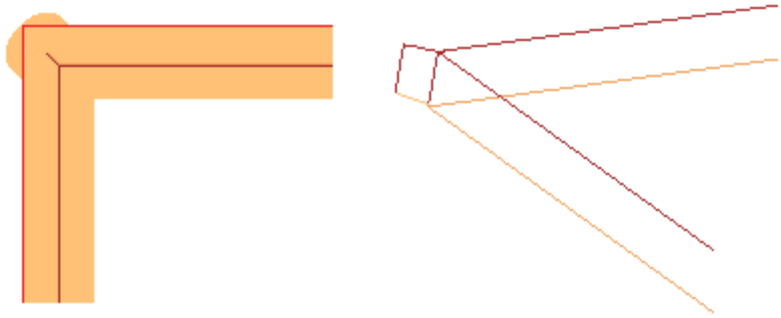
- диаметр, используемый для коррекции, - исходный, заявленный для инструмента (на рисунке: D1)
- оцениваются ребра скорректированного профиля, полученные с решением пересечения
- исключаются ребра, образованные запрограммированными входными и выходными отрезками профиля.
- кроме того, исключаются ребра, образованные линейными отрезками, добавленными для применения приостановки, прерывания или возобновления поправки на инструмент.

на рисунке приводится пример с примененным подъемом на ребре под прямым углом (90°):



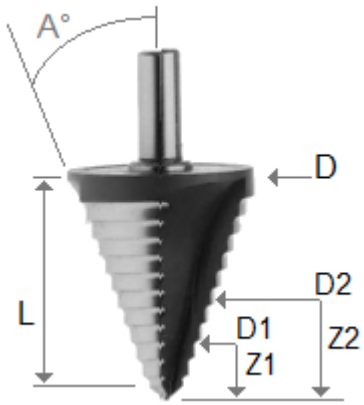
Как видно из рисунка, подъем на ребре добавляется на “наружной части” профиля. В прямоугольном профиле интуитивно определяется наружная часть, но это не всегда так. Профиль может иметь вогнутые и выпуклые части, кроме того, он может быть замкнутым или нет. По всем этим причинам с полем **Насечка с острыми ребрами** связаны четыре различных выбора:
• **Выкл.**: выбор исключен
• **Автоматический**: обнаруживает наружную часть профиля автоматически. Прежде всего, рассматривается, требуется ли профилю поправка на инструмент:
    - при правой коррекции наружная часть находится слева от профиля, а при левой - справа
    - если не требуется поправка на инструмент и профиль замкнут, оценивается направление вращения профиля: по часовой стрелке - соответствует левой наружной части, против часовой стрелки - правой наружной части
    - в противном случае: наружная часть профиля определяется на геометрии первого используемого ребра, найденного на профиле
• **Слева**: наружная часть находится слева от профиля
• **Справа**: наружная часть находится справа от профиля.

Кроме того, в конфигурации можно подключить описанное выше функционирование в более расширенном режиме, включая также случай неконического инструмента. В этом случае добавленные отрезки подтверждают запрограммированную глубину на ребре и, хотя и сохраняется то же направление в плоскости грани, что и в предыдущем случае, длина останавливает внешний габарит инструмента на теоретическом ребре.



Компенсация диаметра коррекции в случае конического инструмента

При использовании конического инструмента можно применить компенсацию диаметра, используемую при коррекции инструмента. Функциональность должна быть включена в **Конфигурации TpaCAD** и не требует Профессионального режима.
Компенсация учитывает запрограммированную глубину при настройке профиля, за исключением возможного входа в профиль.



Со ссылкой на рисунок:

- в случае глубины ввода в деталь запрограммированной Z1 используется диаметр, обозначенный как D1
- в случае глубины ввода в деталь запрограммированной Z2 используется диаметр, обозначенный как D2
- в случае глубины ввода в деталь запрограммированной от значения, равного или превышающего полезную длину инструмента (обозначено как L), используется максимальный заявленный диаметр, обозначенный как D
- в случае запрограммированной глубины в воздухе, диаметр D используется относительно детали.

Если опция не выбрана, всегда используется диаметр D.

Задание профилей в режиме "грань-деталь"

Задание профилей в режиме "грань-деталь" требует некоторых дополнительных уточнений.
На этапе конфигурации TpaCAD можно сделать выбор из двух различных рабочих режимов:

- **распознавание превалирующего профиля в поле F:** в этом рабочем режиме отдается предпочтение непрерывности профилей, а не заданию грани применения. При операциях профиля (дуги и линии) распознавание открытого профиля не учитывает задания поля F (текущей и предыдущей операции):
  - при открытии отрезка профиль сохраняет свое исходное программирование поля F
  - в противном случае - распространяет поле F от предыдущего отрезка

  В случае подготовки или сложной операции с **запросом** зацепления точки поле F распространяется с предыдущей операции.

- **распознавание профиля, обусловленного полем F:** в этом рабочем режиме отдается предпочтение заданию грани, а не непрерывности профилей. При операциях профиля (дуги и линии) распознавание открытого профиля учитывает задания поля F (текущей и предыдущей операции) и другие задания, в любом случае, определяют прерывание профиля. Распространение поля F никогда не применяется с предыдущего отрезка. В случае подготовки или сложной операции с **запросом** зацепления точки поле F не распространяется с предыдущей операции и зацепление точки не определяет продолжение профиля, если предыдущее задание поля F оказывается другим.

9.3 Логические команды

Логические инструкции - это особые простые операции, которым не соответствует какое-либо выполнение на станке. Логическая инструкция может задавать выполнение, обусловленное одной или несколькими операциями, или сама выполнять определенную функцию, выбирая, обуславливать ли ее на основе значения логического выражения (например, ОШИБКА).

Структуры IF ... ELSEIF ... ELSE ... ENDIF

Логическая команда может вводиться при вызове команды **If..EndIf, If...ElseIf...Else..EndIf, If..Else.. EndIf,** из группы **Блоки** вкладки **Применить** или при выборе операция IF, ELSEIF, ELSE, ENDIF в группе ЛОГИЧЕСКИХ КОМАНД.



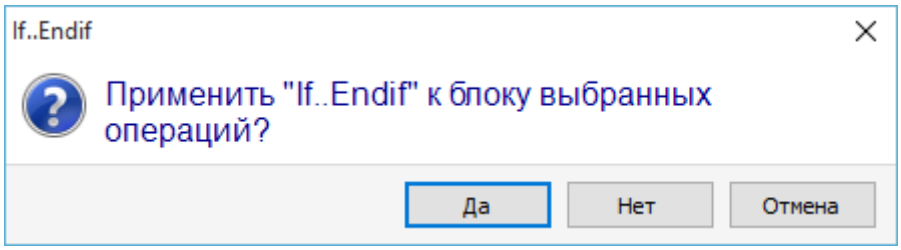
Структура **If..EndIf** - это самая простая форма программируемой альтернативы. Инструкция IF выражает условие, которое:

- если оказывается ВЕРНЫМ, определяет выполнение одной или нескольких указанных операций после IF
- если оказывается ЛОЖНЫМ, определяет невыполнение соответствующих операций.

Инструкция ENDIF ограничивает операции, обуславливаемые IF.
Между IF и ENDIF может задаваться ELSE, отрицающее условие, оцениваемое при помощи IF.
Форму **If...Else...EndIf** можно перефразировать следующим образом: "Если действительно условие, выраженное **If**, выполняет операции, указанные после **If**; в противном случае выполняет операции, указанные после **Else**".

Более сложная форма может выражаться как **If... ElseIf.. ElseIf.. Else.. EndIf**, которая может задавать альтернативные друг другу условия: первое наступившее условие определяет альтернативную форму, с возможностью, что не наступит ни одно условие. Если сложная форма заканчивается ветвью **Else**: она выполняется как альтернатива по умолчанию, если не наступило ни одно из приведенные ранее условий.
Исход заданных в программе логических ограничений отображается при запросе применения логических условий командой, находящейся в группе **Виды** вкладки **Показать**. При активном этом виде отображаются только операции, в которых были соблюдены логические ограничения.
Исход логических ограничений, заданных в циклах IF.. ELSE.. ENDIF, не устанавливает условий для интерпретации операций, как они запрограммированы в последовательности грани. Уточним этот пункт. После цикла IF..ENDIF, который выполняет профиль, программируется операция сверления с относительным значением 100 на координате X: положение X отверстия определяется путем суммирования запрограммированной координаты (100) с конечным положением профиля в цикле IF, **независимо** от проверки логических условий, установленных для инструкции IF.

В некоторых случаях ввод логического блока путем выбора команды в группе **Блоки** вкладки **Применить** может обуславливать отображение следующего сообщения:



Выбрать **[Да]** для ввода группы выборов, к которым относится текущая операция, непосредственно в блок. В случае, показанном в окне выше (ввод If...Endif), предполагая, что мы находимся на строке 5 и выбрали строки с 3 по 12:

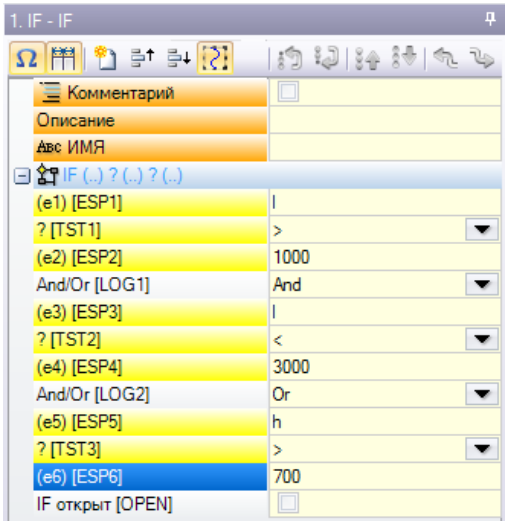
- инструкция IF вводится **до** строки 5
- инструкция ENDIF вводится **после** строки 12.

Логическое состояние операций приводится также в тексте ASCII:

	Текст ASCII			M
6	IF ESP1=r\rot TST1=4 ESP2=0 LOG1=0 TST2=0 LOG2=0 TST3=...	0	0	0
7	IF ESP1=r\arcoin TST1=4 ESP2=0 LOG1=1 ESP3=r\radin TST...	0	0	0
8	SETUP EG0 X0 Y-r31 Zr14 TMCr20 TRr21 Tr22 TPifelse[rem...	0	0	0
9	A42 EG0 EW0 I0 Jr31 UYr30*2 Nr13 TAr0 Ur4 ANr\narchirea...	0	0	0
10	ELSE	0	0	0
11	IF ESP1=r24 TST1=4 ESP2=1 LOG1=0 TST2=0 LOG2=0 TS...	0	0	0
12	SETUP EG0 Xr26 Y-r31+r26 Z0 TMCr20 TRr21 Tr22 TPife...	0	0	0

на рисунке: в структуре IF… ELSE … ENDIF является верной ветвь IF

Условие, выраженное инструкциями IF и ELSEIF, может быть представлено тремя термами. Рассмотрим пример:



(e1) ? (e2): первый терм

And/Or: логическое условие между первым и вторым термом
(e3) ? (e4): второй терм

And/Or: логическое условие между результатом второго условия и третьим термом
(e5) ? (e6): третий терм

Поля **(e..)**, появляющиеся в терме, как правило, имеют параметрическое задание.

Элемент **?** между полями **(e..)** терма задает условие сравнения:

<	меньше	(например, (e1) < (e2))
<=	меньше или равно	(например, (e1) <= (e2))
>	больше	(например, (e1) > (e2))
>=	больше или равно	(например, (e1) >= (e2))
=	равно	(например, (e1) = (e2))
<>	не равно	(например: (e1) <> (e2))

Терм признается ВЕРНЫМ, если соблюдается установленное условие сравнения.
ВНИМАНИЕ: Сравнение между полями **(e..)** всегда выполняется с некоторым допуском, равным 0.001 (эпсилон сравнения): значения, отличающиеся на величину меньше эпсилон, считаются одинаковыми.

Логическое условие между двумя термами в отношении принимает значение:

And	если два терма должны оба быть ВЕРНЫМИ
Or	если достаточно, чтобы только один из двух термов был ВЕРНЫМ.

Можно задать: никакой, один, два или три терма логического ограничения.
Если никакой терм для IF не задается, соответствующая ветвь будет всегда удовлетворяться. В этом случае если IF задает также ветви ELSEIF или ELSE, они никогда не удовлетворяются.
То же самое касается программирования ELSEIF. Если устройство контроля начинает оценивать инструкцию (т. е. не удовлетворяется ни одна указанная ранее ветвей для цикла IF) и не задан ни один терм: соответствующая ветвь удовлетворяется и завершает выполнение цикла IF.

Команды ELSE и ENDIF являются полностью проходными: у них нет задаваемых полей.

Структуры логического ограничения IF …ELSEIF .. ELSE… ENDIF могут встраиваться без каких-либо ограничений.
Программирование, показанное на рисунке, соответствует для оценки логического выражения:

IF (((l > 1000) and (l < 3000)) or (h > 700)) {...} ENDIF

То есть:

если (l) больше 1000 **и** также (l) меньше 3000;
или: если (h) больше 700
тогда: инструкция IF ВЕРНА.

при задании: l=2000, h=500

(l > 1000) ВЕРНО
(l < 3000) ВЕРНО
(h > 700) ЛОЖНО

оценивает: (ВЕРНО and ВЕРНО) or ЛОЖНО => ВЕРНО or ЛОЖНО => ВЕРНО.

IF открыт

Закрытие IF инструкцией ENDIF обязательно, если только IF не выбирает поле **IF открыт**.
В этом случае инструкция IF обуславливает только последующую операцию, которая, однако, не может быть:

- операцией setup или профиля
- логической инструкцией (IF, ELSEIF, ELSE, ENDIF) или Точкой применения (в подпрограмме).

Возможные неправильные применения инструкций IF .. ELSEIF..ELSE.. ENDIF сигнализируются при применении логических условий. Ситуации ошибки приводятся в главе Ошибки логических условий

Команда "Exit"

Команда EXIT позволяет принудительно задавать ситуации логических ограничений с переходом вперед при выполнении запрограммированного текста. Условие перехода выражается тем же формализмом, который используется для команды IF.
Если условие команды ВЕРНО или не задано, команда интерпретирует условие перехода. В этом случае:

- определяет прямой выход после цикла IF на уровне ближайшего вложения
- при выполнении команды вне цикла IF инструкция EXIT вызывает переход к концу программы грани.

Даже внутри цикла IF, в любом случае, можно принудительно задать переход к концу программы, выбирая поле ВОЗВРАТ.
Условие перехода оценивается только на этапе применения логических условий, как условие, уже выраженное для команды IF. Если условие для команды оказывается ВЕРНЫМ, для всех операций, запрограммированных в цикле IF, после команды EXIT, принудительно задается условие ЛОЖНО.
Если результат теста оказывается ЛОЖНЫМ, определяет нормальное продолжение выполнения программы. Результат теста, в любом случае, будет верным, если не предусматривается никакого логического условия.

Команда "Error"

Команда ERROR программирует ситуации ошибки. Условие ошибки выражается тем же формализмом, который используется для команды IF. Если условие команды ВЕРНО или не задано, команда интерпретирует условие ошибки.
Если ошибка генерируется при вызове подпрограммы, ее развертка не выполняется и сигнализируется ошибка.
Если ошибка генерируется непосредственно в главном тексте программы:

- TraCAD сигнализирует ситуацию ошибки при применении логических условий, значок "стоп" на команде ERROR сигнализирует, что условие ВЕРНО
- на этапе выполнения отменяет интерпретацию программы и блокирует ее выполнение.

Команда ERROR может эффективно управлять контролем действительности параметров и/или переменных, заданных при вызове подпрограммы, или переменных, заданных на этапе выполнения программы.
При выборе пункта **Ошибка** при вводе данных операции отображается список, в котором предлагаются заданные ошибки (номер + сообщение).

На уровне производителя можно ввести новое сообщение или изменить существующее. После выбора сообщения для изменения или ввода изменение должно записываться в поле редактирования **Новое сообщение**. Для подтверждения ввода сообщения нажать кнопку **[Применить].**

Команда "Warning"

Команда WARNING программирует ситуации предупреждения. Действительно все сказанное применительно к команде ERROR, только теперь представляется не ошибка, а простое предупреждение, которое, как таковое, не влияет на нормальную разработку или выполнение программ или подпрограмм.

ОШИБКА выполняется: при выборе этого поля можно дифференцировать поведение этой инструкции: на этапе выполнения она активирует ситуацию ОШИБКИ, аннулирует интерпретацию программы и блокирует ее выполнение.

Переменные J

Иногда бывает необходимо или удобно выполнять задания переменных в ходе определения программы грани. Например, это необходимо, когда программа не может полностью определяться заранее или требует информации, вытекающей из применения подпрограмм или макропрограмм. Кроме того, как правило, может оказаться удобным выполнять локальные задания по мере написания программы грани, а не группировать их в таблицу переменных r: это может также облегчить понимание программы.
Для этой цели существуют переменные <j>. Это 100 переменных числового типа, определенных именем: от j0 до j99.
Переменные <J> являются локальными для грани. Это означает, что:

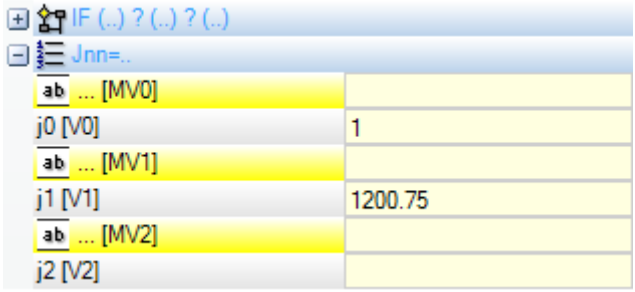
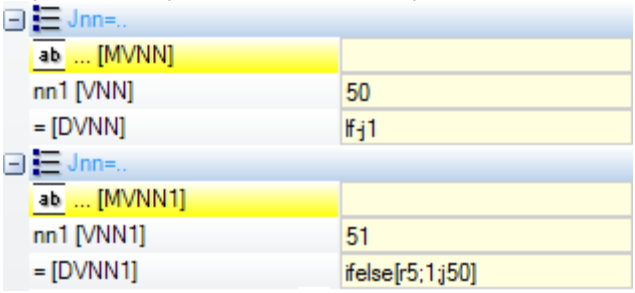
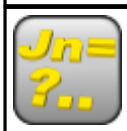
- не существует никакой связи между заданиями и считыванием переменных между различными гранями
- каждая программа грани запускается с совокупностью переменных, заданных на 0.0 (ноль).

Переменные <j> могут использоваться в каждой операции, примененной к грани, на любом уровне:

- можно использовать переменную <j> для задания диаметра отверстия или рабочей координаты, или логического условия
- внутри грани видимость переменных глобальна на всех уровнях применения. Поэтому:
  - главная программа может задать j5=1
  - приложение подпрограммы может менять значение j5 (например, j5=2)
  - после применения подпрограммы главная программа может протестировать значение j5, если она находит его измененным.

При применении вложенных вызовов подпрограммы: переменные <j> имеют значения, заданные в момент главного вызова.

В палитре операций, как правило, определяются три инструкции задания переменных <j>:

	ЗАДАТЬ Jnn	эта инструкция позволяет задавать одну или несколько переменных <j>, связывая, при необходимости, это задание с логическими условиями (заданными в **IF (..) ? (..) ? (..)**): • задания выполняются только если поставленное логическое условие оказывается ВЕРНЫМ • первый пункт Jnn=… группирует определенное число прямых заданий: на рисунке представлено  2 пункта, от j0 до j2. В примере заданы первые две переменные: j0=1; j1=1200.75 • последующие пункты позволяют задавать такое же количество переменных с указанием индекса переменной.  В примере заданы две переменные: J50 со значением “lf-j1” и J51 со значением “ifelse [r5;1,j50]”
	ЗАДАТЬ Jnn при условии (.. ? ..: ..)	эта инструкция позволяет задавать одну или несколько переменных <j> на основе оценки логических условий (заданных в **IF (..) ? (..) ? (..)**). Запрограммированные задания выполняются в любом случае при удовлетворении установленного логического условия: часть заданий действительна при условии ВЕРНО, часть - при условии ЛОЖНО. С точки зрения информатики это условие тернарного типа.

		Для каждой переменной задается пункт с тремя полями: • первое поле задает индекс переменной (значение от 0 до 99) • второе поле указывает задание, которое должно выполняться при удовлетворении логических условий (ВЕРНО) • третье поле указывает задание, которое должно выполняться при неудовлетворении логических условий (ЛОЖНО)
	ЗАДАТЬ Jnn (0 - 99)	• Эта инструкция позволяет задавать все или группу переменных <j>, связывая, при необходимости, эти задания с логическими условиями (заданными в **IF (..) ? (..) ? (..)**): • первое поле задает начальный индекс задаваемой группы (например, 0), второе поле задает конечный индекс задаваемой группы (например, 50). Если поля пусты, они задают не группы, а переменные полностью, от j0 до j99 • третье поле указывает выполняемое задание Задания выполняются только если поставленное логическое условие оказывается ВЕРНЫМ.

Среди параметров этих инструкций могут иметься также описательные тексты, относящиеся к каждой отдельной переменной <j>. строки начинаются как [MV0].. [MV1]...

Ценность переменных J

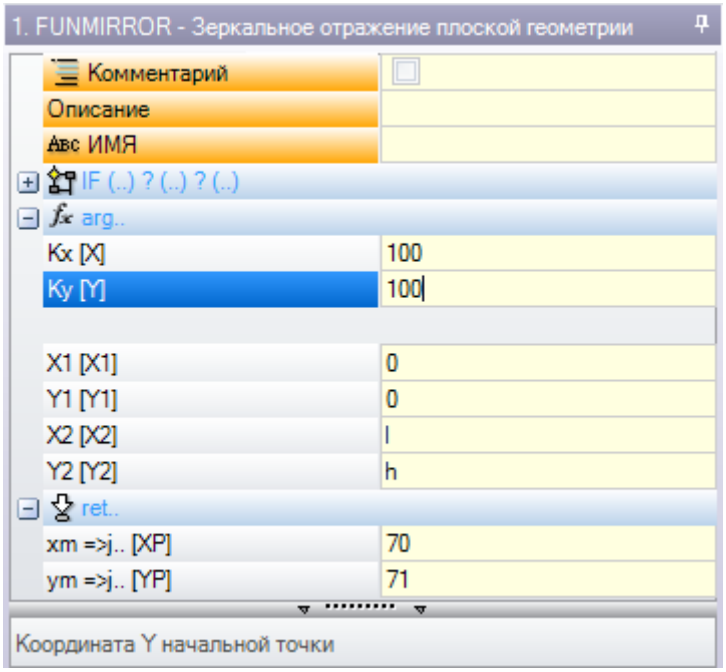
Переменные <j> отображаются в зоне команд в нижней левой части экрана.
В окне 100 переменных <j> представлены в таблице с 10 строками и таким же количеством столбцов:
строка j_: содержит переменные от j0 до j9;
строка j1_: содержит переменные от j10 до j19;
.....
строка j9_: содержит переменные от j90 до j99;

При проведении курсора мыши по клетке отображается сообщение справки (подсказка) с указанием имени соответствующей переменной и заданного ей значения (например, "j4=1025.6").
На Общем виде детали все переменные принимают значение 0.0.
На виде грани: приведенные в этом окне значения могут меняться, если меняется текущая операция. Это окно обновляется по состоянию переменных, доступному после текущей операции.

Глобальные функции

Глобальные функции - это особые логические инструкции, позволяющие выполнять более или менее сложную процедуру расчета и напрямую задавать результаты в переменных <j>.
Они должны предусматриваться на этапе конфигурации приложения с подробной оценкой конкретных потребностей индивидуализации.

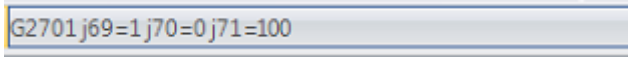
Ниже приводится простой пример.
Необходимо определить положение точки P координат (r0;r1) в зеркальном отображении вокруг общей оси, заданной двумя точками: P1 (r2;h/2), P2 (l/2;r3). Один из возможных способов: взять формулы для требуемой трансформации и задать первую переменную r для координаты x и вторую переменную r для координаты y. Если трансформация служит для отдельного случая, такое решение, несомненно, может использоваться.
Теперь предположим, что необходимо вычислить трансформацию несколько раз и в различных программах, тогда каждый раз мы должны вспоминать формулы и переписывать их.
При использовании глобальных функций можно написать один раз все формулы и вызывать их специальной инструкцией, которая скрывает все промежуточные этапы формул и напрямую выдающую результаты.



arg.. группирует аргументы, требуемые инструкцией:
- координаты точки, требующей зеркального отображения (x;y);
- координаты двух точек оси (P1(x1;y1) и P2(x2;y2)).

ret.. группирует поля возврата:
- **funmirror =>j..:** задает индекс переменной <j>, возвращающей результат функции (здесь: j69): например, 1, если исход правильный, 0, если исход неправильный
- **xm =>j..:** задает индекс переменной <j>, возвращающей координату x трансформации (здесь: j70);
- **ym =>j..:** задает индекс переменной <j>, возвращающей координату y трансформации (здесь: j71).

При задании, как показано на рисунке, в строке состояния указаны переменные, заданные командой:



J69=1 : правильный исход функции (наш пример не предусматривает случаи неправильного решения)
J70=0 : координата X в зеркальном отображении
J71=100 : координата Y в зеркальном отображении

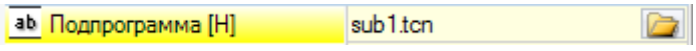
9.4 Подпрограмма

Подпрограмма

Подпрограмма - это файл программы-детали, созданный с типом как программы, так и подпрограммы.
Во вкладке **Операции** в группе ПОДПРОГРАММЫ определены 3 типа кодов для применения подпрограммы:

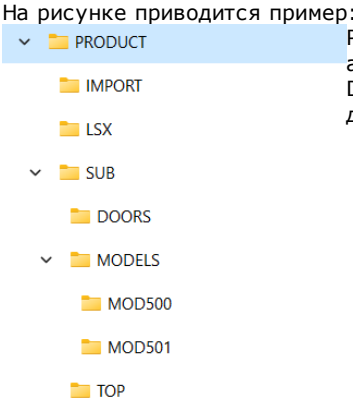
SUB	управляет геометрическими трансформациями и многократным применением со свободным повторением
SMAT	управляет геометрическими трансформациями и многократным применением с повторением в матричной форме
SEMPTY	управляет геометрическими трансформациями за исключением масштабного коэффициента. Не управляет многократным применением. Может генерировать выемки.

Рассмотрим некоторые поля, значимые для применения подпрограммы:
- узел **"IF (..) ? (..) ? (..)"**: возможность задать условия применения подпрограммы, вызываемой напрямую. Подпрограмма применяется только если результат логических ограничений ВЕРНЫЙ.
- **Подпрограмма:** может редактироваться также в параметрической форме или же может задаваться при открытии окна Открыть Деталь. Значок ab в левой части поля указывает, что это параметр типа *строки*. В окне *Открыть Деталь* поиск задается в папке стандартной архивации подпрограмм (SUB). В этом окне перечисляются типы файлов, соответствующих программному формату. При переносе выбора в файл формата, действительного для программы-детали, в окне размеров приводятся комментарий и просмотр графического представления. При закрытии окна имя выбранной подпрограммы приводится в поле SUB. Пример:



В примере не приводится весь маршрут нахождения подпрограммы, а только имя.расширение. Распознается **относительная адресация к стандартной папке архива подпрограмм (SUB).** Это обеспечивает возможность переносимости программ. Если мы скопируем нашу программу на другую установку, для правильной работы достаточно скопировать также подпрограмму sub1.tcn в папке SUB. При относительной адресации если программа имеет расширение макрокоманды (*.TMCR), она ищется в папке макрокоманд, а не подпрограмм. Тип программы выводится из ее считывания и в случае макрокоманды не открывается. Имя и расширение подпрограммы не могут содержать следующие символы: **\ /:* ? " < > | # %.**

Папка SUB может содержать другие папки для распределения архива подпрограмм .
На рисунке приводится пример:



PRODUCT - стандартная папка архива программ; SUB - стандартная папка архива подпрограмм, заданная в папке программ; в SUB создаются папки: DOORS, MODELS, TOP и каждая из этих папок может, в свою очередь, иметь дочерние папки

- Если бы подпрограмма SUB1.TCN была выбрана в "..\PRODUCT\SUB\MODELS\MOD500\", поле SUB было бы задано как: "MODELS\MOD500\ SUB1.TCN"
- если подпрограмма SUB1.TCN выбирается в папке программ "..\PRODUCT\", поле SUB задается как: "..\SUB1.TCN": и в этом случае сохраняется относительная адресация, обеспечивающая переносимость программы
- если бы подпрограмма SUB1.TCN была выбрана в дочерней папке папки программ "..\PRODUCT\DOORS\", поле SUB было бы задано как: "..\DOORS\SUB1.TCN": и в этом случае сохраняется относительная адресация, обеспечивающая переносимость программы
- если подпрограмма выбирается не из папки программ, поле SUB содержит весь маршрут нахождения подпрограммы, больше не гарантируя переносимость программы.
- Rnnn: задает перезадаваемые переменные "r" подпрограммы. См. главу Задать переменные подпрограммы

- **Зацепление точки:** возможность продолжения профиля
- **Относительный, Rel->:** абсолютный или относительный режим по сравнению с предыдущей операцией в списке
- **Позиционировать габаритный прямоугольник:** позиционирует точку применения подпрограммы исходя из габаритного прямоугольника согласно следующим опциям:
  - **Центровать на XY:** по центру габаритного прямоугольника
  - **X-Y-:** в точке минимального габарита по X и по Y
  - **X-Y+:** в точке минимального габарита по X и в точке большего габарита по Y
  - **X+Y-:** в точке большего габарита по X и в точке меньшего габарита по Y
  - **X+Y+:** в точке большего габарита по X и по Y
- имеющиеся в распоряжении геометрические трансформации:
  - **X1, Y1, Z1**: перенос (поля задают точку применения)
  - **Угол вращения**: вращение
  - **Зеркальное отображение по горизонтали, Зеркальное отображение по вертикали**: зеркальные исполнения
  - **Инвертировать:** выполняет инверсию исполнения подпрограммы

- **Вложенные грани:** перечисляет грани, которые должны применяться в случае вложенных вызовов
- **Исключенные грани:** перечисляет грани, которые не должны применяться в случае вложенных вызовов
- **Вложенные XY:** выбирает между различными режимами адаптации точки применения (размещения) во второстепенных вызовах (вложенные XY)

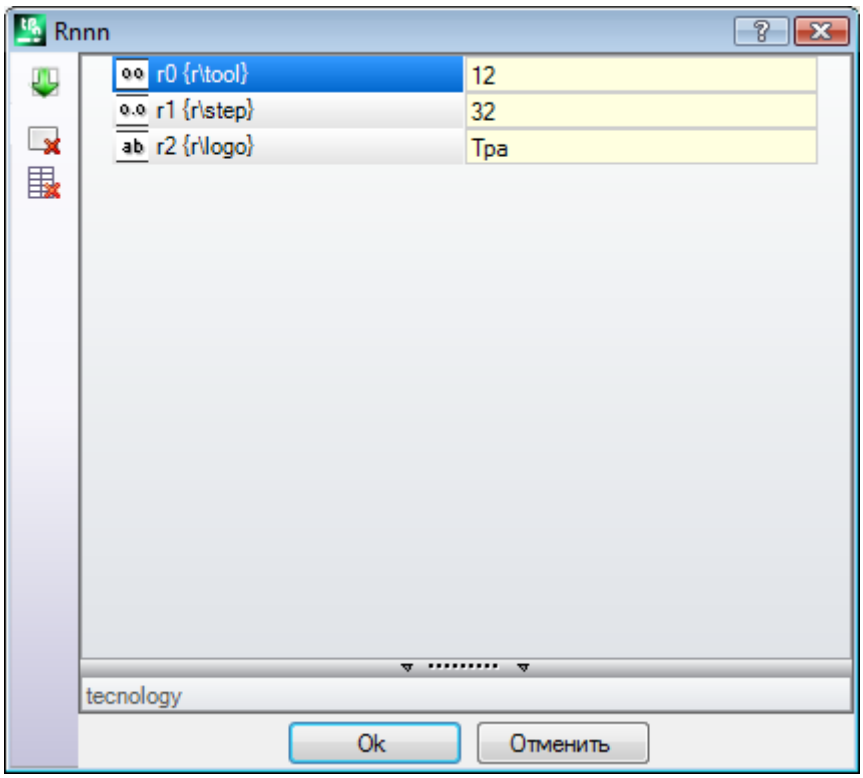
- **Выемка**: запрос создания выемок
- узел: **Масштабный коэффициент**: задает масштабный коэффициент для изменения размера
- узел: **Повторения:** задает многократное применение подпрограммы с возможностью выбора между свободным повторением и повторением в матричной форме

- **Свойства**:
  Все свойства могут задаваться в коде подпрограммы. В целом можно сказать, что это касается всех сложных кодов, если только при конфигурации операций не определяются другие характеристики. Рассмотрим ниже некоторые особые аспекты:

- поле “C” (Комментарий): вся операция представляет собой комментарий и нет применения подпрограммы;
- поле “L” (Уровень): при чисто положительном значении (>0) вся развертка кода SUB принимает заданное значение (значение распространяется). В случае нулевого значения (0): значение, как правило, не распространяется (это настройка по умолчанию, но можно решить распространять и значение 0). Если подпрограмма выполняет зацепление профиля (продолжает профиль, который начинается ранее): распространяет значение поля “L” от подготовки профиля;
- поле “B” (Конструкция): действительны те же положения, которые принимаются для поля “L”;
- поля “O”, “M”: распространение значения решается на уровне конфигурации как для чисто положительных значений, так и для значения 0. Если подпрограмма выполняет зацепление профиля (продолжает профиль, который начинается ранее): может распространять значение поля от подготовки профиля или сохранять другую настройку, как это определено в конфигурации;
- поля “K”,”K1”,”K2”: распространение значения решается на уровне конфигурации как для чисто положительных значений, так и для значения 0. Если подпрограмма выполняет зацепление профиля (продолжает профиль, который начинается ранее): распространяет значение поля от подготовки профиля.

Задать переменные подпрограммы

Пункт **Rnnn** задает перезадаваемые переменные <r> подпрограммы, которые могут меняться только в специальном окне. Приводятся только заданные переменные, перезадаваемые в подпрограмме. Этот пункт не управляется, если не задано поле SUB или если его задание недействительно или если подпрограмма не имеет перезадаваемых переменных.
Предлагается следующее окно:



Во втором столбце отображаются имя переменной (r0, r1) и расширенное имя или описание, если не задано расширенное имя. В третьем столбце приводится задание переменной. Значок в первом столбце каждой строки указывает тип переменной. На рисунке r0 она целого типа 00, r1 - двойного типа 0.0, r2 - типа строки ab.

При вводе подпрограммы поля столбца инициализируются со значениями, заданными переменным в подпрограмме. Если поле пустое, может наблюдаться один из следующих двух случаев:
- задается значение 0.
- задается значение, заданное переменной в тексте подпрограммы

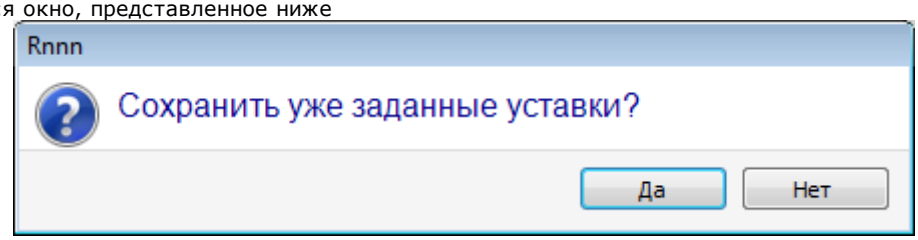
Это зависит от того, как сконфигурирован TpaCAD производителем станка.

При задании переменной можно вызвать справку параметрического программирования: меню справки открывается при щелчке правой кнопкой мыши на поле редактирования переменной. Если задание

оказывается недействительным, например, вследствие ошибки синтаксиса, подается сигнал ошибки непосредственно в подтверждении данных и представленное сбоку изображение остается для визуальной сигнализации состояния ошибки. В любом случае, закрытие окна не является подтверждением и не оказывается под влиянием сигнализаций ошибки.

Кнопки инструментальной панели, слева от таблицы, позволяют:

Импортировать задания всех переменных из подпрограммы. Если в списке имеются уже заданные поля, отображается окно, представленное ниже



Выбрать **[Да]** для задания только незаданных полей
Выбрать **[Нет]** для перезаписи всех полей с заданиями, считанными подпрограммой.

Обнулять значение выбранной переменной

Обнулять значение всех переменных

Для задания переменных действительны те же положения, что и для любого другого поля операции. В любом отношении, это информация, относящаяся к задаваемой операции, только с более высокой степенью конфигурации. При изменении подпрограммы или изменении имени подпрограммы окно переменных может меняться.
В частности, можно использовать любую действительную форму параметризации. Рассмотрим несколько примеров задания цифровых переменных:

- r0=r5+32 : использует переменную r5 программы
- r1=100.5 : задана только численно
- r12=lf/2 : использует длину грани, на которой применяется подпрограмма

Все переменные r подпрограммы, заданные, но неперезадаваемые, пересчитываются на основе новых заданий. В тексте подпрограммы, например, заданы две неперезадаваемые переменные:

- r100=lf-r0*2
- r101=r10

Значение переменной r100 задается с:

- lf: длина грани, на которой применяется подпрограмма
- r0=r5+32

Значение переменной r101 задано со значением r10, как задано в подпрограмме. Если подпрограмма не задает r10, переменную следует искать среди переменных программы, к которой применяется подпрограмма, как поясняется в следующем параграфе.

Автоматическое задание переменных Rnnn
Автоматическое задание переменных **Rnnn** происходит, когда в подпрограмме используется одна или несколько переменных r, но им не присваивается никакого значения (поле пустое). При вызове подпрограммы из программы поиск указанных переменных осуществляется в вызывающей программе, а в случае нескольких вызовов каскадом поиск задаваемого значения может выполняться в обратном направлении до вызывающей программы.
Это удобный механизм для полностью автоматической передачи одного или нескольких сведений подпрограммам в случае, если весь архив программ всегда использует эти сведения. Однако могут быть получены нежелательные результаты при неправильном использовании функции, например, если вы забываете оставлять свободной переменную r. Описанный здесь механизм может полностью устареть вследствие возможности использования переменных, которые всегда являются общедоступными в программе, например, переменные <o> и <v>. Поэтому рекомендуется ограничить использование автоматического задания переменных **Rnnn** случаями реальной необходимости и взять за правило использовать только четко заданные переменные.

Для лучшего понимания механизма автоматического задания рассмотрим следующий пример:
в подпрограмме для задания диаметра инструмента для сверления используется переменная r0, но переменная остается незаданной. В этом случае в подпрограмме значение r0 определяется равным нулю числового (двойного) типа.
Подпрограмма затем вызывается в новой программе:

- если программа не задает переменную r0, все остается без изменений: диаметр инструмента для сверления равен 0.0;

- если программа присваивает значение 10 переменной r0, применение подпрограммы меняется: диаметр инструмента сверления теперь равен 10.0

При использовании переменной r с символическим именем рекомендуется сделать специальную заметку. В нашем примере предположим, что мы используем эту переменную не как "r0", а как "r\fitool", и не задаем переменную в подпрограмме: теперь в редактировании подпрограммы подается предупреждение (несерьезная ошибка) 103: "Параметрическое программирование: переменная "r", вызванная по имени, не найдена".
Подпрограмма затем вызывается в новой программе:
- если программа не присваивает переменную r с именем "fitool", все остается без изменений: диаметр инструмента для сверления равен 0.0 и сигнал (предупреждение) еще отображается;
- если программа присваивает переменной "r" имя "fitool" и значение 10, применение подпрограммы меняется: диаметр инструмента для сверления теперь равен 10.0 и сигнал стирается.

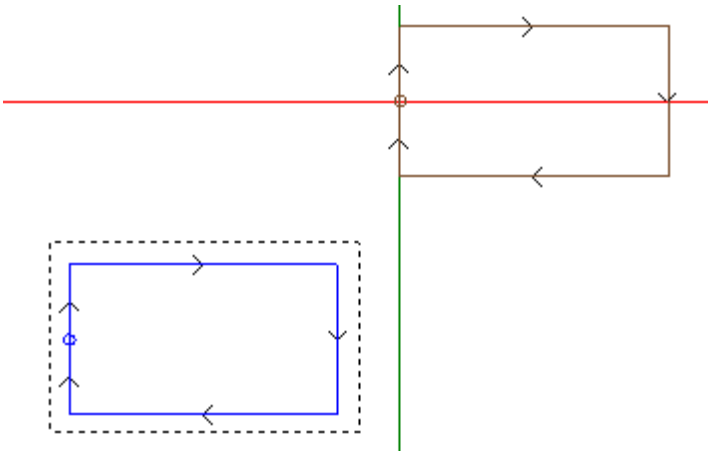
Описанный здесь механизм поиска и задания переменных можно, в любом случае, изменить в конфигурации TpaCAD с возможностью:
1. Полного его исключения: в этом случае незаданная переменная Rnnn всегда равна 0.0. При использовании переменной по имени, которое не представляется заданным (например, "r\fitool"): диагностическое сообщение, соответствующее ошибке 103, теперь больше не является предупреждением, а является настоящей ошибкой;
2. Подключать исключительно при использовании переменной по имени. Продолжая рассматривать наш пример: для "r\fitool" подключился бы механизм автоматического задания, но не для "r0".

Другие автоматические задания подпрограммы
Если в подпрограмме используется размер детали (l, h, s) или переменная 'o'/ 'v' или настройка клиентского раздела или переменной геометрии, на что ссылается эта информация?
Ответ ясен: на главную программу, вызывающую подпрограмму, независимо от точки вызова подпрограммы (см. параграф "Встроить вызовы подпрограмм").

Позиционировать подпрограмму

Подпрограмма позиционирована на плоскости XY грани, а глубина - в направлении Z, перпендикулярно плоскости грани: расчетные значения трех координат (x, y, z) определяют **точку применения** (точку, которую мы обозначим P1)
Программирование точки применения выполняется в системе **картезианских координат** с возможностью задания абсолютных и относительных координат.

При выбранном **относительном** режиме можно принудительно задать абсолютный режим на одной отдельной координате, указывая перед координатой надпись "a;".

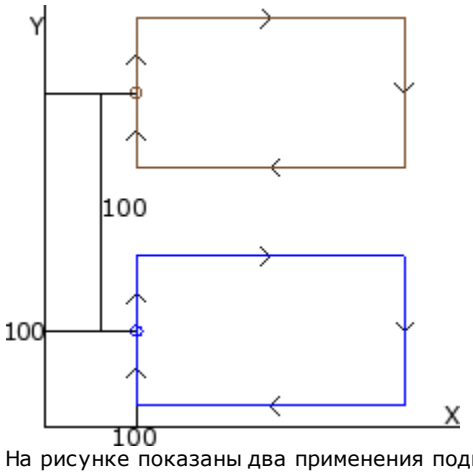


Выбранный на рисунке прямоугольник представляет создание подпрограммы (прямоугольник, проходящий по часовой стрелке с точкой начала, позиционированной на половине левой вертикальной стороны).
Курсор в виде креста указывает точку применения P1:
в P1 позиционируется точка начала подпрограммы (подготовка прямоугольника).

Если координаты точки P1 не заданы (поле пустое), наблюдается один из двух описанных ниже случаев:
- не применяется перенос относительно изначального положения подпрограммы. Например, если для P1 заданы только две координаты на плоскости XY, позиционирование прямоугольника по Z остается без изменений.
- распространяется координата предыдущей операции с использованием тех же критериев, что и для позиционирования точечной операции (например, одиночное сверление).

Это зависит от того, как сконфигурирован TpaCAD производителем станка.

Если активизирован режим относительного позиционирования и операции предшествует другой сложный код (будь то макрокоманда или операция SUB), оценивается также поле **Отн <-.** Если активизировано и последнее указанное поле, точка применения P1 считается относительной по сравнению с точкой применения (P1) предыдущей операции.



На рисунке показаны два применения подпрограммы, рассмотренной в качестве примере (которая выполняет прямоугольник):

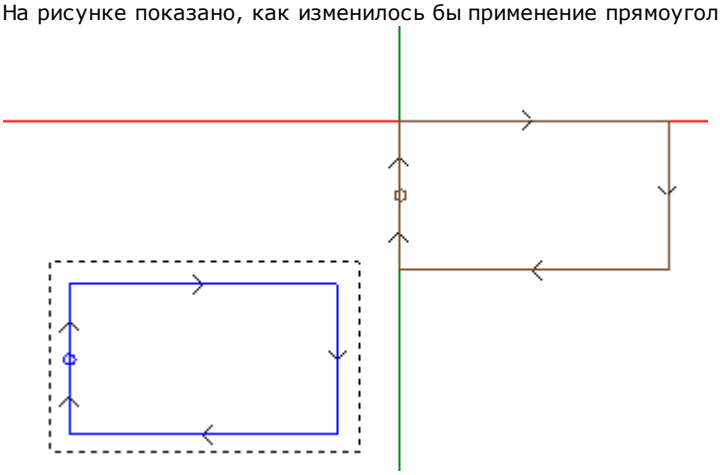
- внизу точка применения запрограммирована в абсолютном отношении на (X=100; Y=100);
- вверху точка применения запрограммирована в относительном отношении и с выбранным полем **Rel<-** и координатами x = 0 и y = 100:
  - относительная координата X=0 задает координату x точки P1 на той же координате x, что и точки P1 первого применения (прямоугольник внизу)
  - относительная координата Y=100 задает координату y точки P1, прибавляя 100 к координате y точки P1 первого применения (прямоугольник внизу)

Возможно было бы решить по-другому, какую точку подпрограммы приводить в P1, приводя, например, не операцию setup, а центр прямоугольника?

Ответ положителен и предусматривает различные решения.
Одним из таких решений является выбор в окне задания кода SUB применения подпрограммы пункта **Позиционировать габаритный прямоугольник**. Это поле выбора с несколькими пунктами:

- **Не применять**: это поле не влияет на позиционирование подпрограммы
- **Центровать на XY**: в P1 приводится центр габаритного прямоугольника подпрограммы
- **X- Y-**: в P1 приводится точка минимального габарита как по x, так и по y
- **X- Y+**: в P1 приводится точка минимального габарита по x и большего по y
- **X+ Y-**: в P1 приводится точка большего габарита по x и меньшего по y
- **X+ Y+**: в P1 приводится точка большего габарита как по x, так и по y

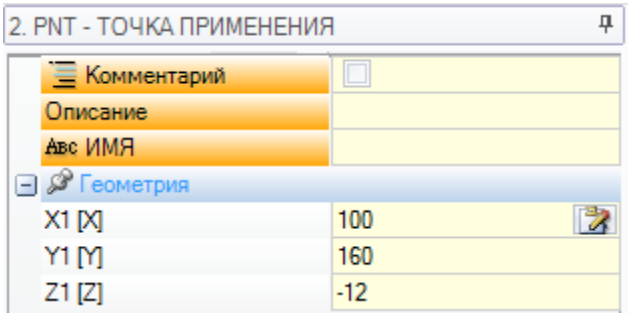
На рисунке показано, как изменилось бы применение прямоугольника при выборе X- Y+:



ВНИМАНИЕ:
Из-за простоты подпрограммы, взятой для примера, габаритный прямоугольник подпрограммы совпадает с запрограммированным рисунком.

Запрограммированная точка применения

В подпрограмме можно запрограммировать координаты точки, которая должна переноситься на точку применения P1.
Для программирования необходимо вызвать логическую инструкцию **Точка применения**:

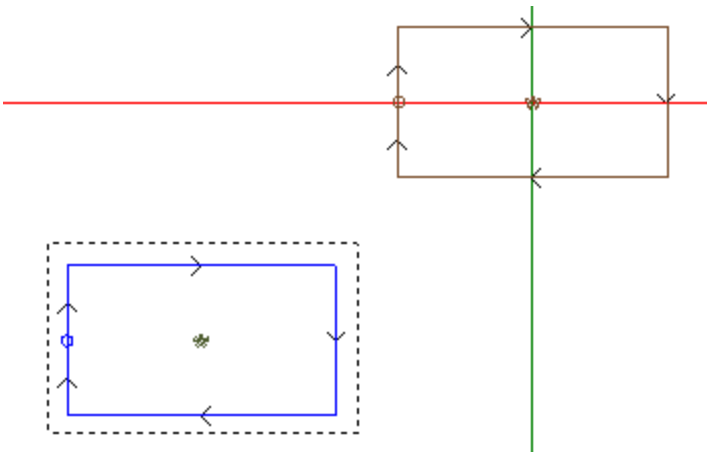


Три поля X1, Y1, Z1 задают точку, которая должна позиционироваться при вызове подпрограммы.
Программирование интерпретируется в абсолютных координатах и действительно для всех трех координат, для возможных незаданных полей принимается значение 0.0.
ВНИМАНИЕ:
Код интерпретируется **только** при применении подпрограммы.

При применении подпрограммы значимой признается только точка применения: первая, удовлетворяющая логические условия.

Заданная здесь точка применения не обязательно должна соответствовать рабочему положению. В прямоугольнике в примере она может обоснованно задавать координаты центра прямоугольника, как это запрограммировано в подпрограмме.

На рисунке показано, как меняется применение подпрограммы при добавлении инструкции ТОЧКА ПРИМЕНЕНИЯ для центра прямоугольника.



Инструкция ТОЧКА ПРИМЕНЕНИЯ подпрограммы игнорируется, если код SUB применения подпрограммы задает действительный выбор в пункте **Позиционировать габаритный прямоугольник**.

Зацепление точки

Выбор опции зацепления точки:
- принимает для 3 координат точки применения (P1) относительный режим с нулевыми смещениями (делает неэффективными другие задания точки P1);
- делает неэффективным выбор полей: **Отн <-** и **Позиционировать габаритный прямоугольник**
- игнорирует команду ТОЧКА ПРИМЕНЕНИЯ, заданную в подпрограмме.

Зацепление точки **всегда** применяет подпрограмму, связанную в относительном плане с нулевыми смещениями.

Если до кода SUB (предыдущая программная строка, не комментарий) имеется элемент профиля, на котором может быть выполнено зацепление (подготовка, дуга или линия, другая сложная операция, завершающая свою развертку элементом профиля) и если текущая подпрограмма начинается с элемента профиля, на котором может выполняться зацепление (подготовка, дуга или линия):
приложение подпрограммы продолжает профиль, начатый ранее, фактически исключая исполнение подготовки, которая начинает саму подпрограмму.
В этом случае зацепление точки распознало ситуацию: **зацепления между профилями**.

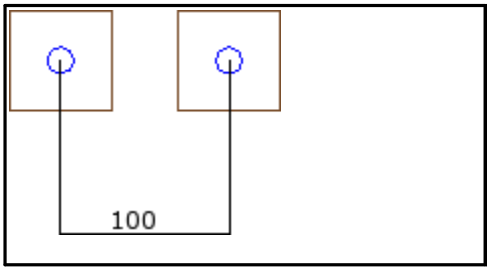
Конечная точка применения

Определение последней выполненной в создании подпрограммы операции зависит от типа этой операции. Если это:
- точечная или подготовительная операция (например, одиночное сверление), последняя обработанная точка определяется ее точкой применения
- отрезок профиля (линия или дуга), последняя обработанная точка определяется конечной точкой отрезка
- подпрограмма, последняя обработанная точка определяется созданием подпрограммы.

Определим эту точку: **конечная точка применения**.
При создании подпрограммы конечная точка применения релевантна для применения:
- повторений при выполнении самой подпрограммы
- последующей операции с заданием координаты в относительном режиме или в случае распространения координат.

В подпрограмме можно запрограммировать координаты КОНЕЧНОЙ ТОЧКИ ПРИМЕНЕНИЯ, используя специальную логическую инструкцию. Эта инструкция интерпретируется только внутри подпрограммы и при использовании нескольких инструкций внутри одной подпрограммы считается правильной последняя проверенная логическими условиями.

Три поля X1,Y1, Z1 задают конечную точку применения, которая не обязательно должна совпадать с координатами рабочей точки. Программирование координат выполняется в абсолютном режиме и для незаданных координат принимается значение 0.0.



На примере, приведенном сбоку, представлена подпрограмма с одним отверстием и одним прямоугольником конструкции вокруг отверстия. КОНЕЧНАЯ ТОЧКА ПРИМЕНЕНИЯ задается так, чтобы совпадать с отверстием. При этом вызов, например, в относительном режиме, этой же подпрограммы, запрограммированной со смещением по X, равным 100, определяет ее позиционирование относительно отверстия. Если также ТОЧКА ПРИМЕНЕНИЯ начального размещения совпадает с отверстием, положение отверстия полноправно становится единственным значимым положением подпрограммы.

Использование команды КОНЕЧНАЯ ТОЧКА ПРИМЕНЕНИЯ исключает возможность зацепления после подпрограммы с признанием продолжения профиля.
Инструкция КОНЕЧНАЯ ТОЧКА ПРИМЕНЕНИЯ игнорируется, если в коде SUB вызова подпрограммы задается трансформация **Инверсия**.

Применить операции к соответствующей грани

Подпрограмма - это файл программы-детали, созданный с типом как программы, так и подпрограммы. Поэтому операции подпрограммы могут применяться на одной или нескольких гранях. Как правило, необходимо указать, какую грань подпрограммы вы желаете применить, заданием параметра **Грань** операции SUB. Это задание может определять два различных функционирования:
- подключение вложенных вызовов: соответствует случаю невыполненного задания (поле пустое);
- прямое подключение вызова: соответствует случаю выполненного задания (поле НЕ пустое).

Вложенные вызовы (автоматические)

Применение подпрограммы определяет автоматическое выполнение всех непустых граней подпрограммы, имеющих соответствие грани в программе, применяющей эту подпрограмму.
Термин "вложенный" означает, что применение подпрограммы распространяется, здесь автоматически, на другие грани.
Функционирование этого типа соответствует самому общему случаю механизма применения вложенных вызовов и определяется также термином ***автоматическое*** в противоположность механизму *Запрограммированные вложенные вызовы* (см. ниже).
Применение автоматических вложенных вызовов всегда работает (т. е. не подключается каким-либо образом) и активизировано, если оставлено незаданным поле *Грань*.

Например:

- создадим и сохраним подпрограмму ОДИН со следующими заданными операциями:
  - отверстия на грани 1
  - паз на грани 3
  - отверстия на грани 4
- теперь создадим программу PRG1, выберем грань 1 и введем код SUB, вызывающий подпрограмму ОДИН, оставляя поле Грани незаданным. Посмотрим, как меняется графическое представление детали при введенной операции:
  - выполняются операции, заданные в грани 1 подпрограммы ОДИН
  - выполняются также операции, заданные в подпрограмме ОДИН на гранях 3 и 4: это выполнения, соответствующие *вложенным вызовам.* Программную строку, определяющую разложение вложенных вызовов, называют *главный вызов,* а грань, к которой она относится, - *главной гранью.*
  - теперь сохраним программу PRG1
- изменим подпрограмму ОДИН, задавая операции также на грани 5
- теперь снова откроем программу PRG1: сразу видно, что вызов подпрограммы вводит также операции на грани 5
- еще раз изменим подпрограмму ОДИН: удалим все операции с грани 3
- теперь снова откроем программу PRG1. Видно, что были удалены операции на грани 3.

Можно увидеть структуру кода (SUB или аналогичный) и всех вложенных вызовов, открыв окно расширения обработки из **текста ASCII** (см.: Показать развертку подпрограммы)

Применение вложенного вызова имеет некоторые особенности:
- каждый вызов соответствует одной добавленной программной строке, одной для каждого заданного вызова, управляемой только автоматически и оставленной скрытой (невидимой)
- игнорирует возможное относительное программирование или выбор зацепления точки главного вызова
- каждый вызов применяет состояние переменных J, соответствующее главному вызову.

Механизм автоматических вложенных вызовов реализован только на базовом уровне программирования. Для ясности продолжим приведенный выше пример:
- снова откроем программу PRG1 и введем несколько отверстий на грань 3
- теперь создадим программу PRG2, войдем в программирование грани 3 и введем код SUB, вызывающий программу PRG1, оставляя поле Грани незаданным. Теперь посмотрим, как меняется графическое представление детали при введенной операции:
  - выполняются операции, заданные в грани 3 PRG1
    - выполняются также операции, заданные в грани 1 PRG1 и обуславливаемые вызовом подпрограммы ОДИН
    - не выполняются операции в других гранях PRG1, которые вначале обуславливались применением подпрограммы ОДИН: вызов подпрограммы ОДИН теперь не определяет никакого вложенного вызова, т. к. не находится на базовом уровне программирования.

Если бы программа PRG1 была создана с типом подпрограммы, она сама заблокировала бы механизм вложенных вызовов, изначально избегая ситуаций непонимания.

Выбор вложенных граней

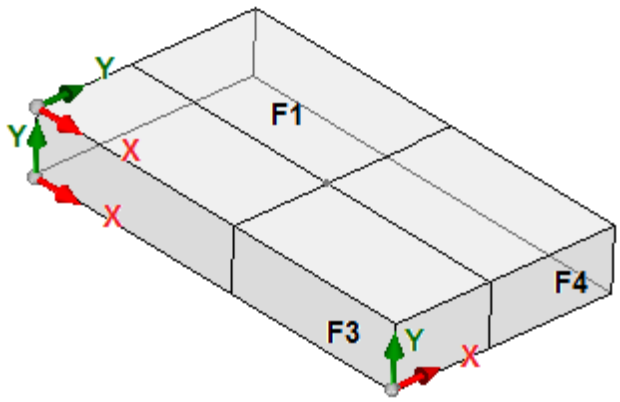
Применение вложенных вызовов может быть селективным:

ab Вложенные грани [SON]	3;5
ab Исключенные грани [SOFF]	

Вложенные грани: при задании указывает грани, задействованные во вложенном вызове. На рисунке задание "3;5" означает, что вложенные вызовы применены только на гранях 3 и 5
Исключенные грани: при задании указывает грани, не задействованные во вложенном вызове. Задание "3;5" указывало бы, что вложенные вызовы применены ко всем граням, за исключением граней 3 и 5. Поле Исключенные грани интерпретируется только если не задано поле Вложенные грани.
В обоих полях перечислять номера граней, разделяя их символом ";" (точка с запятой).

Размещение вложенных вызовов

Во вложенном вызове точка применения может задаваться различными способами с выбором различных позиций поля **Вложенные XY:**
- **По умолчанию:** это поле не влияет на позиционирование. Применяет режим, заданный в конфигурации ТраCAD: это один из трех перечисленных ниже пунктов выбора.
- **Адаптировать XY:** адаптировать точку применения
- **Пройти XY=:** для каждого вложенного вызова проходит поля, как они заданы в главном вызове
- **Не проходить XY:** для каждого вложенного вызова проходит незаданные поля.



Необходимость принятия точки применения для вложенных вызовов возникает ввиду вероятности физического несоответствия осей X и/или Y между различными гранями.
Рассмотрим рисунок (на нем показаны три видимые грани детали)

- зададим применение подпрограммы на грани 1: указана точка применения на плоскости грани 1
- вызовы вложены на двух остальных указанных гранях: грани 3 и 4

Теперь рассмотрим оси X и Y на двух вложенных гранях:

- грань 3: ось X физически соответствует оси X грани 1, а ось Y физически не соответствует оси Y грани 1;
- грань 4: ось X физически соответствует оси Y грани 1, а ось Y физически не соответствует оси X грани 1.

Поэтому может оказаться логичным иметь автоматические ассоциации:

- на грани 3: координата применения X = координата применения X грани 1; координата применения Y не задана;
- на грани 4: координата применения X = координата применения Y грани 1; координата применения Y не задана.

В таблице ниже представлены соответствия, примененные параметром **Вложенные XY**, заданным в выборе **Адаптировать XY**:

Главная грань	**Вложенная грань**	**Координата на вложенной грани**
(1,2)	(4,6)	X =координата Y с главной грани (если не задана ="") Y = ""
(1,2)	(3,5)	X=координата X с главной грани Y = ""
(3,5)	(1,2)	X=координата X с главной грани Y = ""
(4,6)	(1,2)	X=координата Y с главной грани (если не задана ="") Y=""
(любой другой случай)	(любой другой случай)	X=координата X с главной грани Y=координата Y с главной грани

В частности, вложенный вызов на фиктивной грани всегда применяет те же задания X и Y, что и главный вызов.

Для координат точки P1, не заданных во вложенном вызове (поле пустое), не применяется перенос относительно исходного положения подпрограммы. Таким образом, к вложенным вызовам никогда не применяется распространение координат предыдущей операции, как может произойти для главного вызова.

Решение переменных <j> в автоматических вложенных вызовах

Даже создание автоматического вложенного вызова может использовать переменные <j>.
Рассмотрим пример:

- напишем подпрограмму ОДИН с операциями, заданными на гранях 1 и 3:
  - запрограммируем на грани 1:
    - строка 1: код <ОТВЕРСТИЕ> на координате x = 100.0
  - запрограммируем на грани 3:
    - строка 1: код <ОТВЕРСТИЕ> зададим координату X=100
    - строка 2: IF (j0 > 0)
    - строка 3: код <ОТВЕРСТИЕ> зададим координату X=50 в относителбном режиме
    - строка 4: ENDIF

- напишем программу PRG1 с операцией, заданной на грани 1:
    - строка 1: код <SUB> применяю подпрограмму ОДИН: подпрограмма применяет <ОТВЕРСТИЕ> на грани 1 (координата X=100) и создает вложенный вызов, запрограммированный на грани 3;
    - вложенный вызов на грани 3 применяет только одно <ОТВЕРСТИЕ> на координате X=100

- изменим программу PRG1, вставив новые инструкции (в грань 1):
    - строка 1: <ЗАДАТЬ Jnn> зададим значение j0=1
    - строка 2: код <SUB> применяю подпрограмму ОДИН: подпрограмма применяет <ОТВЕРСТИЕ> на грани 1 (координата X=100) и создает вложенный вызов, запрограммированный на грани 3
    - вложенный вызов на грани 3 задает тепер два отверстия:
        - <ОТВЕРСТИЕ> на координате X=100
        - <ОТВЕРСТИЕ> на координате X=150
    - строка 3: <ЗАДАТЬ Jnn> зададим значение j0=0
    - строка 4: код <SUB> применяю подпрограмму ОДИН: подпрограмма применяет <ОТВЕРСТИЕ> на грани 1 (координата X=100) и создает вложенный вызов, запрограммированный на грани 3
    - вложенный вызов на грани 3 задает одно <ОТВЕРСТИЕ> на координате X=100.

Автоматические вложенные вызовы используют переменные J, назначенные во время основного вызова (в примере: на грани 1):
- любые изменения переменных J, запрограммированных в *основном* вызове, не влияют на собственные *вложенные* вызовы, но на последующие строки программы.
- любые изменения переменных J, запрограммированных во вложенных вызовах, влияют на один вложенный вызов.

Прямые вызовы

Применение подпрограммы определяет выполнение только программы грани, четко указанной в параметре *Грань*.
Все также со ссылкой на предыдущий пример, к грани 1 программы PRG1 можно применить любую грань подпрограммы ОДИН, записывая только соответствующий номер в поле "Грань" (в т. ч. в параметрической форме).

Запрограммированные вложенные вызовы

Мы говорим о поведении, которое решается как альтернатива автоматическим вложенным вызовам с приоритетом оценки, который рассматривается ниже.
Термин “вложенный” означает, что применение подпрограммы распространяется на другие грани, но не автоматически, а программно, при помощи специальной операции.
Применение запрограммированных вложенных вызовов требует специального подключения, которое зависит от функции и от грани:
- эта функция может оказаться полностью нерабочей: в этом случае порядок работы определяется двумя предыдущими случаями. В случае функции *Essential*: функция всегда отключена;
- в противном случае все равно можно получить ограниченное разрешение на применение подпрограммы (или макропрограммы) в грани-детали или комплексно на каждой грани.

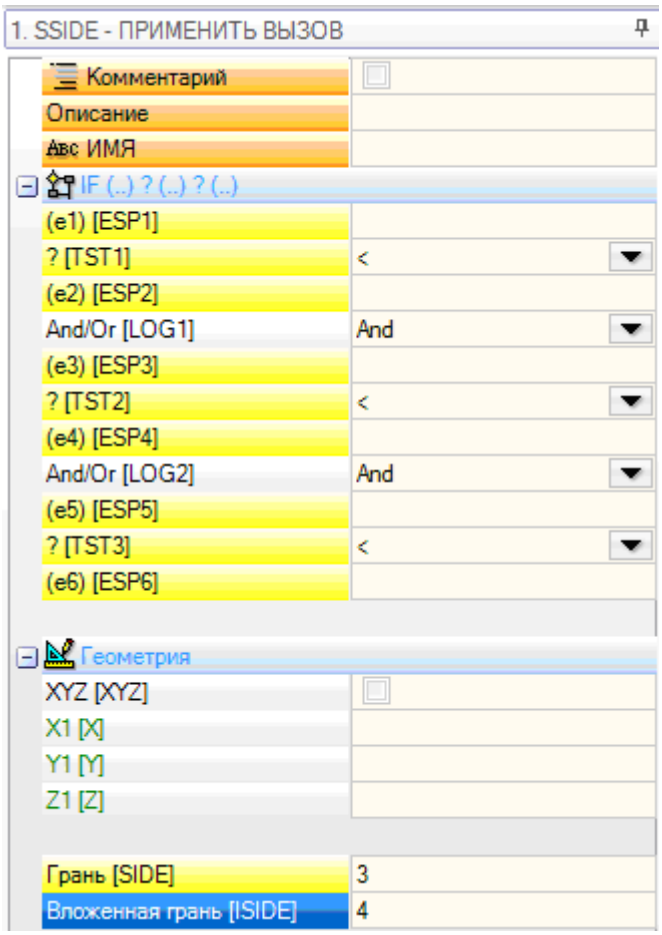
Поэтому характеристики грани-детали могут отличаться или полностью соответствовать характеристикам любой другой грани.

Распознавание этой функции осуществляется, если применение подпрограммы решает один код или оба:
- SSIDE: код применения вложенного вызова;
- NSIDE: код создания автоматической грани (при применении в режиме "грань-деталь").
  А это напрямую исключает применение автоматических вложенных вызовов.

Оба кода рассматриваются в последующих параграфах.
Необходимо подчеркнуть, что решение кода SSIDE или NSIDE обуславливает вызов неразбиваемой подпрограммы.

Применение запрограммированных вложенных вызовов не зависит от задания поля *Грань*:
- при задании (например, 1) задает грань подпрограммы, которая должна применяться к текущей грани
- если не задается: к текущей грани применяется та же грань, что подпрограммы.

Программирование осуществляется с операцией SSIDE, которую можно выбрать в группе ПОДПРОГРАММЫ вкладки **Операции**. Эта операция может вводиться только в текст подпрограммы (или макропрограммы) и становится рабочей когда подпрограмма вызывается (например, в грань-деталь):



- узел **"IF (..) ? (..) ? (..)":** условия применения задаются с прямым условием IF, до трех логических условий между двумя выражениями. Если условие оказывается ВЕРНЫМ, команда может интерпретировать применение вложенного вызова подпрограммы
- **Грань:** задание грани подпрограммы для применения во вложенном вызове. Если этот параметр не задан (т. е. поле пустое), операция не определяет никакого дополнительного применения, но отключает возможное решение *Автоматических вложенных вызовов*
- **Вложенная грань:** задает грань применения вложенного вызова

В примере на рисунке команда SSIDE программирует вложенный вызов в грани 4 вызывающей программы с применением грани 3 подпрограммы.

При оценке реальной возможности применения инструкции SSIDE учитываются другие задания:
- **Вложенные грани/Исключенные грани:** селективные настройки по включаемым или исключаемым граням. Если, например, поле **Исключенные грани** указывает грань 4, рассматриваемая здесь инструкция SSIDE не применяет запрограммированный вызов
- исключаются вложенные применения в:
  - гранях неуправляемых или вспомогательных (фиктивных или автоматических)
  - текущая грань (грань главного применения)
  - уже вложенная грань.

- Узел **"Геометрия":** может задавать точку применения конкретно для запрограммированного здесь вложенного вызова. Выбрать ячейку “XYZ” для подключения распознавания точки применения и задания координат в следующих полях (в абсолютном режиме программирование считается действительным для всех 3 координат).
  - незаданная координата (то есть: пустое поле) применяет значение 0.0
  - если поле не выбрано, решение точки применения вложенного вызова определяется главным вызовом (см.: Размещение вложенных вызовов)

В отличие от автоматических вложенных вызовов, механизм запрограммированных вложенных вызовов управляется без изменений на каждом уровне программирования.

Решение переменных <j> в запрограммированных вложенных вызовах

Даже создание запрограммированного вложенного вызова может использовать переменные <j>, что повышает производительность по сравнению со случаем автоматического вложенного вызова.

Рассмотрим пример:
- напишем подпрограмму ОДИН с операциями, заданными на гранях 1 и 3:
  - запрограммируем на грани 1:
    - строка 1: код <ЗАДАТЬ Jnn> зададим переменной j0 значение 100
    - строка 2: код <SSIDE-ПРИМЕНИТЬ ВЫЗОВ> зададим значение 3 в поле **Грань**, значение 4 в поле **Вложенная грань**
  - запрограммируем на грани 3:
    - строка 1: код <ОТВЕРСТИЕ> зададим координату X=j0. Отверстие окажется на координате x = 0.0.
    - строка 2: код <ЗАДАТЬ Jnn> зададим значение j0=j0+100;
    - строка 3: код <ОТВЕРСТИЕ> зададим координату X=j0. Отверстие окажется на координате x = 100.0.

- напишем программу PRG1 с операцией, заданной на грани 1:
  - строка 1: код <SUB> применяю подпрограмму: подпрограмма на грани 1 задает переменную j0 и создает вложенный вызов, запрограммированный на грани 3;
  - на грани 3: первое отверстие теперь выполняется на координате X=100, второе - на координате X=200.

Из примера видно, как запрограммированный вложенный вызов использует изначально переменные J как они были заданы на момент самого вложенного вызова (код SSIDE). Последующие новые задания прибавляются к исходной ситуации.

Применить геометрические трансформации

При применении подпрограммы могут активироваться некоторые геометрические трансформации, применимые в порядке, в котором они приводятся ниже.
Если подпрограмма применяет сложный код макропрограммы, для которого требуемая трансформация не разрешена, пользователь предупреждается сообщением об ошибке.

Инверсия
Инверсия подпрограммы вызывает инверсию порядка выполнения разработанных операций: последний блок становится первым и т. д.
В случае профиля трансформация определяет также геометрическую инверсию самих профилей и задания следующих параметров:
- коррекция на инструмент (правый или левый) каждой подготовки.
- выбор начального/конечного отрезков профиля (все также в подготовке) в случае задания правой или левой дуги.

При подключении из конфигурации TpaCAD применение этого инструмента к профилю может предусматривать зеркальную технологию.
При подключении из Конфигурации TpaCAD применение этого инструмента к ориентируемой подготовке может предусматривать трансформацию для осей ориентации (только если текущая грань плоская, т. е. не закруглена или задана как поверхность).

Вращение
Вращение подпрограммы задается в числовом поле с программированием угла вращения (в градусах или десятых долях градуса) в плоскости XY грани относительно оси X. Вращение осуществляется вокруг точки применения подпрограммы.
При подключении из Конфигурации TpaCAD применение этого инструмента к ориентируемой подготовке может предусматривать трансформацию для осей ориентации (только если текущая грань плоская, т. е. не закруглена или задана как поверхность).

Зеркальное отображение
Симметрия подпрограммы задается в двух полях выбора:

Зеркальное отображение по горизонтали: дает зеркальное отображение с поворотом вокруг вертикальной оси
Зеркальное отображение по вертикали: дает зеркальное отображение с поворотом вокруг горизонтальной оси
Если выбираются оба пункта, эти опции суммируются. Трансформация, только в случае только одного активированного выбора, инвертирует также следующие настройки:
- коррекция на инструмент (правый или левый) каждой подготовки только в случае только одного активированного выбора.
- выбор начального/конечного отрезков профиля (все также в подготовке) в случае задания правой или левой дуги.

Если требуется также поворот, он выполняется до симметрии.
При подключении из конфигурации TraCAD применение этого инструмента к профилю может предусматривать зеркальную технологию.
При подключении из Конфигурации TraCAD применение этого инструмента к ориентируемой подготовке может предусматривать трансформацию для осей ориентации (только если текущая грань плоская, т. е. не закруглена или задана как поверхность).

Масштаб (ветвь Растягивания)
Применяет коэффициент уменьшения или увеличения к подпрограмме и активизируется следующими пунктами:
- **Активировать:** при выборе активизирует применение трансформации;
- **Коэффициент:** коэффициент уменьшения или увеличения (мин. задаваемое значение: 0.001). Рассматриваются следующие ситуации:
  - менее 1: применяет уменьшение
  - более 1: применяет увеличение
  - =1: без изменений.
- **Трехмерная шкала:** при выборе активизирует применение в т. ч. на глубину (ось Z грани). Выбор этого пункта обязателен, если подпрограмма выполняет дуги, заданные в плоскости, отличной от xy.

Повторения при выполнении подпрограммы

Коды SUB обеспечивают два различных режима автоматического повторения подпрограммы:
- SUB реализует многократное применение со свободным повторение
- SMAT реализует многократное применение с повторением в матричной форме

Повторения со свободным распределением

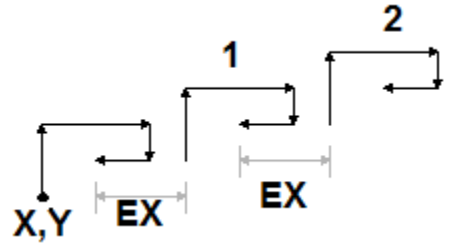
- **Повторения:** число повторений, которые должны добавляться к базовому применению. Минимальное значение для активизации повторений - 1
- **Сдвиги X, Y, Z:** сдвиги, применяемые при каждом повторении. Значения применяются относительно и суммируются при каждом повторении
- **Rel<-:** при выборе применяет сдвиги к точке начала применения предыдущего повторения. Можно принудительно задать координату Сдвига в абсолютном значении, вставляя непосредственно перед координатой **"a;"**.
- **Зацепление точки:** при выборе зацепляет каждое повторение за предыдущее. В этом случае игнорируются задания, относящиеся к Сдвигам X, Y, Z и к полю Отн <-
- **Сдвиг A(°):** задает приращение вращения, применяя его к каждому последующему повторению. Начальное значение определяется значением, заданным в поле вращения в базовом применении. Если, например, базовое вращение выполняет вращение на 30° и Сдвиг A(°) не задан, все повторения поворачиваются на 30°; если же Сдвиг A(°)=10°, тогда первое повторение поворачивается на 40°, второе - на 50° и т.д. для всех последующих.

Все трансформации с зеркальным отображением, заданные для базового применения, применяются и к повторениям. В частности, трансформации применяются также к соответствующим сдвигам:
- **Зеркальное отображение по горизонтали:** выполняет зеркальное отображение также заданного сдвига по горизонтальной оси
- **Зеркальное отображение по вертикали:** выполняет зеркальное отображение также заданного сдвига по вертикальной оси

Все трансформации инверсии и/или масштаба, заданные для базового применения, применяются и к повторениям.

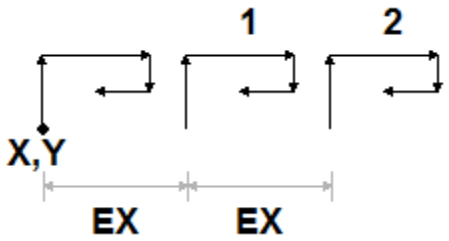
Рассмотрим ниже два примера со следующими заданными общими значениями:
зададим следующие значения:
- Повторения: 2
- Сдвиг X: 100
- Сдвиг Y: 0 (не задан)

- **Пример 1:**



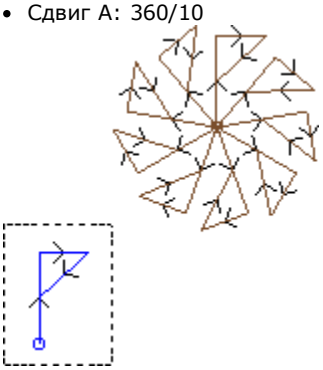
На рисунке показано развитие после задания **Rel<**- не активировано:
- **X,Y:** точка базового применения (может быть точка габаритного прямоугольника или точка применения, указанная в подпрограмме, или первая запрограммированная точка)
- **1:** соответствует первому повторению. Точка его применения добавляет 100 по X к конечному положению базового применения и 0 по Y
- **2:** соответствует второму повторению. Точка его применения добавляет 100 по X к конечному положению первого повторения и 0 по Y

Пример 2:



На рисунке показано развитие после задания **Отн<-** активировано:
- **X,Y:** точка базового применения
- **1:** соответствует первому повторению. Точка его применения добавляет 100 по X к точке P1 и 0 по Y
- **2:** соответствует второму повторению. Точка его применения добавляет 100 по X к положению точки применения первого повторения и 0 по Y

Пример 3:
В примере получается вертушка с неоднократным применением одного элемента
зададим следующие значения:
- Повторения: 9
- Rel<-: активировано
- Сдвиг A: 360/10



рамка на рисунке соответствует одному элементу, как запрограммировано в подпрограмме. Все повторения применяются к базовой точке применения и поворачиваются для завершения полного угла.

Повторения с распределением в матричной форме

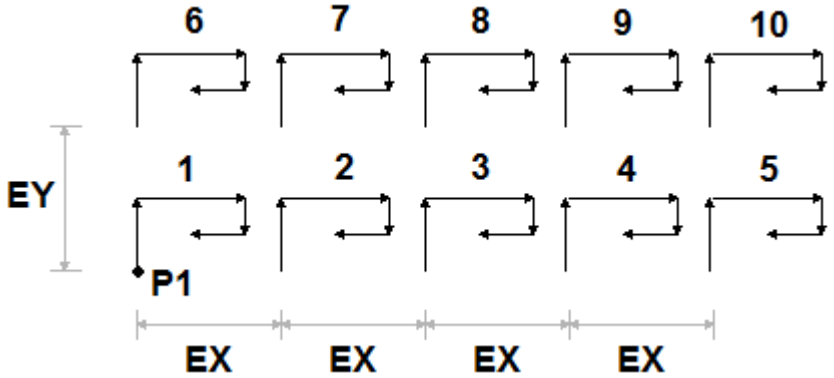
- **строки, Столбцы:** число строк и столбцов матрицы повторений. Минимальное значение для активизации повторения - 1 для обоих полей. Общее число выполняемых применений определяется произведением (строки * Столбцы), **включая** базовое применение Развертка на строках всегда ассоциируется с осью Y грани, а на столбцах - с осью X грани.
- **Расстояние между столбцами:** расстояние между столбцами матрицы
- **Расстояние между строками:** расстояние между строками матрицы
- **Rel<-:** при выборе применяет сдвиги строк и столбцов к точке начала применения предыдущего повторения. Можно задать размер Сдвиг в качестве абсолютного, вводя до этого размера **"a;"**.

Все трансформации с зеркальным отображением, заданные для базового применения, применяются и к повторениям. В частности, трансформации применяются также к соответствующим сдвигам:

- **Зеркальное отображение по горизонтали:** выполняет зеркальное отображение также заданного сдвига по горизонтальной оси
- **Зеркальное отображение по вертикали:** выполняет зеркальное отображение также заданного сдвига по вертикальной оси

Все трансформации инверсии и/или масштаба, заданные для базового применения, применяются и к повторениям.

Рассмотрим пример со следующими заданными значениями:
- строки: 2
- Столбцы: 5
- Расстояние между строками: 100
- Расстояние между столбцами: 100



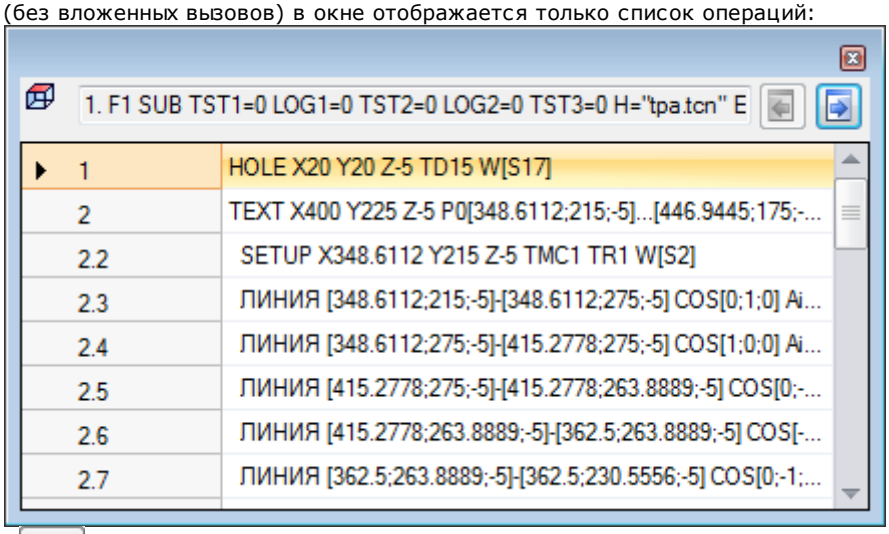
На рисунке показано развитие после задания **Отн<-** активировано:
- **P1:** точка базового применения
- **EX:** расстояние между столбцами;
- **EY:** расстояние между строками.

Общее число выполненных применений - (2 *5)=10, **включая** базовое применение.

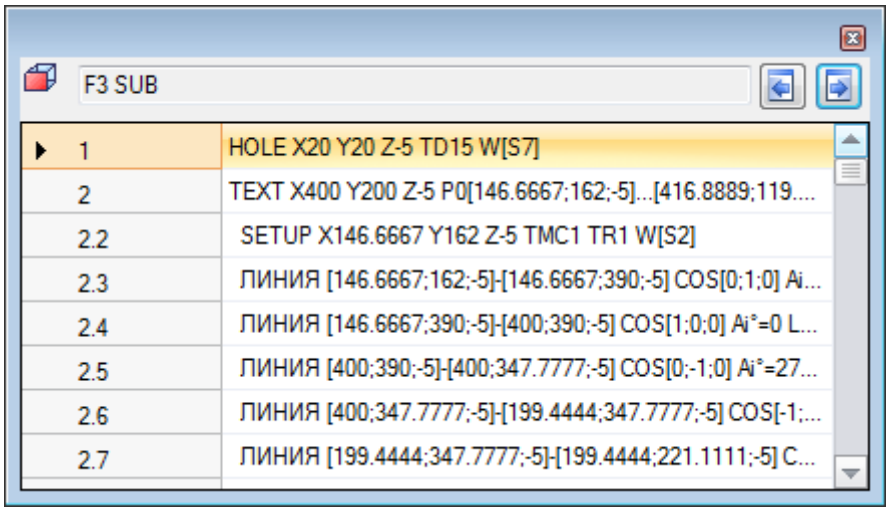
Показать развертку подпрограммы

Можно видеть разложение подпрограммы, щелкая правой кнопкой мыши на соответствующей ячейке **Текст ASCII**. Отображается список операций с геометрической и технологической информацией и заданными свойствами. Этот формализм аналогичен тому, который используется в строке состояния.

Этот рисунок относится к случаю применения и вложенных вызовов. В случае нормального вызова подпрограммы (без вложенных вызовов) в окне отображается только список операций:



Кнопки  и  позволяют переходить к предыдущему или следующему вызову. На рисунке представлен пример вложенного вызова.



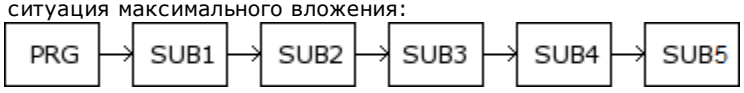
Встроить вызовы подпрограмм

Можно встроить сложные коды, заданные как макропрограммы или как вызовы подпрограммы, но с пределом вложения до 5 вызовов.

Предположим, что мы находимся на этапе редактора программы (PRG: базовый уровень программирования):

- на одной грани детали мы можем применить один вызов подпрограммы (SUB1)
- SUB1 может осуществлять вызовы других подпрограмм. Например, подпрограммы SUB2
- SUB2 может осуществлять вызовы другого/их подпрограмм. Например, подпрограммы SUB3
- SUB3 может осуществлять вызовы другого/их подпрограмм. Например, подпрограммы SUB4
- SUB4 может осуществлять вызовы другого/их подпрограмм. Например, подпрограммы SUB5
- SUB5 не может осуществлять вызовы другого/их подпрограмм

На рисунке показана ситуация максимального вложения:

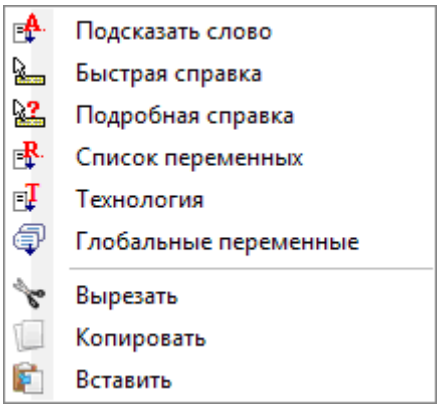


(SUB1, SUB2, SUB3, SUB4, SUB5) не обязательно должны быть подпрограммами, они могут также быть макропрограммами.
ВНИМАНИЕ: Программируемые инструменты также являются сложными кодами и как таковые учитываются при подсчете числа встроенных вызовов.
Максимально допустимое число вложений уменьшается автоматически один в случае редактора подпрограммы или макропрограммы.

Направляемое редактирование и функции с использованием дополнительных средств

При вводе или изменении операции, как правило, можно применять функцию редактора с использованием дополнительных средств. Ниже приводится перечень возникающих ситуаций.

Уже говорилось о возможности вызова меню немедленной справки функций, переменных аргументов и переменных, предусмотренных для параметрического программирования. И в этом случае речь идет о контекстуальном меню, которое открывается при щелчке правой кнопки мыши в поле редактирования параметра операции.



Состав меню меняется в зависимости от оценки различных типов:
- тип изменяемого параметра (например, если параметр - типа строки, пункт **Технология** в списке не отображается)
- доступность настроек (например, команды **Вырезать**, **Копировать**, **Вставить** проверяют возможность быть примененными)
- подключения среды (например, пункты **Технология** или **Глобальные переменные** отображаются только если они реализованы)
- Настройки среды и уровня доступа (например, доступ к уровню Производителя может отображать пункты, добавленные в список, или пополнять меню, связанные с некоторыми пунктами).

 **:** этот значок обозначает параметры, для которых можно активировать интерактивное считывание в графической зоне. Режим считывания аналогичен представленному в параграфе **Ввод геометрических элементов из меню "Начертить"**. Как уже указывалось в параграфах описания типов операции, считывание может быть автоматически связано с нескольким параметрам или быть действительно только для одного поля.

 **:** этот значок обозначает параметры задания технологии, инструмента /или электрошпинделя. При щелчке на этом значке открывается окно технологии с возможностью интерактивного выбора технологии для операции. В некоторых особых случаях предусматривается возможность делать выбор двух различных технологий, отличающихся между собой.

 **:** этот значок используется для выбора имени/имен операций, например, при использовании **Запрограммированных инструментов** (см. следующий параграф) или, в целом, - во всех операциях, позволяющих присваивать один или несколько параметров выбору операций, указанных Именем.

 : этот значок используется для открытия окна выбора некоторых свойств, например, Уровень и Конструкция. На грани-детали этот значок используется также для Грани применения.

 : этот значок, или другие значки, открывает окна, связанные со специальными настройками, например, выемками, выбором в списке шрифтов, выбором в списке подготовительных или точечных операций.

 **:** соотнесение двух значков используется, если параметр может выбирать операции, указанные именем, и выбирать код операции (например, код подготовки).

9.5 Запрограммированные инструменты

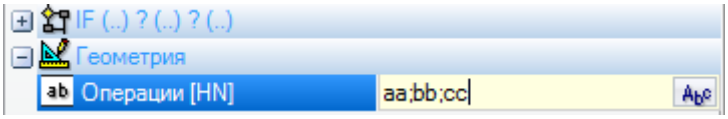


Для понимания того, что представляют собой операции, именуемые ***Запрограммированные инструменты,*** и каково их назначение, рассмотрим следующий пример.
Предположим, что нам необходимо выполнить выемку замкнутой зоны. Можно:
- применить инструмент Выемка непосредственно к профилю. При этом профиль, генерированный выемкой, не будет адаптироваться к возможным изменениям исходного профиля. В частности, выемка не учитывает возможные параметрические программирования исходного профиля

- сохранить профиль в подпрограмме и применить трансформацию, вызывая сложный код выемки. Эта возможность позволяет иметь параметрический профиль, т. е. изменяемый, и адаптировать выемку соответствующим образом, однако необходимо использовать подпрограмму.

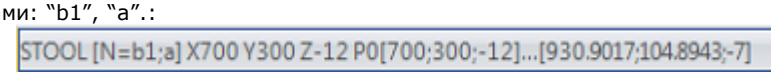
Операции, применяющие геометрические трансформации, объединяют положительные характеристики двух перечисленных выше процедур: в нашем примере они могут применять инструмент Выемка непосредственно к профилю, без прохождения через подпрограмму, но оставляют возможность менять изначальный профиль и, автоматически, - также выемку.
В группе ИНСТРУМЕНТЫ вкладки **Операции** находятся:
STOOL применяет общие трансформации (перенос, зеркальные исполнения, вращение, масштаб, повторения)
STOOL: EMPTY: выполняет выемки замкнутых зон
STOOL: SPLINE: генерирует кривые сплайн из полилиний
STOOL: RADIUS: генерирует профили для коррекции
STOOL: ATTAC: генерирует профили с применением точек крепления
STOOL: ZSTEP: генерирует профили с применением подач по оси развертки
STOOL: MULTI: генерирует профили с применением повторения по оси развертки
STOOL: LINE: генерирует профили для фрагментации и линеаризации
STOOL: ZLINE: генерирует профили с линеаризацией развертки по Z
STOOL: LINK: генерирует профили, соединяя отдельные профили
STOOL: STPLANE: поворачивает на плоскости прямоугольных координат.

К каким операциям применяются эти сложные операции?
Сложные операции могут применяться ко всем операциям, введенным ранее и имеющим **Имя (или Поле N),** заданное в пункте **Операции.** Одно имя могут иметь несколько операций.



В примере на рисунке сложная операция выемки применяется ко всем операциям, **Имя** которых - "aa", "bb" или "cc". Если текущая операция (например: STOOL) введена в строку 10 программы, поиск операций выполняется в первых 9 строках программы.
Порядок рассмотрения операций соответствует не порядку их перечисления в поле **Имя**, а номеру по порядку в программе. В приведенном примере операции с именем “bb” будут применяться первыми только если они отображаются в списке **до** операций с именем “aa” или “cc”.
Если имя задано на элементе профиля, оно учитывается задающим имя элементом до конца профиля. Поэтому рекомендуется задавать имя на первом элементе профиля (обычно - подготовке), если только вам не требуется рассматривать только одну часть этого профиля, который в этом случае оказался бы заданным без подготовки.
Синтаксис поля **Операции** - “Имя1;Имя2;….”. Имена должны быть разделены символом ';' (точка с запятой), а число символов не должно превышать 100.

И в этом случае допускается параметрическое программирование поля с переменными и параметрами типа строки. Решение параметрического синтаксиса должно дать строку типа “Имя1;Имя2;…”. В строке состояния приводится строка, получаемая из решения параметрического синтаксиса и дополнительного анализа, который удаляет недействительные задания. В примере на рисунке эта операция применяется к операциям с именами: “b1”, “a”.:



В поле **Операции** можно также задавать поиск с использованием символа ‘*’:
- если строка - =”*”, или указывает имя, равное “*” (например, “aa;*;bb”): трансформация применяется ко всем предыдущим операциям с заданным именем
- если поле Имя завершается символом ‘*’: поиск применяется к именам, которые начинаются с заданной части имени и заканчиваются одним или несколькими заданными символами. Например, “aa;sp*”: осуществляется поиск имен “aa” и всех имен, которые начинаются с “sp”.

Дополнительный анализ строки удаляет все имена с недействительными символами (напоминаем, только буквенно-цифровыми) или с неправильным использованием символа ‘*’ (напоминаем, что он может только завершать поле Имя). Изменения поля, выполненные на этом уровне, не определяют подачу сигнала диагностики.
Несколько примеров могут помочь понять соответствие между запрограммированным и приведенным в строке состояния:

“aa;*;cc”	интерпретирует: “*”	(имя “*” задает полный поиск)
“a*a;cc”	интерпретирует: “cc”	(символ ‘*’ не завершает имя)
“abc*;cc??;d1”	интерпретирует: “abc*;d1”	(имя “cc??” недействительно)
“aa,*,cc”	интерпретирует: “”	(разделитель неправильный)
“AA;BB”	интерпретирует: “aa;bb”	(приводит к строчным буквам)

Если поле Операции пустое, трансформация не задействует ни одну операцию. При выборе кнопки  открывается окно, в котором представляется список выбираемых имен. Если программирование операции осуществляется в режиме "грань-деталь", принимаются только операции, заданные на этой грани сложным кодом трансформации.
В любом случае, исключаются операции комментариев, логические команды, отличные от: цикла (IF... ELSEIF.. ELSE..ENDIF), EXIT или написания переменных J.
Для всех кодов, за исключением первого группы (STOOL):
- фильтруются сложные операции, которые оказываются неразбиваемыми
- рассматриваются только операции профиля или логические команды (IF ..ELSEIF.. ELSE.. ENDIF; EXIT; запись переменных J)

Поэтому для всех кодов, за исключением первого группы (STOOL), исключаются:
- сложные неразбиваемые операции (типичные примеры: прорези, вставки),
- клиентские точечные и логические операции, как запрограммированные напрямую, так и полученные в результате разбивки сложных операций (например, РЯД ОТВЕРСТИЙ для сверления).

Кроме того, все сложные операции применяются разбитыми.

Код STOOL применяет более общие критерии, т. к. имеет более широкое применение и не предназначен конкретно для особых трансформаций профилей:
- сложные операции не разбиваются, а применяются напрямую
- все клиентские точечные или логические операции задействуются трансформацией.

Если операции, соответствующие для индикации *Имени*, исключены в результате применения приведенных выше критериев, подается предупреждение, соответствующее ошибке номер: 225 - Запрограммированный инструмент: операции исключены.

Особое внимание должно уделяться применению *Программируемых инструментов* к логическим командам. Если необходимо понять ветвь IF .. ENDIF, задать требуемое имя всей структуре. Например, ситуация IF без закрытия инструкцией ENDIF определяет сигнал ошибки в результате неправильного соответствия между IF и ENDIF.

1	aa	HOLE X100 Y100 Z-10 TD10 TP1
2	aa	HOLE X132 Y100 Z-10 TD10 TP1
3	aa	HOLE X164 Y100 Z-10 TD10 TP1
4	aa	IF ESP1=r0
5	aa	HOLE X196 Y100 Z-10 TD10 TP1
6	aa	HOLE X228 Y100 Z-10 TD10 TP1
7		ENDIF
8	aa	HOLE X260 Y196 Z-10 TD10 TP1
9		STOOL HN="aa" Xf/2

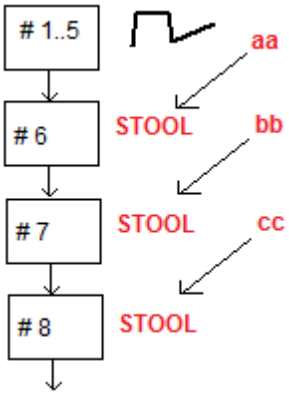
Рассмотрим пример, соответствующий рисунку:
- со строки 1 до строки 6 задано имя “aa”
- строка 7 не имеет заданного имени. Это инструкция ENDIF, закрывающая IF строки 4
- строка 8 также имеет имя “aa”.

STOOL программируется на строке 9 и применяется к операциям с именем “aa”. Компиляция строки определяет ситуацию ошибки: сигнализируется число ENDIF меньше IF.

Программные строки, не удовлетворяющие установленные логические условия, исключаются из трансформации.

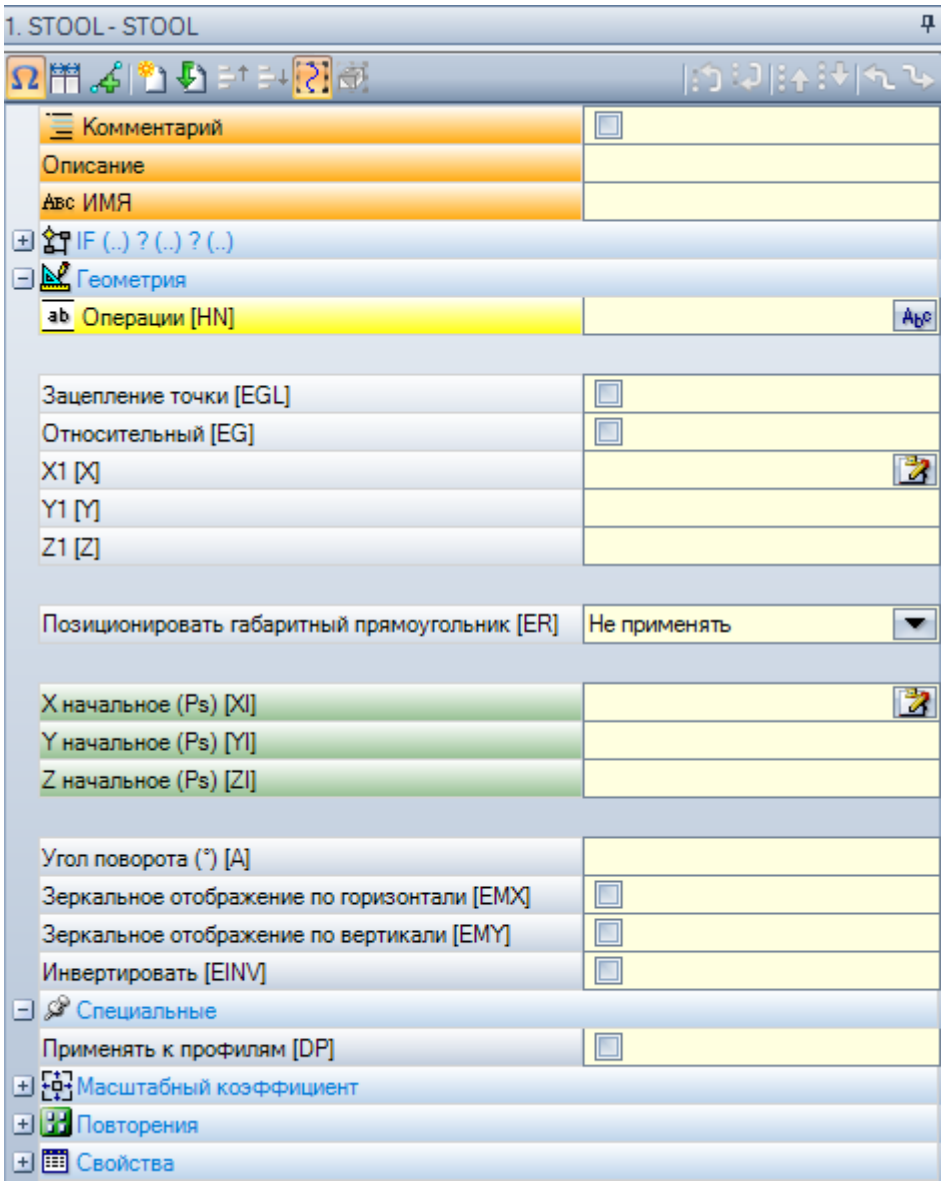
На исходных операциях, задействованных в применении Запрограммированного инструмента, обнуляется ненулевое значение Поля B (конструкция). Это позволяет выполнять только операции, производные из кода трансформации, а не исходные, как это обычно требуется.
Свойства, напрямую заданные операции STOOL, затем применяются к операциям, которые выполняются по нормальным критериям, применяемым ко всем кодам подпрограммы.
При редактировании макропрограммы применение Запрограммированных инструментов не определяет никакого немедленного выполнения. Фактическое выполнение будет обеспечиваться только на этапе применения этой макропрограммы.

Код STOOL - это сложный код, приравниваемый к вызовам подпрограммы или макропрограммы, и как таковой учитывается при оценке числа встроенных вызовов, в т. ч. с рекурсивными вызовами на один слой.
Рассмотрим в этой связи рисунок:



#1..5 - это программные строки, заданные с именем **aa**: разместим профиль
#6 - это следующая строка, код **STOOL**: применяется к операциям **aa** и в свою очередь задается с именем **bb**. Эта операция применяет первый слой вложения
#7 - это следующая строка, код **STOOL**: применяется к операции **bb** и в свою очередь задается с именем **cc.** Эта операция применяет второй слой вложения
#8 - это следующая строка, код **STOOL**: применяется к операциям **cc.** Эта операция применяет третий слой вложения.

Рассмотрим более подробно код STOOL (применяет общие трансформации: перенос, зеркальные исполнения, вращение, масштаб, повторения). Рассмотрение других кодов приводится в параграфах, в которых рассматриваются соответствующие этим кодам инструменты.



Эта операция подобна коду SUB применения подпрограммы. Теперь поле Операции заменяет поле выбора подпрограммы.
Помимо координат точки применения (X1; Y1; Z1), добавлены 3 координаты (X начальная, Y начальная, Z начальная) для определения вспомогательной точки с функцией:

- центр вращения;
- ось симметрии.

Узел применения повторений напрямую предлагает выбор из двух возможных схем: свободная или матричная.
Для полного анализа операции следует обращаться к справке, которая вызывается напрямую из окна задания данных операции.

Продвинутое использование Запрограммированных инструментов

Использование, которое мы до данного момента представляли для **Программируемых инструментов**, на этапе программирования представляется в виде списка операций. Возьмем более сложный пример, который мы рассмотрим подробно:

- создадим программу (например, PRG1)
- введем простую операцию сверления (например, HOLE), с заданным диаметром от 10 до 150 мм, конструкции, с именем “aa”
- вызывается подпрограмма (SUB1) и имя “aa” передается общедоступной переменной “r” типа строки.

Цель такова, чтобы SUB1 могла:
- отобрать операцию "aa"
- проанализировать ее и понять, может ли оснащение станка выполнить сверление или же запрограммированный диаметр требует выполнения цикла фрезерования (т. е. выемки круга на запрограммированную глубину).

С учетом сказанного выше это не представляется возможным: очевидно, что проблема имеет решение, которое мы рассмотрим далее.

- Мы уже говорили, что подпрограмма SUB1 имеет общедоступную переменную "r" типа строки, предположим r0: при вызове SUB1 переменная подпрограммы записывается ="aa";
- подпрограмма SUB1 может отобрать операцию "aa" с использованием кода STOOL и задания пункта **Операции** на

  -*r0

  - если не принимать во внимание начальный знак минус (-), значение остальной части ("*r0") - формат параметрического программирования параметра типа строки
  - начальный знак минус (-) интерпретируется кодом STOOL как запрос на поиск операций не до себя самого (т. е. не в подпрограмме SUB1), а до вызова подпрограммы SUB1 (т. е. в программе PRG1)
- если код STOOL запрограммирован как Конструкция, операция "aa" вызывается программируемым инструментом еще как Конструкция;
- теперь подпрограмма SUB1 должна программировать интерпретацию операции, вызываемой из строки STOOL, и решать, что делать.

Что делать и как зависит от конкретной проблемы и не представляет для нас основного интереса. Важнее предоставить несколько указаний по поводу того, как интерпретировать операцию, которую вызывала строка STOOL: в нашем примере - простое сверление с кодом HOLE.

Как можно проанализировать операцию, созданную из строки STOOL?
И в этом случае программирование помогает нам универсальными функциями геометрической библиотеки. В частности:
- функция **geo[param; ..]** позволяет считывать основную информацию строки STOOL, например, число созданных операций. Для этого:
  - зададим имя операции STOOL, например: "tt"
  - используем функцию с синтаксисом: **geo[param;"tt";"#list"]**, например, в логической команде цикла (IF .. ELSE) или задания переменной "j"
  - функция вернет ненулевое значение, если операция с именем "tt" создала операции

- функция **geo[lparam; ..]** позволяет считывать основную информацию каждой операции, которую создала, например, строка STOOL. Для этого:
  - используем функцию с синтаксисом: **geo[lparam;"tt";"#tip",1]**, например, в логической команде цикла (IF .. ELSE) или задания переменной "j". Функция вернет цифровое значение, соответствующее типу первой операции, которую создала операция с именем "tt". В частности: значение 0 соответствует типу отверстия;
  - используем функцию с синтаксисом: **geo[lparam;"tt";"td",1]** для считывания значения диаметра сверления, запрограммированного на этой операции.

Раздел, посвященный **Параметрическому программированию**, в описательной части использования функции **geo[lparam; ..]** говорит нам, что можно расширить контроль на структуру операции STOOL также в случаях разработки на уровнях выше первого, позволяя реализовывать подпрограммы и макропрограммы, намного более сложные и комплексные по сравнению с представленными до сих пор.

9.6 Автоматические грани



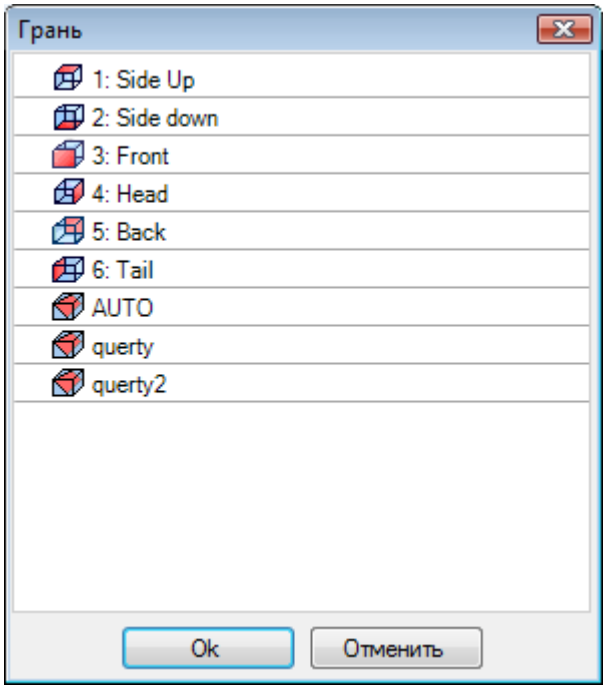
Это необязательная функция.
Автоматические грани - это грани, созданные непосредственно на этапе программирования грани-детали. Нумерация граней выполняется автоматически по порядку (от 101 до 500). Автоматические грани отображаются только на грани-детали.
Создание автоматической грани позволяет затем применять на ней операции, всегда при программировании только грани-детали. Невозможно:
- получить доступ непосредственно к отдельному виду автоматической грани
- создать и/или задать операции на автоматической грани с грани, отличной от грани-детали

Автоматическая грань не может выбираться номером напрямую присвоенным (автоматически) грани: можно получить доступ к последней заданной автоматической грани или грани с заданным именем.
Если не заданы или не используются имена граней, механизм использования автоматических граней соответствует следующей схеме:

- ...
- задание автоматической грани (автоматический номер: первый свободный, например, - 105);
- применение операции на автоматической грани (последней созданной -> 105);
- ...
- задание автоматической грани (автоматический номер: первый свободный, например, - 106);
- применение операции на автоматической грани (последней созданной -> 106);
- ...

Таким образом, в любой точке программы грани-детали имеется только одна конкретная автоматическая грань, последняя созданная ранее. В этой связи используется общая надпись *применении на автоматической грани*.

Вернемся к примеру открытия окна выбора, представленному ранее для Поля F:



По порядку в нем представлены:
- 6 граней базовой детали (в этом примере они все могут выбираться)
- две фиктивные грани (номер 7 и 8)
- пункт AUTO (АВТ.), соответствующий применению автоматической грани, заданной до текущей программной строки, за исключением вспомогательных граней. Автоматическая грань может быть создана также на предыдущем расширенном уровне. Последняя автоматическая грань может быть также операцией NSIDE, примененной в подпрограмме или макропрограмме.
- две строки (последние в списке), соответствующие прямому выбору одной из автоматических граней, заданных ранее с именем. В этом примере имеется две автоматические грани с именами, соответственно, "qwerty2" и "qwerty". В случае граней с повторяющимися именами будет действительна последняя заданная ранее. Из перечня исключаются вспомогательные грани. В случае автоматической грани, заданной на расширенном уровне (т. е. с вызовом подпрограммы или макропрограммы), она приводится в списке только если соответствующая операция в списке грани-детали имеет заданное имя. В этом случае последняя заданная грань приводится с тем же именем.

Создание автоматической грани осуществляется операцией NSIDE, которая выбирается в группе ЛОГИЧЕСКИЕ ИНСТРУКЦИИ вкладки **Операции**. Операцию можно вставить:
- непосредственно в грань-деталь или
- в текст подпрограммы (или макро): в этом случае она становится активной, когда подпрограмма вызывается в грань-деталь.

- Узел **"IF (..) ? (..) ? (..)":** условия применения задаются с прямым условием IF, до трех логических условий между двумя выражениями. Если условие оказывается ВЕРНЫМ, команда интерпретирует создание автоматической грани.
- **P0 {} P1 {} P2 {}:** открывает окно, идентичное окну задания фиктивных граней для определения ребер автоматической грани. Задание автоматической грани отражает режимы, предусмотренные на этапе задания фиктивной грани:
  - исходная грань
  - три точки задания плоскости (P0, P1, P2)

- направление оси Z
- толщина
- режим графического представления
- дополнительные параметры
- задание в качестве **вспомогательной грани** (может использоваться только в качестве исходной грани для задания последующей автоматической непрограммируемой грани)

Автоматическая грань не может быть назначена в качестве *поверхности*.
Если операция NSIDE вызывается из грани, отличной от грани-детали, оказывается невозможным задать в качестве исходной грани фиктивную грань. Графическое представление на грани-детали включает также автоматические грани за исключением вспомогательных граней.

Запрограммированные вложенные вызовы
К автоматической грани может применяться запрограммированный вложенный вызов.
О коде SSIDE:
- для запроса применения на автоматической грани необходимо оставить незаданным поле **Вложенная грань**.

В этом случае применение будет ссылаться на последнюю созданную автоматическую грань (перечисленное выше).

9.7 Ввод геометрических элементов из меню "Начертить"

Команды для вставки геометрических элементов напрямую находятся во вкладке **Применить** группы **Начертить.**
Функции команд определяются заданием поля отображения программы (в строке состояния): при неактивном пункте сообщение информирует о невозможности продолжения активации команды.
Выбрать одну из команд группы для интерактивного ввода геометрических элементов. В режиме "грань-деталь" пункты меню Чертеж отключены, если активно каркасное представление, при текущей операции, заданной на нереальной грани

	**Точка:** вставляет точечную операцию. Этот пункт имеется в меню, если задан код по умолчанию для точечной операции (для текущей грани или независимо от грани). (см.: ***Индивидуализировать->Технология->Коды по умолчанию*** из меню **Применение**).
	**Линия:** вставляет линейный сегмент. Этот пункт имеется в меню, если задан код операции COPL01. Вставляет линейный сегмент по указаниям, приведенным в зоне командной строки: **Начальная точка отрезка** **Конечная точка сегмента** Если операция COPL01 не предусматривает параметры задания начальной точки отрезка в плоскости xy, должно быть возможно зацепить линейный сегмент за предыдущий элемент в программном списке, а начальная точка отрезка позиционируется автоматически на точку зацепления
	**Дуга (центр, начало, конец):** вставляет дугу. Этот пункт имеется в меню, если задан код операции COPA01, который управляет параметрами задания начальной точки отрезка в плоскости xy. Вставляет дугу по указаниям, приведенным в зоне командной строки: **центр дуги** **радиус;** **начальный угол;** **конечный угол.** Вращение дуги может определяться на этапе задания конечного угла. Если нажать кнопку **[I]** или если из контекстуального меню выбрать пункт **Инвертировать вращение,** направление вращения меняется с вращения по часовой стрелки (CW) на вращение против часовой стрелки (CCW) и наоборот. Если выбирается эта команда, в зоне команд отображается надпись **[CW]** при вращении по часовой стрелке и **[CCW]** при вращении против часовой стрелки.
	**Дуга (три точки):** вставляет дугу, заданную по трем точкам. Этот пункт имеется в меню, если задан код операции COPA04. Вставляет дугу по указаниям, приведенным в зоне командной строки: **начальная точка дуги** **точка прохода по дуге** **конечная точка дуги**. Если операция COPA04 не предусматривает параметры задания начальной точки отрезка в плоскости xy, должно быть возможно зацепить дугу за предыдущий элемент в программном списке, а начальная точка отрезка позиционируется автоматически на точку зацепления
	**Дуга (начало, конец, радиус)**: вставляет дугу, заданную по двумя точкам, и радиус. Этот пункт имеется в меню, если задан код операции COPA11, который управляет параметрами

	задания начальной точки отрезка в плоскости xy. Вставляет дугу по указаниям, приведенным в зоне командной строки: **начальная точка дуги** **конечная точка дуги** **радиус дуги.** Поворот дуги может определяться на этапе задания радиуса. Если нажать кнопку **[I]** или если из контекстуального меню выбрать пункт **Инвертировать вращение,** направление вращения меняется с вращения по часовой стрелки (CW) на вращение против часовой стрелки (CCW) и наоборот. Если выбирается эта команда, в зоне команд отображается надпись **[CW]** при вращении по часовой стрелке и **[CCW]** при вращении против часовой стрелки.
	**Круг:** вставляет круг. Этот пункт имеется в меню, если задан код операции COPA45, который управляет параметрами задания начальной точки отрезка в плоскости xy Вставляет круг по указаниям, приведенным в зоне командной строки: **центр дуги** **радиус**. Вращение круга задано по часовой стрелке.
	**Круг (2 точки)**: вставляет круг, заданный по двумя точкам. Этот пункт имеется в меню, если задан код операции COPA46, который управляет параметрами задания начальной точки отрезка в плоскости xy Вставляет круг по указаниям, приведенным в зоне командной строки: **точка начала (и конца) круга** **противоположная точка на круге.** Вращение круга задано по часовой стрелке.
	**Круг (3 точки)**: вставляет круг, заданный по трем точкам. Этот пункт имеется в меню, если задан код операции COPA46, который управляет параметрами задания начальной точки отрезка в плоскости xy Вставляет круг по указаниям, приведенным в зоне командной строки: **точка начала (и конца) круга** **первая точка прохода по кругу** **вторая точка прохода по кругу.** Вращение круга задано по часовой стрелке.
	**Винт**: вставляет винт. Этот пункт имеется в меню, если задан код операции COPA48, который управляет параметрами задания начальной точки отрезка в плоскости xy. Вставляет винт по указаниям, приведенным в зоне командной строки: **центр винта** **Конечная глубина, выводимая как расстояние от центра по оси глубины** **радиус/ точка начала.** Вращение винта задается по часовой стрелке, число повторений равно 5.
	**Спираль**: вставляет спираль. Этот пункт имеется в меню, если задан код операции COPA49, который управляет параметрами задания начальной точки отрезка в плоскости xy. Вставляет спираль по указаниям, приведенным в зоне командной строки: **центр винта** **Конечная глубина, выводимая как расстояние от центра по оси глубины** **радиус/ точка начала** **конечный радиус.** Вращение спирали задается по часовой стрелке, число повторений равно 5.
	**Эллипс:** вставляет эллипс. Этот пункт имеется в меню, если задан код операции COPA42, который управляет параметрами задания начальной точки отрезка в плоскости xy Вставляет эллипс по указаниям, приведенным в зоне командной строки: **центр** эллипса **крайняя точка** оси **крайняя точка** на второй оси. Вращение эллипса задано по часовой стрелке.
	**Эллипс (3 точки)**: вставляет эллипс, заданный по трем точкам. Этот пункт имеется в меню, если задан код операции COPA42, который управляет параметрами задания начальной точки отрезка в плоскости xy Вставляет эллипс по указаниям, приведенным в зоне командной строки: **первая крайняя точка** оси

	вторая крайняя точка этой же оси **расстояние центра от второй оси.** Вращение эллипса задано по часовой стрелке. Исходная точка эллипса определяется на точке смены квадранта по большей оси.
	**Дуга эллипса:** вставляет дугу эллипса. Этот пункт имеется в меню, если задан код операции COPA43, который управляет параметрами задания начальной точки отрезка в плоскости xy Вставляет дугу эллипса по указаниям, приведенным в зоне командной строки: **центр** эллипса **крайняя точка на первой оси** **крайняя точка на второй оси** **начальный угол** **конечный угол** Вращение дуги эллипса может определяться на этапе задания конечного угла. При нажатии кнопки **[I]** или из контекстуального меню при выборе пункта **Инвертировать вращение,** меняет направление вращения с направления по часовой стрелки (CW) на направление против часовой стрелки (CCW) и наоборот. Выбор команды сигнализируется в зоне команд надписями **[CW]** при вращении по часовой стрелке и **[CCW]** при вращении против часовой стрелки.
	**Многоугольник:** вставляет многоугольник. Этот пункт имеется в меню, если задан код операции COPL17, который управляет параметрами задания начальной точки отрезка в плоскости xy Вставляет многоугольник по указаниям, приведенным в зоне командной строки: число **сторон** (от 3 до 99): значение задается диалоговом окне **центр** многоугольника **начальное ребро**.
	**Прямоугольник:** вставляет прямоугольник. Этот пункт имеется в меню, если задан код операции COPL16, который управляет параметрами задания начальной точки отрезка в плоскости xy Вставляет прямоугольник по указаниям, приведенным в зоне командной строки: **первая вершина** прямоугольника **вторая вершина** прямоугольника, противоположная первой.
	**Прямоугольник (Центр, Р):** вставляет прямоугольник путем задания центра и ребра. Этот пункт имеется в меню, если задан код операции COPL16, который управляет параметрами задания начальной точки отрезка в плоскости xy Вставляет прямоугольник по указаниям, приведенным в зоне командной строки: **центр** прямоугольника **вершина** прямоугольника.
	**Прямоугольник (Сторона, Р):** вставляет прямоугольник путем задания стороны и точки для ограничения противоположной стороны. Прямоугольник обычно может ориентироваться. Этот пункт имеется в меню, если задан код операции COPL16, который управляет параметрами задания начальной точки отрезка в плоскости xy Вставляет прямоугольник по указаниям, приведенным в зоне командной строки: **первая вершина** прямоугольника **крайняя точка на первой стороне** точка, завершающая первую сторону **крайняя точка на второй стороне** точка, определяющая положение стороны, противоположной заданной
	**Звезда:** вставляет звезду. Этот пункт имеется в меню, если задан код операции COPL25, который управляет параметрами задания начальной точки отрезка в плоскости xy Вставляет звезду по указаниям, приведенным в зоне командной строки: число **сторон** (от 3 до 99): значение задается диалоговом окне **центр** многоугольника **начальное ребро** для задания внешнего радиуса **внутренний радиус** Выражения «внутренний радиус» и «внешний радиус» не должны интерпретироваться абсолютно: отношение между двумя радиусами может инвертироваться.
	**Полилиния:** вставляет полилинию, которая может задаваться последовательностью смежных линейных отрезков и/или дуг, заданных для точек, по указаниям, приведенным в зоне командной строки. Этот пункт имеется в меню, если задан код операции COPL01. В частности, можно: • переключаться с линии на дугу, соответственно, клавишами: **[L]**и **[A]** или из контекстуального меню при выборе, соответственно, пунктов **Перейти к линии** и **Перейти к дуге**

	• замкнуть отрезок на начальной точке полилинии клавишей **[C]** или из контекстуального меню при выборе команды **Замкнуть на начальной точке** Если операция COPL01 не предусматривает параметры задания начальной точки отрезка в плоскости xy, должно быть возможно зацепить дугу за предыдущий элемент в программном списке, а начальная точка отрезка позиционируется автоматически на точку зацепления. Переход на дугу имеется в меню, если задан код операции COPA04.
	**Разметка**: вставляет геометрический элемент типа разметки. Этот пункт имеется в меню, если задан код операции COPL24, который управляет параметрами задания начальной точки отрезка в плоскости xy Вставляет элемент по указаниям, приведенным в зоне командной строки: **начальная точка отрезка** **конечная точка отрезка** **начальная касательная отрезка** (поймать графический прицел, устанавливая мышь поблизости) **конечная касательная отрезка** (поймать графический прицел, устанавливая мышь поблизости) завершить ввод кнопкой **[ENTER]**.

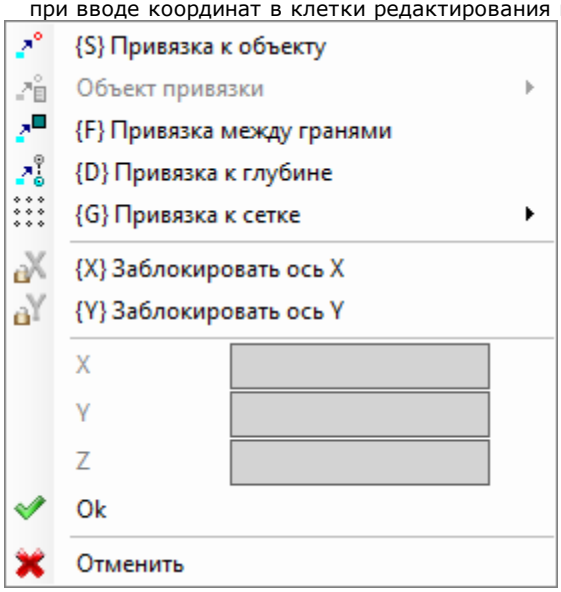
На этапе вычерчивания курсор мыши персонализируется и в зоне команд приводится индикация элемента (положение на плоскости xy, радиус, угол и т.д.) который должен вставляться.

Этап вставки остается активным до тех пор, пока он не будет отменен кнопкой **[ESCAPE]** или командой **Отменить** из контекстуального меню.


Можно перемещать мышь небольшими шагами, в одном из двух направлений, выбирая кнопку направления (стрелки: вправо, влево, вверх, вниз). Перемещение прекращается при отпускании кнопки. Примененный шаг, в пикселах, равен двойному значению, заданному в *Персонализации TpaCAD* как **Минимальный порог перемещения мыши**. Подчеркиваем, что данное перемещение в целом не связано с осями координат грани или детали, а соответствует горизонтальному или вертикальному положению на экране.

Всегда можно открыть контекстуальное меню правой кнопкой мыши: состав меню может меняться в зависимости от выбранного объекта чертежа.

Например, можно задать координаты (x,y,z):
- перемещая курсор мыши в нужное положение и щелкая левой кнопкой
- из контекстуального меню, которое открывается при нажатии правой кнопки мыши в графической зоне, при вводе координат в клетки редактирования и подтверждая выбором пункта **Ok.**



Обязательное условие задания выделено изображением слева от текста (восклицательный знак) и другим цветом фона поля редактирования.
Таким же образом можно задать значение другими методами. Рассмотрим радиус круга:
- при перемещении курсора мыши вычерчивается радиус от центра круга до нужного положения
- контекстуального меню: значение радиуса вводится напрямую.

Для стирания последнего выполненного в поле задания нажать значок  , расположенный рядом с этим полем.

Наличие и использование контекстуального меню чертежа, а также приведенные ниже положения по режиму привязки действительны не только для этапа вставки из меню чертежа, но и для других интерактивных режимов вставки и/или изменения программы: непосредственно из окна ввода данных текущей операции или применения инструментов.

Рассмотрим другие команды, предусмотренные в контекстуальном меню:
'P'= Использовать последнюю точку доступна только на этапе ввода начальной точки некоторых геометрических элементов, например, линии или дуги по трем точкам, и задает точке координаты последнего вставленного элемента. Первая точка отрезка может зацепляться за предыдущий элемент в программном списке. Зацепление выполняется с продолжением профиля если только это реально возможно. Если, например, предыдущий элемент является точкой, при активизации этой команды происходит позиционирование начальной точки линейного отрезка в положение точки, но как начало нового профиля. Подключение или отключение может осуществляться также при нажатии кнопки **'P'**.
'S'= Привязка к объекту требуемое позиционирование вставляется по координатам, определенным опцией, активированной в меню, которое открывается при выборе опции **Объект привязки:**

- **Запрограммированная точка:** позиционирование выполняется на ближайшую к курсору запрограммированную точку. Может также подключаться сочетанием клавиш **[CTRL+P]**.
- **Точка в окрестности:** позиционирование выполняется на ближайшей к курсору точке (например, положение определяется по дуге или линейному сегменту). Может также подключаться сочетанием клавиш **[CTRL+N]**.
- **Средняя точка:** позиционирование выполняется на среднюю точку дуги или на линейный сегмент. Может также подключаться сочетанием клавиш **[CTRL+M]**.
- **Центр дуги:** позиционирование выполняется в центр дуги, круга или эллипса. Может также подключаться сочетанием клавиш **[CTRL+C]**.
- **Точка пересечения:** позиционирование выполняется на пересечение отрезков (за исключением отрезков Разметки L24). Может также подключаться сочетанием клавиш **[CTRL+I]**.
- **Точка на перпендикуляре:** позиционирование выполняется на линейный сегмент, дугу, круг или эллипс начиная с предыдущей точки применения в направлении, перпендикулярном этому отрезку. Может также подключаться сочетанием клавиш **[CTRL+O]**.
- **Точка на касательной:** позиционирование выполняется на линейный сегмент, дугу, круг или эллипс начиная с предыдущей точки применения в направлении, касательном этому отрезку. Может также подключаться сочетанием клавиш **[CTRL+T]**.
- **Точка на смене квадранта:** позиционирование выполняется на дуге, круге или эллипсе в точке смены квадранта, ближайшей к положению курсора. Может также подключаться сочетанием клавиш **[CTRL+Q]**.
- **Ребро грани:** позиционирование выполняется на краю или на ребре грани ближе всего к положению курсора. В частности, позиционирование оказывается точно на одном из ребер грани, когда указатель находится за пределами габаритного прямоугольника грани или очень близко к нему. Может также подключаться сочетанием клавиш **[CTRL+E]**.

На строке состояния указывается тип привязки, подключенного на этапе черчения. Подключение привязки к запрограммированному объекту ограничивается данным вводом и при необходимости должно вызываться повторно для последующего ввода.

'F'= Привязка между гранями при привязке, активной на запрограммированном объекте, поиск объекта привязки распространяется на все отображенные операции, даже если они заданы на другой грани. Подключение/отключение привязки между гранями может выполняться также нажатием клавиши **'F',** а выбор команды сигнализируется в зоне команд надписями **[F On]** или **[F Off]**. Привязка между гранями является реально рабочей при обнаружении действительного объекта привязки вблизи положения мыши или если положение мыши оказывается как минимум в пределах зоны представления грани. Подразумевается, что поиск изначально ограничивается определенной графической зоной, отцентрованной в точке щелчка (отображение графической зоны поиска приводится ниже): если в этой зоне не обнаруживается никакого объекта, но положение мыши оказывается в пределах зоны представления грани, начинается поиск среди всех операций в программном списке этой грани. Положительный исход этого поиска решает выбор привязки между гранями, в противном случае поиск решает остальные выбранные режимы привязки.
При выборе типа привязки точки на перпендикуляре, касательной или ребре грани все равно применяется привязка **Запрограммированной точки**. Выбор привязки между гранями подготавливает активизацию также привязки к глубине.
В режиме "грань-деталь" необходимо выбрать Привязку между гранями для подключения поиска среди всех операций, запрограммированных в режиме "грань-деталь", даже на грани, отличной от текущей. В этом случае, если необходимо ограничить поиск только операциями "грань-деталь", можно отключить отображение запрограммированных списков на других гранях.
Привязка между гранями, в любом случае, доступна только в режиме Professional.

'D'= Привязка к глубине с активной привязкой к запрограммированному объекту, требуемое позиционирование определяется также составляющей глубины. Подключение/отключение привязки к глубине может выполняться также нажатием клавиши **'D',** а выбор команды сигнализируется в зоне команд

надписями **[ZETA On]** или **[ZETA Off]**. Привязка к глубине является реально рабочей, если это позволяет объект привязки. В частности, исключаются привязки к точке: пересечения, на перпендикуляре или на касательной.

'G'= Привязка к сетке требуемое позиционирование определяется как вершина сетки, ближайшая к точке щелчка мыши. Подключение/отключение привязки к сетке может выполняться также нажатием клавиши **'G',** а выбор команды сигнализируется в зоне команд надписями **[G On]** или **[G Off]**. Привязка к сетке может активизироваться даже если сетка не отображается, в противном случае привязка применяется к отображенной (решеткой или точками) сетке.
Если привязка применяется к сетке решеткой, можно применить дополнительную опцию, выбирая один из трех вариантов, предусмотренных в меню пункта привязки:

- Привязка к сетке: притягивается на точках сетки (функционирование по умолчанию)
- Точка на горизонтальной прямой: притягивается на точках сетки в вертикальной привязке, а затем перемещается дискретно по определенной таким образом горизонтальной прямой.
- Точка на вертикальной прямой: притягивается на точках сетки в горизонтальной привязке, а затем перемещается дискретно по определенной таким образом вертикальной прямой.

'X'= Блокировать ось x и **'Y'= Блокировать ось Y** не допускают перемещение курсора в направлении X или Y. Подключение/отключение может осуществляться также нажатием клавиш **[X]** или **[Y]**. Блок движения применяется к текущей операции или, если подключена привязка, суммируется к выбранной характеристике привязки. Блокировка в одном направлении автоматически разблокировывает другое.

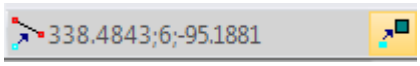
'T'=Касательный отрезок активизирован при вводе отрезка, продолжающего профиль, и заставляет отрезок (линейный или дугу) выйти по касательной из этого профиля. Порядок задания геометрического отрезка следующий:

- дуга определяется позиционированием конечной точки
- линейный отрезок определяется позиционированием конечной точки, в любом случае, приведенным на направление касательной, или давая длину этого отрезка.

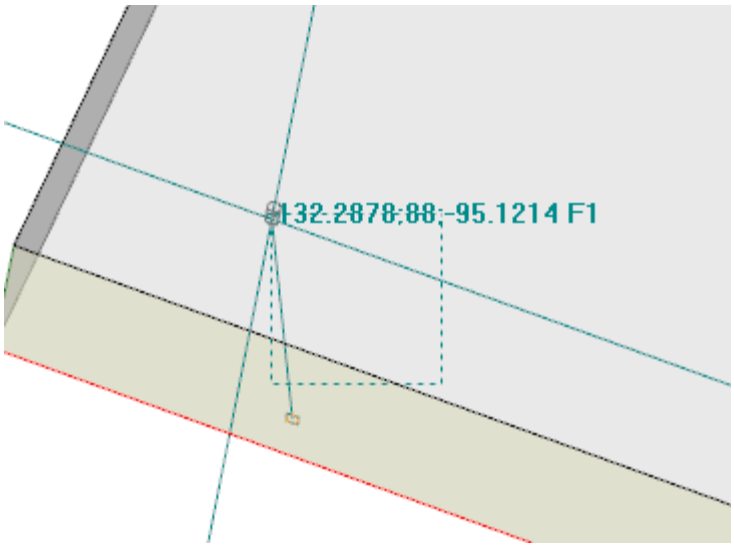
Подключение или отключение может осуществляться также при нажатии кнопки **'T'**.

'Z'= Отменить последний отрезок имеется в случае ввода полилинии и отменяет последний введенный отрезок. Подключение или отключение может осуществляться также при нажатии кнопки **'Z'**.

На этапе вставки из меню Чертеж или, в любом случае, - в интерактивном режиме, в строке состояния приводится информация по текущему считыванию:



определенное положение для мыши после процедур привязки. В частности: изображение слева указывает тип решенной привязки (например, запрограммированная точка, средняя точка и т. д.),  выбор ячейки указывает, что была решена привязка на гранях.
В интерактивной процедуре при запросе привязки к объекту, можно отобразить прицел, используемый для решения привязки, выбирая ячейку  в строке состояния. На рисунке поясняется функционирование:



выделенная пунктиром зона указывает размер, до которого была расширена графическая зона поиска объекта привязки. Ее можно указать как прицел привязки. Как уже говорилось, графическая зона поиска ограничена: она начинается от минимальной размерами в несколько пикселов и расширяется до

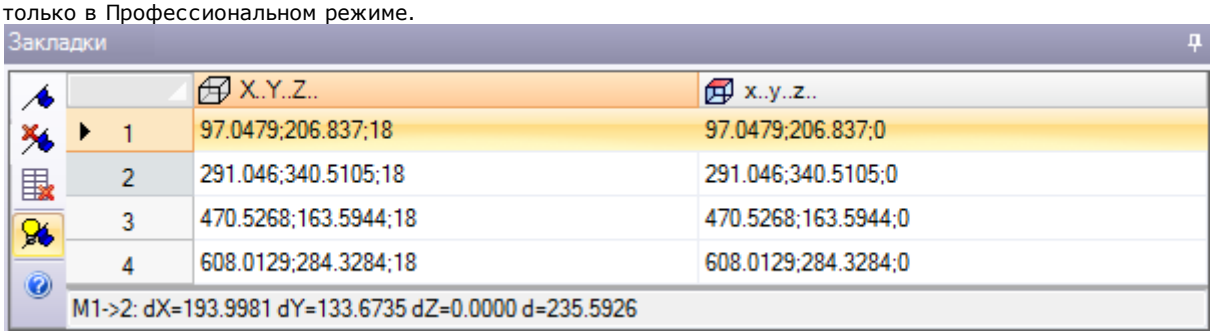
максимальной зоны, пока не будет найдено соответствие (см. также Индивидуализация TpaCAD Виды Мышь).
Рисунок может соответствовать случаю переноса отверстия, запрограммированного на грани 3, на отверстие на грани 1: на графике показано, как была расширена зона поиска до включения отверстия в грани 1. Положение мыши соответствует центральной точке прицела привязки.

Много было сказано о выбираемых режимах привязки: объекта, глубины, грани.
В любом случае, очевидно, что самое элементарное условие для интерактивного размещения действует без какой-либо активной привязки. В этих условиях положение графического указателя в двухмерном измерении приводится в трехмерное за счет применения преобразования из координат экрана на плоскости xy определения текущей грани в нулевую координату z глубины.

9.8 Ввод Закладок

PROFESSIONAL

Закладки представляют собой вспомогательные объекты, которые можно добавить к графическому представлению для отметки значимых положений. Например, можно задать закладки в режиме привязки к: точкам пересечения, центрах, точкам смены квадранта.
Управление закладками - это специальная вкладка, подключенная на этапе конфигурации TpaCAD, но только в Профессиональном режиме.



Боковая строка соответствует командам:

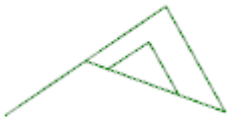
Присвоить закладку: эта команда активна на виде грани и запускает интерактивную процедуру задания нового положения закладки. Порядок управления процедурой подобен командам чертежа. В частности:
- не управляет **Привязкой между гранями**
- **Привязка к глубине** принудительно подключается без возможности отключения выбора (привязка к характеристике - всегда привязка к глубине).

В подтверждение положения в таблице задается новая строка, в которой указывается положение закладки в абсолютных координатах детали (столбец: X..Y..Z..) и текущей грани (столбец: x..y..z..).
Для выхода из процедуры нажать ESCAPE.
Можно задать до 50 закладок.

Убрать закладку, соответствующую выбранной в таблице строке

Убрать все закладки в таблице

Показать закладки: выбрать это поле для отображения закладок



Закладка отображается с символом, похожим на флажок, и номером, соответствующим номеру по порядку строки в таблице.

При наличии закладок интерактивные процедуры обучения координат могут выбирать привязку также на этих объектах из меню выбора **Объект привязки**:
- **Закладка**: позиционирование выполняется на закладку, ближайшую к курсору. Может также подключаться сочетанием клавиш **[CTRL+F]**.

Закладки считаются относящимися к программе в комплексе и адаптация положений к системе координат текущей грани выполняется автоматически. Таблица закладок остается без изменений при закрытии программы: обнуление должно запрашиваться выбором команды из командной строки.

Вкладка Закладки обеспечивает наличие также расстояний между двумя позициями:
- переместить выбор на строку таблицы (как показано на рисунке: строка 1);
- затем переместить мышь на другую строку

В зоне под таблицей приводятся расчетные расстояния между двумя позициями для каждой из трех осей абсолютной системы координат детали (dX, dY, dZ) и как абсолютное расстояние (d) . Как показано на рисунке: между двумя позициями в таблице (M1->2).

9.9 Изменение и Ввод

Выбрать точку вставки в программном списке

На виде грани текущая операции выделяется как в зоне текста ASCII, так и в зоне графического представления грани. Зона задания операции содержит настройки операции.

Прокрутить и выбрать текущую операцию на графическом виде
Зона графического представления грани становится интерактивной при щелчке курсором в этой зоне.

Реализованы следующие ситуации:
- **прямое наведение на операцию (щелчок в зоне)**: перемещает активированную операцию на операцию, больше всего приближающуюся к положению, показанному мышью. В частности, поиск выполняется по всему программному списку от первого до последнего блока, но только для операций, которые отображаются в данный момент. Из поиска исключаются логические операции (IF, ELSE, ENDIF) или комментарии и операции, не отображенные вследствие применения фильтров вида. Все выборы грани обнуляются. Если активно трехмерное представление детали, требуется пояснение: в этом случае поиск операции выполняется на графическом уровне с подключением процедуры, определяющей операцию, представленную внутри определенной области поиска, вокруг положения мыши. Если этот поиск имеет положительный исход, определение операции подтверждается, в противном случае *щелчок* игнорируется: это означает, что графическое определение операции требует выбора положения *рядом* с операцией. Для определения области поиска на видео см. параграф, соответствующий ***Индивидуализировать->Виды->Мышь*.**
- **прокрутка программы**. Нажимаются кнопки:
  - **<Стрелка вверх>**: перемещает активированную строку к предыдущему блоку в списке,
  - **<Стрелка вниз>**: перемещает активированную строку к следующему блоку в списке,
  - **<Предыдущая страница>**: перемещает активированную строку на страницу вверх (размер страницы определен в 10 строк),
  - **<Следующая страница>**: перемещает активированную строку на страницу вниз,
  - **<Начало>**: перемещает активированную строку на первый программный блок,
  - **<Конец>**: перемещает активированную строку на последний программный блок.

В перечисленных здесь случаях текущая операция может не отображаться.
Каждый раз при изменении активной операции все выборы грани обнуляются или остаются без изменений в зависимости от настройки **Добавить выборы** в **Индивидуализировать->Среда->Действие.**

Прокрутить и выбрать активизированную операцию в тексте ASCII
Можно прокручивать программу непосредственно в тексте ASCII.
Реализованы следующие ситуации:
- прямое наведение на операцию (щелчок в зоне): перемещает текущую строку на строку, указанную мышью. Все выборы грани обнуляются.
- прокрутка программы. Нажимаются те же кнопки, которые используются для прокрутки программы в графическом представлении. Каждый раз при изменении активированной операции все выборы грани сбрасываются

Прокрутить и выбрать активизированную операцию командами в меню
Можно выбрать текущую операцию также командами, предусмотренными в группе **Расположить в линию** вкладки **Редактирование**:

	**Первая операция:** перемещает текущую операцию на первую операцию в списке.
	**Последняя операция:** перемещает текущую операцию на последнюю операцию в списке.
	**Предыдущая операция:** перемещает текущую операцию на предыдущую операцию. (см. также пункт: **Соответствие на профилях)**
	**Следующая операция:** перемещает текущую операцию на следующую операцию. (см. также пункт: **Соответствие на профилях)**
	**Соответствие на профилях:** при подключении этого пункта две предыдущие команды рассматривают профили как единый элемент, заданный на подготовке или, в отсутствие подготовки, - на операции начала профиля. Если этот пункт не подключен: две предыдущие команды прокручивают программный список, применяя соответствие для каждой отдельной строки.

	**Операция начала профиля:** перемещает текущую операцию на операцию начала профиля: действует, если текущая операция относится к профилю. Эта команда может предусматриваться также в локальном меню Текст ASCII и Графический контекст.
	**Операция конца профиля:** перемещает текущую операцию на операцию конца профиля: действует, если текущая операция относится к профилю. Эта команда может предусматриваться также в локальном меню Текст ASCII и Графический контекст.
	**Перейти к строке ...:** перемещает текущую операцию на операцию с заданным номером по порядку. Эта команда может подключаться также из строки состояния при щелчке на зоне, в которой отображается номер по порядку текущей программной строки и общее число строк
	**Следующее соответствие:** при подключении этого пункта один щелчок курсора в графической зоне выполняет поиск в программном списке начиная с текущей операции до последнего блока. Если этот пункт не подключен, поиск осуществляется, в любом случае, по всему программному списку и всегда с начала списка. Ввиду того, что это графическое сопоставление, из поиска исключаются операции, не отображенные в графической зоне. В случае наложенных операций выбор этого типа позволяет прокручивать все операции, заданные в одном положении. Если поиск не дает результатов, выбор **Следующее соответствие** сбрасывается автоматически.

Вставить относительно текущей операции

На строке состояния имеется значок  **:**
- при активированном выборе операции вставляются после текущей операции;
- при неактивированном выборе операции вставляются до текущей операции.

Если вставка осуществляется посередине профиля, можно сместить точку вставки в положение до или после этого профиля.

Выбор

Выбор операций подключен только на виде грани и с программой с непустой гранью. Невозможно выполнять частичные выборы сложной операции (подпрограммы или макропрограммы) или множественного отрезка профиля.

Порядок выбора операций на графическом виде
Реализованы следующие сочетания клавиш с приоритетом в соответствии с последовательностью пунктов:

Для вычерчивания окна удерживать нажатой левую кнопку мыши и буксировать до получения нужного окна.
Выбираются операции, заключенные в окне. В частности:
- учитываются только операции, представленные на виде (применяет активированные виды и фильтры изображения)
- поэтому исключаются из поиска логические и комментируемые операции
- выбор зоны не меняет активную строку

[Shift+(левая кнопка мыши нажата)]: запускает выбор зоны.
[Shift+(левая кнопка мыши нажата)]+[ALT]: выбор расширяется для включения профилей, которые попали в это окно только частично
[Shift+(левая кнопка мыши нажата)]+[CTRL]: выбрать зону с операциями сохраняя предыдущие выборы

В режиме "грань-деталь" выбор зоны воздействует только на текущую грань.

[CTRL+(левая кнопка мыши нажата)]: выбирает или отменяет выбор операции, больше всего приближающейся к положению, показанному мышью.
- учитываются только операции, представленные на виде (применяет активированные виды и фильтры изображения)
- если оказывается нажатой также кнопка **[ALT]** и если выбор затрагивает элемент профиля: распространяется на весь профиль
- предыдущие выборы сохраняются
- не меняется активная строка

На графическом представлении выбранные операции отображаются различными цветами в зависимости от заданий, определенных на этапе индивидуализации TpaCAD.

Порядок выбора операций в тексте ASCII
Реализованы ситуации в списке с приоритетом в соответствии с последовательностью пунктов:

[Shift+(левая кнопка мыши нажата)]: выбирает от активной строки до программной строки, указанной мышью
- предыдущие выборы утрачиваются
- не меняется активная строка

[CTRL+(левая кнопка мыши нажата)]: выбирает или отменяет выбор программной строки, указанной мышью
- если оказывается нажатой также кнопка **[ALT]** и если выбор затрагивает элемент профиля: распространяется на весь профиль
- предыдущие выборы сохраняются
- не меняется активная строка

В тексте ASCII можно выбрать также операции, не представленные в этом виде.

Общие команды выбора

Эти команды находятся в группе **Изменить** вкладки **Редактирование.**

	**Выбрать все (CTRL+A):** выбирает все операции грани. В случае грани-детали можно выделить две ситуации: • при активированном трехмерном изображении или каркасном представлении выбор касается всего списка операций; • при активированном двухмерном изображении выбор касается только операций, примененных на грани на текущем виде
	**Отменить все выборы:** отменяет все выборы грани. В Виде грани эта команда осталась подключенной, но она практически устарела: выборы грани аннулируются автоматически щелчком на списке операций, в графическом представлении или в тексте ASCII. Однако интерес вызывает тот факт, что эта команда подключена и на Общем виде, аннулируя выборы в каждой грани.
	**Найти и выбрать:** открывает окно, в котором можно определить критерии поиска для применения к запрограммированным операциям, к которым применять выбор. Эта команда абсолютно аналогичная команде ***Заменить***, к которой следует обращаться для получения большей информации, только здесь вместо Заменить выполняется Выбрать. Эта команда подключена как на общем виде, так и на виде грани (в этом случае с программой с непустой гранью).
	**Сгруппировать:** перемещает все выборы грани в конец списка, обеспечивая их последовательность. По завершении группирования выборы остаются без изменений. Эта команда подключена на виде грани.

Эти команды могут предусматриваться в локальном меню **Текст ASCII** и **Графический контекст**:

	**Выбрать отсюда до начала профиля**: выбирает часть профиля между текущей операцией и началом профиля, к которому относится текущая операция.
	**Выбрать отсюда до конца профиля**: выбирает часть профиля между текущей операцией и концом профиля, к которому относится текущая операция.

Изменить текущую операцию

Прямое изменение операции состоит в изменении заданий параметров и/или свойств операции.
Прямое изменение всегда применяется к активированной операции и не может применяться к операциям в заблокированном состоянии (например, с заблокированным Уровнем, Конструкцией или Полем О) или с недействительным рабочим кодом.
Как правило, изменение применяется только если не сигнализированы ситуации ошибок, связанные с настройками этой операции. В этом случае при подаче сигнала ошибки необходимо устранить ошибочные ситуации или отменить изменения.
Можно также запросить возможность применения изменений операции также в случае, когда операция подала сигнал об ошибке. В этом случае вначале дается сигнал о возникшей ситуации ошибки, а затем можно запрашивать подтвердить выполненное изменение.

Подключение возможности подтверждения ситуации ошибки при изменении и вводе имеется в **Индивидуализировать** (раздел **Среда**, страница **Действие**, пункт: **Допустить подтверждение операции, составленной с ошибкой**).
После выполнения изменения текущей становится последующая операция в программном списке.

Общие команды Изменения в программе грани

Доступны команды, позволяющие изменять выбранную группу операций. Примером выбранной группы операций является применение специальных видов и/или фильтров изображения. Для общих команд изменения необходимо создать выбранную группу операций до выбора команды изменения.
Примеры выбранных групп операций:
- выбранные операции, отвечающие логическим ограничениям
- выбранные операции, заданные на определенном уровне

Многие команды изменения рассматривают в качестве выбранной группы привилегированных операций группу, образованную выбранными операциями. В этом случае если не активизирован никакой выбор, изменение применяется к текущей операции.

Общие команды изменения рассматриваются в последующих параграфах.

Изменение свойств

Изменение свойств возможно на виде грани с программой с непустой гранью.
Эти команды находятся в группе **Задать свойство** вкладки **Редактирование.**
В отображаемом окне можно подключить некоторые критерии для применения задания:
- **Соответствие на виде**: при подключении учитывает только операции, представленные на виде (применяет активированные виды и фильтры изображения). Подробное рассмотрение возможных ситуаций см. в описании команд **Найти и заменить**
- **Применить к выбранным операциям**: при подключении учитывает только выбранные операции. Выбор доступен при наличии выбранных операций.

Применение этого задания, в любом случае, зависит также от возможных фильтров блокировки изменения.

В случае грани-детали можно выделить две ситуации:
- при активированном трехмерном изображении или каркасном представлении изменение касается всего перечня операций
- при активированном двухмерном изображении грани изменение касается только операций, примененных на грани на текущем виде.

Поля свойств могут оказаться нередактируемыми (L, B, K, K1, K2 всегда, а также M и O), если в конфигурации не подключено их прямое редактирование в профиле в случаях:
- операций профиля (дуги и линии): если операция открывает профиль (открытый профиль), значение остается на 0, в противном случае они принимают значение операции начала профиля.
- подготовки или сложной операции с запросом зацепления точки: нередактируемые свойства в профиле распространяются из операции начала профиля.

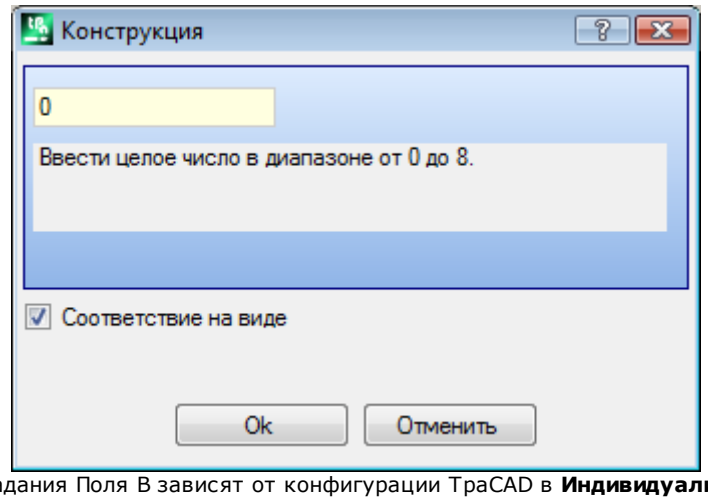
Можно запустить команды группы **Задать свойство** вкладки **Редактирование** также из зоны текста ASCII, щелкая на ячейке заголовка столбца, соответствующего этому свойству.

Свойство C или Комментарий
это необязательное свойство.

Задает поле Комментарий (C).
Выбор опции **Обнулить свойство** удаляет настройку свойства C для операций, для которых выполняется это изменение. В этом случае задание не проверяет активные фильтры вида, т. к. на экране не представляются программные строки с комментариями. Операции с заданным полем теперь не влияют на программу. Выбор опции **Задать свойство** обуславливает превращение изменяемых операций в комментарии, которые остаются в списке, но не влияют на программу. Если операция имеет активизированное свойство "C", она может меняться только после отключения комментария. Свойство C может подключаться также в случае операции, заданной недействительным рабочим кодом.
В отношении свойства Комментарий, напоминаем, что когда ссылаются на операцию до или после какой-либо операции, всегда подразумевается, что "операции комментария исключены".

Свойство L или Слой
это необязательное свойство.
Значение и порядок задания поля L зависят от конфигурации TpaCAD в **Индивидуализировать ->Среда->Редактирование операции** меню Применение.
В окне приводится случай выбора в списке значения, которое должно присваиваться операциям. В качестве альтернативы выбор может осуществляться в поле прямого редактирования (как для следующего случая: *Свойство B или Конструкция*).

Свойство B или Конструкция
это необязательное свойство.



Значение и порядок задания Поля B зависят от конфигурации TpaCAD в **Индивидуализировать ->Среда->Редактирование операции** меню Применение.
В окне приводится случай выбора в поле прямого редактирования значения, которое должно присваиваться операциям. В качестве альтернативы выбор может осуществляться в списке (как для предыдущего случая: *Свойство L или Слой*).
Операция, указанная как конструкция, компилируется, но не выполняется

Свойство O
это необязательное свойство.
Значение и порядок задания Поля O зависят от конфигурации TpaCAD в **Индивидуализировать ->Среда->Редактирование операции** меню Применение. В окне может приводиться выбор в списке или в поле прямого редактирования значения, которое должно присваиваться операциям.

Свойство M
Свойство K
Свойство K1
Свойство K2
Это необязательные свойства.
Задание значения свойства возможно только при прямом редактировании.

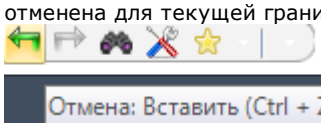
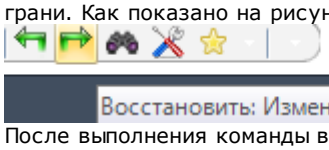
Свойство N или Имя
это необязательное свойство.
Свойству N присваивается строка длиной максимум 16 буквенно-цифровых символов, а первый символ должен быть алфавитным. При подтверждении окна с пустым полем задания запрашивается подтверждение для удаления Имени соответствующих операций. Это свойство используется, например, для применения запрограммированных инструментов или в особых функциях параметрического программирования.

Свойство Грань
доступно только в режиме "грань-деталь" задает грань применения операциям, задействованным в изменении.
Выбор грани осуществляется в списке, а перечисленные пункты соответствуют доступным реальным или фиктивным граням.

Общие команды изменения

К операциям могут применяться следующие команды: Копировать, Вставить, Вырезать, Удалить, Удалить все, Отмена. Это команды, подключенные только в Виде грани.
Эти команды воздействуют на выбранные операции, при их наличии, или же на текущую операцию. Они касаются только операций, проверяющих активированные фильтры изображения: варианты выбора, логические условия, уровни, специальные фильтры.
В случае текста макропрограммы при выборе "применить к профилю" он оценивается с включением логических операций, которые могут прерывать построение профиля.
Эти команды находятся в группе **Примечания** вкладки **Редактирование**.

	**Копировать (CTRL+C)**: копирование соответствующих операций в Локальный буфер приложения. Если копирование затрагивает только одну операцию, относящуюся к одному профилю, можно подтвердить копирование всего профиля. В случае грани-детали можно выделить две ситуации: • при активированном трехмерном изображении или каркасном представлении копирование касается всего списка операций; • при активированном двухмерном изображении копия касается только операций, примененных на грани на текущем виде.

	Эта команда может предусматриваться также в локальном меню Текст ASCII и Графический контекст.
	**Вставить (CTRL+V)**: вводит содержимое Буфера в точку вставки (до или после текущей операции). Если прямая вставка осуществляется посередине профиля, можно сместить точку вставки в положение до или после этого профиля. В случае грани-детали открывается диалоговое окно. • **[Да]**: оставить без изменения программирование поля **Грань для применения** вставленных операций • **[Нет]**: копировать в поле Грань для применения вставленных операций значение текущей грани. Эта команда может предусматриваться также в локальном меню Текст ASCII и Графический контекст.
	**Вырезать (CTRL+X)**: удаляет соответствующие операции, копируя их в локальные Примечания приложения. Если удаление затрагивает только одну операцию, относящуюся к одному профилю, можно подтвердить удаление всего профиля. Эта команда может применяться к операциям в заблокированном состоянии (заблокированный уровень, конструкция или поле O). В случае грани-детали можно выделить две ситуации: • при активированном трехмерном изображении или каркасном представлении удаление касается всего списка операций; • при активированном двухмерном изображении удаление касается только операций, примененных на грани на текущем виде. Эта команда может предусматриваться также в локальном меню Текст ASCII и Графический контекст.
	**Удалить**: удаляет соответствующие операции абсолютно аналогично предыдущей команде, но без копирования в Локальный буфер приложения. **(DEL):** при фокусе, активном на графике детали или на текстовом списке ASCII, при нажатии кнопки DEL выполняется команда **Удалить**.
	**Удалить все:** удаляет все операции без их копирования в Локальный буфер приложения и без оценки активизированных фильтров вида. В случае грани-детали можно выделить две ситуации: • при активированном трехмерном изображении или каркасном представлении удаление касается всего списка операций; • при активированном двухмерном изображении: удаление касается только операций, примененных на грани на текущем виде
	**Отмена (CTRL+Z)**: отменяет последнюю команду изменения программы грани. При установке курсора на значке команды подсказка указывает, какая команда первой в списке может быть отменена для текущей грани. Как показано на рисунке: команда **Вставить**  После подачи команды в зоне "Команды" приводится индикация отмененной команды. Перечень команд, которые можно отменить, сбрасывается при закрытии активированной программы. Эта команда имеется также на панели инструментов меню Применение. На Виде грани можно отменить также команды изменения, которые были выполнены на общем виде: например, полную замену параметров (см. ниже: команда **Заменить**).
	**Восстановить (CTRL+Y)**: восстанавливает последнюю отмену изменения программы грани, запрошенную командой **Отмена (CTRL+Z)**. При установке курсора на значке команды подсказка указывает, какая команда первой в списке может быть восстановлена для текущей грани. Как показано на рисунке: команда **Изменить операцию**  После выполнения команды в зоне "Команды" приводится индикация восстановленной команды. Перечень команд, которые можно сбросить, соответствует отменам, которые были запрошены без последующего сброса. Этот перечень, в любом случае, обнуляется при закрытии активной программы. Эта команда имеется также на панели инструментов меню **Применение**.

Найти

Команда **Найти**  подключена только на виде грани и с программой с непустой гранью. окно задания данных поиска можно вызвать из группы **Изменить** вкладки **Редактирование.**

Это окно при открытии представляет настройки, как они были заданы при предыдущем вызове этой команды.
Кроме того, доступны управляемые табуляторы в группе команд: Заменить, Заменить переменные.
Первый выбор этой кнопки **[Найти следующий]** позволяет запустить поиск первой операции грани, удовлетворяющей заданным критериям:

- **Операция**: - это код ASCII искомой операции (в примере - "HOLE"). Если это поле не задано, поиск не применяется к коду ASCII. Кнопка  , расположенная рядом, позволяет задавать поле коду текущей операции.
- **Параметры**: задания искомых параметров (в примере - "TMC=1 TR=1"). Если это поле не задано, поиск не применяется к параметрам. Это поле должно задавать пункты, разделенные пробелом. Каждый пункт должен называться именем ASCII параметра, за которым следует значение, как оно было запрограммировано. Для имен, содержащих десятичные цифры, или для параметрического задания обязательна форма "имя=..".
  Пример действительного задания: "TMC=1 TD=r5 P1=12" где:
  "TMC=1" присваивает значение 1 (числовое) параметру "TMC", "TD=r5" присваивает значение r5 (параметрическое) полю "TD", "P1=12" присваивает значение 12 (числовое) полю "P1". Для указания поиска незаданного параметра (т.е. пустого поля) указать только имя (указывая за ним "=" если оно содержит десятичные цифры). Примеры: "TMC=" и "TMC" - эквивалентные формы, "P1=" - это форма, обязательная для параметра имени "P1". Изменение этого поля может вызывать автоматические изменения, обуславливаемые автоматическими проверками. В частности: части, распознаваемые как имена параметров, задаются прописными буквами.
- **Свойства**: задания исходных свойств (в примере - "L=1"). Если это поле не задано, поиск не применяется к свойствам. Это поле должно задавать пункты, разделенные пробелом. Каждый пункт должен называться именем свойства (L для слоя, B для конструкции, а затем - O, M, K, K1, K2), за которым следует соответствующее значение (для полей K1 и K2 или для параметрического задания обязательна форма "K1=..").
  Примеры действительного задания:
  - "L4 M5000 K12 K1=5" где: "K12" присваивает значение 12 полю K, "K1=5" присваивает значение 5 полю K1.
  - "L4 M=r5 K=12 K1=5" где: "M=r5" присваивает параметрическое задание "r5" полю M, "K=12" присваивает значение 12 полю K с формой, эквивалентной "K12".

  В отличие от параметров свойство считается всегда заданным с заданием по умолчанию ="0". Изменение этого поля может вызывать автоматические изменения, обуславливаемые автоматическими проверками. В частности: части, распознаваемые как имена свойств, задаются прописными буквами.

Следующие опции могут выбираться в зоне Опции поиска:
- **Искать во всех направлениях:** при подключении запускает поиск во всем списке, в противном случае - только после текущей операции;
- **Соответствие на виде**: при подключении учитывает только операции, представленные на виде (применяет активированные виды и фильтры изображения). Рассмотрим подробно виды и примененные фильтры:
  - поиск исключает следующие операции: логические, с активным полем C или с недействительным рабочим кодом (т. е. операция, не имеющая соответствия в базе данных операций)
  - с активным видом выборов: принимает во внимание только выбранные операции
  - с активным видом логических условий: принимает во внимание только операции, удовлетворяющие логическим условиям, включая исключения
  - с активным видом фильтров уровней: принимает во внимание только операции, заданные с отображенным уровнем
  - с активным видом специальных фильтров: принимает во внимание только операции, проверенные специальными фильтрами вида (поля B, O, K, K1; технология).

Поле **Соответствие на виде** может меняться, если активно поле отображения программы (в строке состояния). В противном случае этот пункт выбрать нельзя.

- **Применить к выбранным операциям**: при подключении учитывает только выбранные операции. Выбор доступен при наличии выбранных операций. Подключение этой опции рассматривается только если не подключен пункт **Соответствие на виде**, который уже включает ее.

В режиме грань-деталь при выбранном поле **Соответствие на виде** можно выделить следующие ситуации:
- при активированном трехмерном изображении поиск касается всего перечня операций

- при активированном каркасном представлении поиск касается только операций, применимых к реальным граням детали
- при активированном двухмерном изображении поиск касается только операций, применимых на грани на текущем виде.

При невыбранном поле **Соответствие на виде** поиск все также касается всего перечня операций.

Кнопка **[Найти следующий]** позволяет запускать или продолжать поиск:
- Выбор этой кнопки отсутствует, если не задано ни одно из полей поиска (**Операция, Параметры, Свойства**)
- сообщение извещает, что поиск не дал положительного результата
- в противном случае найденная операция становится активизированной операцией.

Поиск задается здесь без применения условий, препятствующих изменению операций (например, заблокированное поле L).

Кнопка **[Найти все результаты]** позволяет найти все соответствия. Результат поиска представляется в окне Команды.

Заменить

Команда **Заменить**  подключена как на общем виде, так и на виде грани (в этом случае с программой с непустой гранью). окно задания данных поиска можно вызвать из группы **Изменить** вкладки **Редактирование.**
Это окно при открытии представляет настройки, как они были заданы при предыдущем вызове этой команды.
Кроме того, доступны управляемые табуляторы в группе команд: Найти, Заменить переменные.

В области Найти задаются данные, определяющие критерии поиска (см. команду **Найти**):
- **Операция**: код ASCII искомой операции.
- **Имя**: название искомых операций
- **Параметры**: задания искомых параметров.
- **Свойства**: задания искомых свойств.

В области Заменить на задаются новые задаваемые данные:
- **Операция**: код ASCII операции (должен соответствовать действительной операции, применимой в программном списке).
- **Параметры**: задания параметров. Для задания этого поля действительно указанное для соответствующего поля в области Найти. В частности, для указания удаления задания параметра (оставляя поле пустым) привести только имя (указывая за ним "=", если оно содержит десятичные цифры). Примеры: "TMC" или "TMC=" - эквивалентные формы, "P1=" - это форма, обязательная для параметра имени "P1".
- **Свойства**: задание свойств. Для задания этого поля действительно указанное для соответствующего поля в области Найти.

Ячейка выбора рядом с полем подключает выполняемое задание кода операции, параметров или свойств.

Если команда **Заменить** активирована в общем виде:
- область Опции поиска недоступна;
- доступна только кнопка **[Заменить все]:** заменяет все операции всей программы, удовлетворяющие критерии поиска, новыми заданными данными. Выбор этой кнопки не будет иметь никакого действия, если не задано выбранное поле замены (**Операция, Параметры, Свойства**).

Если команда **Заменить** активирована в виде грани:
- область Опции поиска доступна (см. команду Найти);
- доступны кнопки**:**

  [Найти следующий] позволяет запускать или продолжать поиск без выполнения замен.
  Сообщение извещает, что поиск не дал положительного результата.
  При обнаружении соответствия: найденная операция становится текущей.
  В отличие от сказанного для вкладки *Заменить* здесь поиск выполняется с возможностью применения условий, не допускающих изменения операций (например, заблокированное поле L): найденных пригодных для замены операций может быть меньше операций, выявляемых обычным поиском.
  Если все поля зоны Найти пусты, поиск выполняется с применением только Опций поиска и общих условий изменения операций.

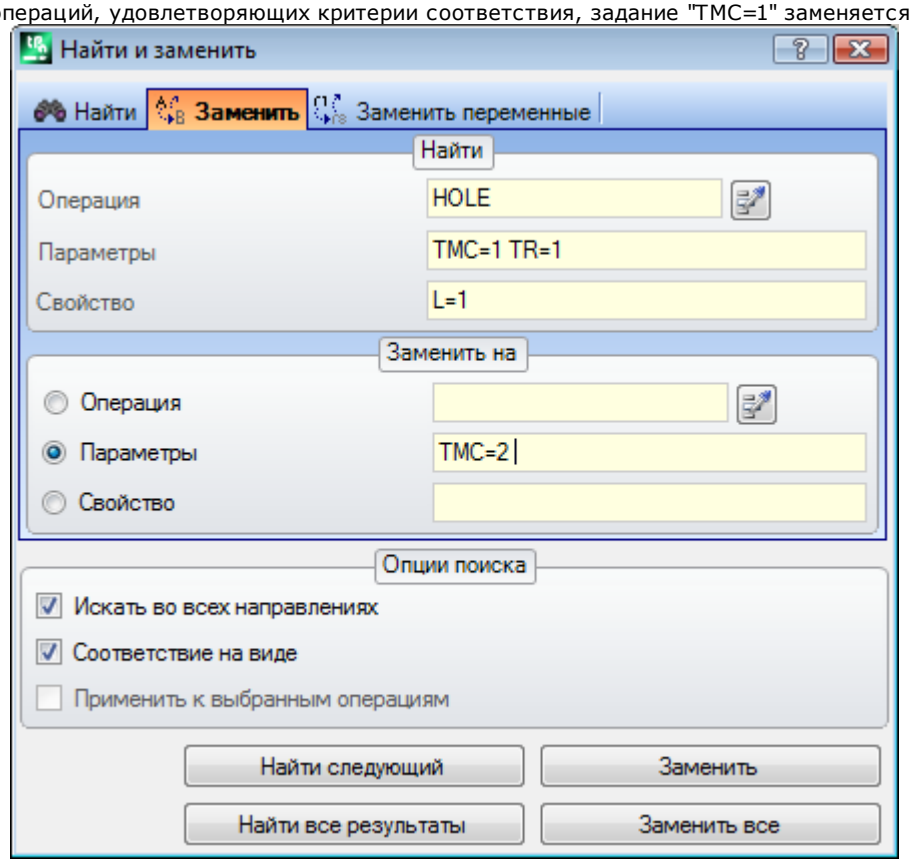
  [Заменить] применяет замены к найденной операции, соответствующей заданным параметрам;
  [Заменить все] заменяет все операции грани, удовлетворяющие критерии поиска, новыми заданными данными.
  Выбор кнопок замены не будет иметь никакого действия, если не задано выбранное поле замены (**Операция, Параметры, Свойства**)

[Найти все результаты] находит все соответствия и приводит результат поиска в окне Команды.
В случае грани-детали при выбранном поле **Соответствие на виде** различаются следующие ситуации:

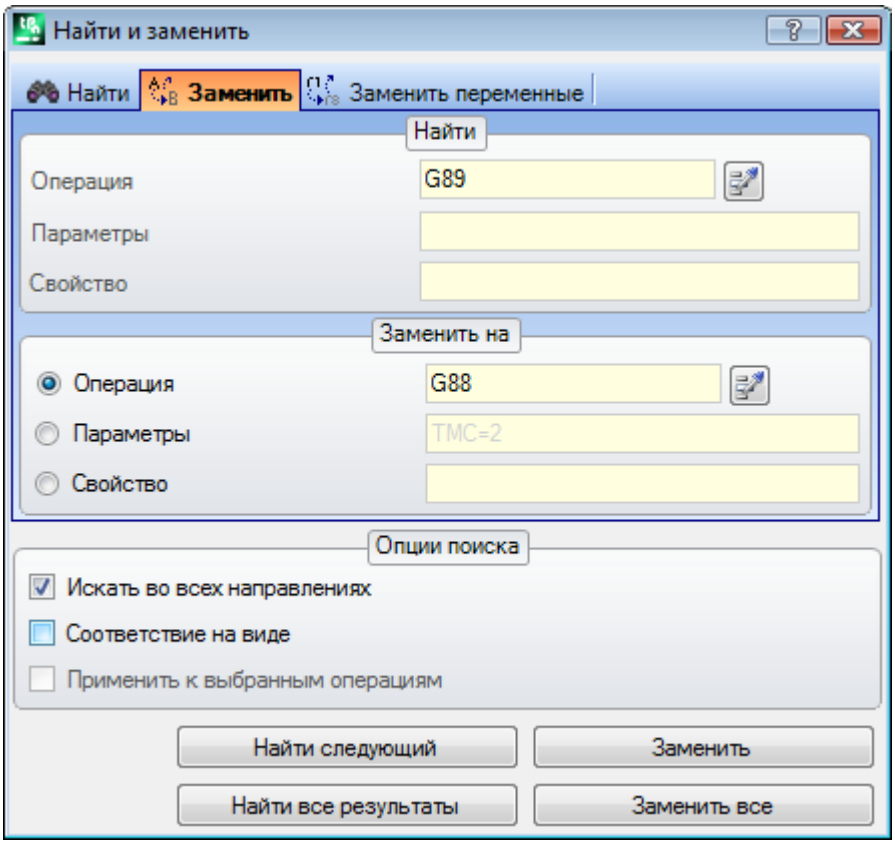
- при активированном трехмерном изображении поиск касается всего перечня операций
- при активированном каркасном представлении поиск касается только операций, применимых к реальным граням детали
- при активированном двухмерном изображении поиск касается только операций, применимых на грани на текущем виде.

Как уже говорилось, все замены, даже выполненные из общего вида, могут затем отменяться выбором на виде грани.

На представленном ниже рисунке требуется выполнить селективную замену параметров операции. Замена касается только операций "HOLE" с настройками, указанными для параметров ("TMC=1 TR=1") и свойств ("L=1"). Для операций, удовлетворяющих критерии соответствия, задание "TMC=1" заменяется на "TMC=2".



На втором рисунке требуется заменить рабочий код. Замена касается только операций "G89", без других настроек, и они заменяются операциями с именем "G88".



Предположим, что мы открываем программу, использующую коды операции, не определенные или не действительные в текущей конфигурации приложения (в нашем примере "G89" не соответствует ни одному рабочему коду операции).
Одним из способов обеспечения действительности операций является замена операций "G89" правильной операцией (в примере - "G88"). Однако при поиске должно быть дезактивировано **Соответствие на виде** , т. к. невозможно получить графическое представление операций "G89".
Кнопка **[Найти следующий]** позволяет продолжать поиск без выполнения замен, кнопка **[Заменить]** выполняет замены для найденной (текущей) операции, кнопка **[Заменить все]** выполняет замены во всех операциях грани, отвечающих заданным критериям.

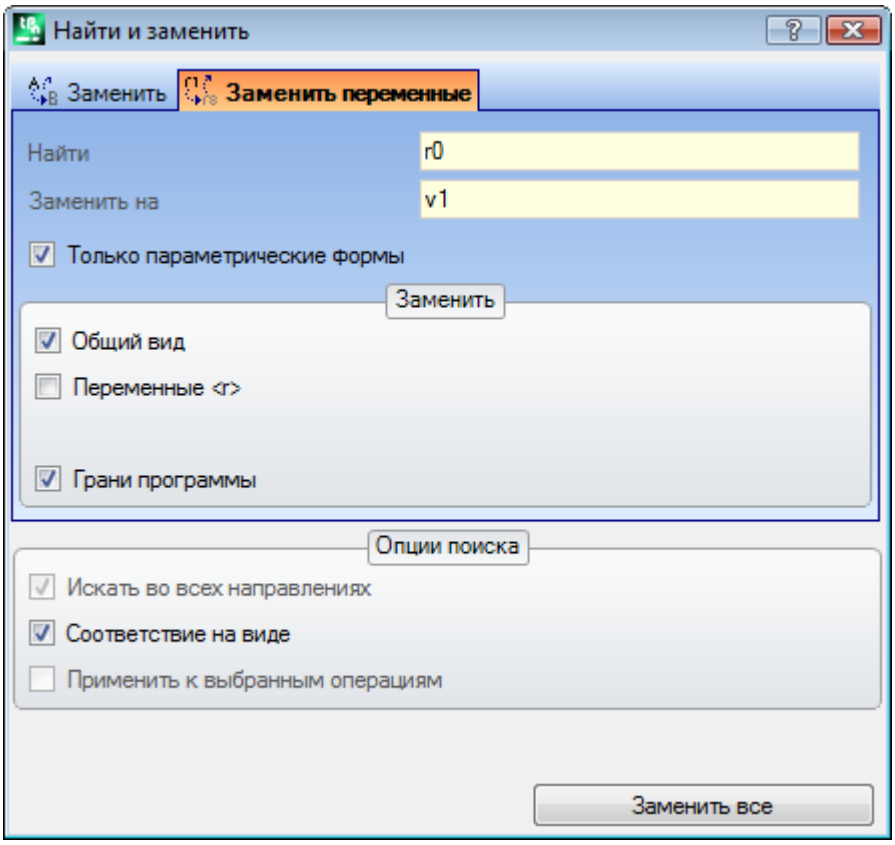
Заменить переменную

Это окно при открытии представляет настройки, как они были заданы при предыдущем вызове этой команды.
Кроме того, доступны управляемые табуляторы в группе команд: Найти, Заменить.
Находит случай заданной параметрической формы и выполняет ее замену. Команда **Заменить переменную** подключена как на общем виде, так и на виде грани (с программой с непустой гранью) и различна в этих двух случаях.
окно задания данных поиска можно вызвать из группы **Изменить** вкладки **Редактирование.**
Эта команда позволяет задавать поиск с возможной заменой следующих элементов:

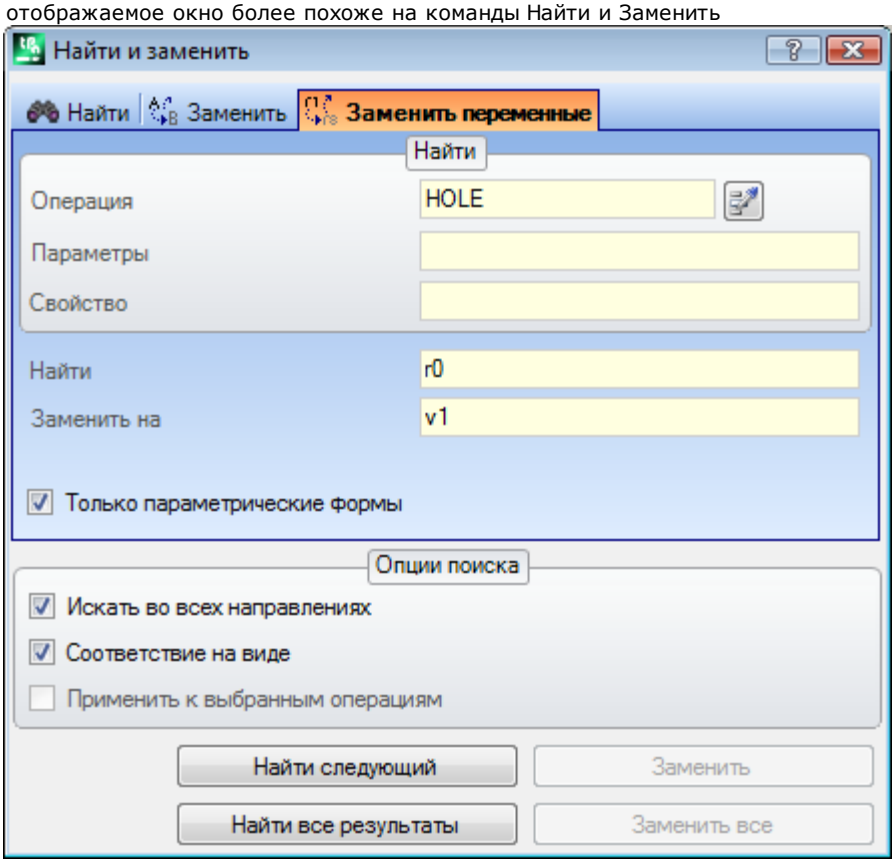
- переменные o, v, r, j (только в тексте макропрограммы $). например, позволяет заменять использование переменной типа <v> другой типа <r> (заменяет "v5" на "r15")
- переменные и/или переменные аргументы. Например, позволяет заменять "r5" на "lf", "lf" на "r\dim", "r\dim" на "100.6"
- общие подстроки. Например, позволяет заменять "geo[lface;" на "geo[isface;", "r5" на "abs[r5]", "r5" на "-100.6".

В общем виде, помимо программ грани, можно запросить замену также в общих заданиях программы (переменные, переменные геометрии).
На виде грани можно задавать дополнительные критерии поиска, в порядке, абсолютно аналогичном командам Найти/Заменить.

В общем виде отображается следующее окно:



На виде грани отображаемое окно более похоже на команды Найти и Заменить



На общем виде зона Заменить позволяет выбирать области поиска для выполнения замены:

- Общий вид или Текущий раздел, как показано на рисунке: Переменные <r>,

- Программы грани.

На виде грани в области Найти задаются данные, определяющие критерии поиска (см. команду Найти: Операция, Параметры, Свойства).

Настройки для задания переменных находятся в двух полях:
- **Найти**: параметрическая форма заменяемой переменной
- **Заменить на**: заменяемая параметрическая форма.

Оба поля должны быть заданы.

Опция **Только параметрические формы** выбирает заявленный формат для полей **Найти** и **Заменить на:**
- выбрана: заявляет, что два поля задают параметрическую форму переменной или переменного аргумента. Действительны, например, следующие формы: "r5", "r\dim", "o7", "lf";
- не выбрана: заявляет, что эти два поля задают общие подстроки.

В первом случае поиск и замены выполняются при контроле синтаксиса заданных полей и заменяются строки, если им не предшествуют цифры или буквы. Например, заменим "r1" на "r\abc":
- строка "lf+r1/4" меняется на "lf+r\abc/4"
- строки "lf+pr1/4", "lf+r12/4" не меняются.

Сигнал ошибки указывает, что поля заданы неправильно.

Во втором случае поиск и замены выполняются без контроля синтаксиса.

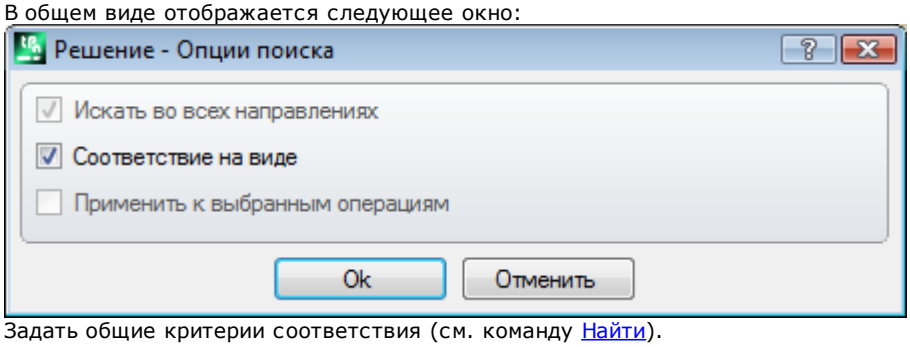
Кнопка **[Найти следующий]** позволяет запускать или продолжать поиск без выполнения замен, кнопка **[Заменить]** выполняет замены для найденной (и текущей) операции. Эти две кнопки подключены только на виде грани.
Кнопка **[Заменить все]** выполняет замены во всех случаях, отвечающих заданным критериям:
- на общем виде: замена касается разделов, выбранных в зоне Заменить: переменные <o,v>, переменные <r>, переменные геометрии и программы граней.
- на виде грани: заменяет параметрическую форму, указанную в операциях грани.

Кнопка **[Найти все результаты]** подключена на виде грани: она находит все соответствия и приводит результат поиска в окне Команды.
Как уже говорилось, все замены, даже выполненные из общего вида, могут затем отменяться выбором на виде грани.

Решить

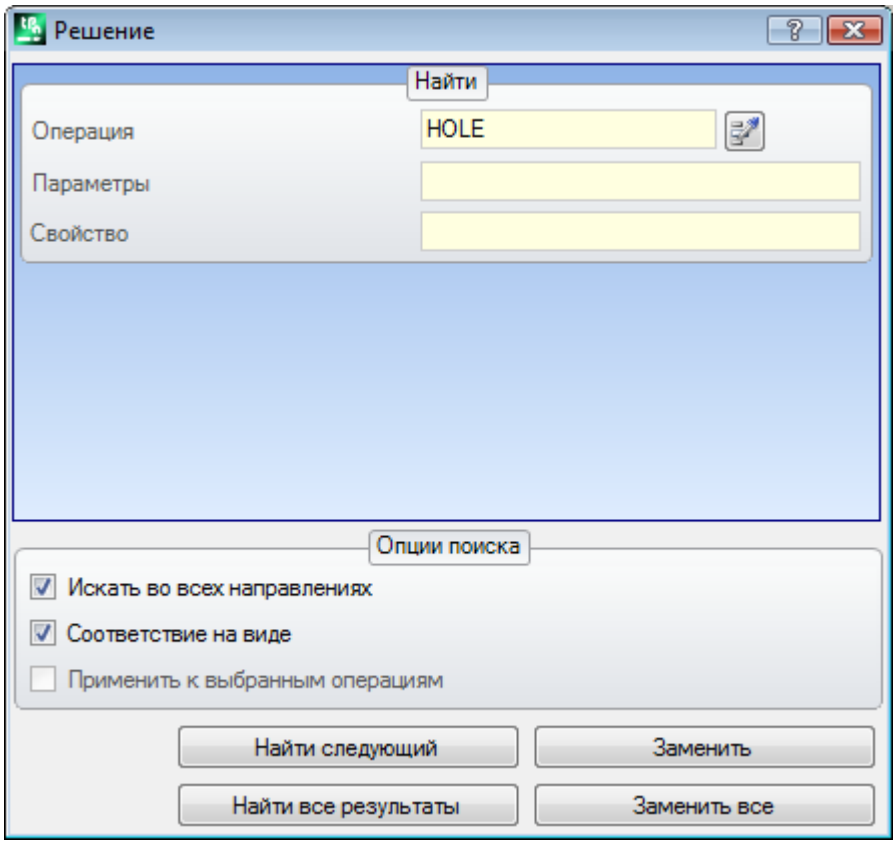
Команда Решить  подключена как на общем виде, так и на виде грани (в этом случае с программой с непустой гранью).
окно задания данных поиска можно вызвать из группы **Изменить** вкладки **Редактирование.**
Эта команда находит любую параметрическую форму числового типа, использованную при задании операций, и заменяет ее соответствующим значением, которое оценивается по текущему состоянию параметризаций (размеры, переменные).
Остаются без изменений параметрические задания нечислового типа, как, например, надпись или имя подпрограммы, если они заданы в параметрическом формате.

В общем виде отображается следующее окно:



Задать общие критерии соответствия (см. команду Найти).
При подтверждении этого окна эта команда применяется ко всем запрограммированным граням.

На виде грани отображаемое окно более похоже на команду Найти



В области Найти задаются данные, определяющие критерии поиска (см. команду Найти: Операция, Параметры, Свойства).
В области Опции поиска задаются общие критерии соответствия (см. команду Найти).

Замена параметрических форм может контролироваться кнопками **[Найти следующий]** и **[Заменить]**.
Кнопка **[Заменить все]** выполняет замены во всех случаях, отвечающих заданным критериям.

10 Инструменты

10.1 Введение

Инструментами считаются все команды, направленные на изменение, в основном, геометрического характера операций. Однако к группе инструментов относятся также некоторые команды, которые меняют технологию операций.
Открываемые окна предлагают настройки, какими они были заданы при предыдущем вызове инструмента. Инструменты применяются к операциям, проверяющим активированные фильтры изображения: варианты выбора, логические условия, уровни, специальные фильтры. Если инструмент применяется непосредственно к первоначальным операциям (выбранным или текущим), изменение не может быть применено к операциям в заблокированном состоянии (т. е. с: заблокированным уровнем, конструкцией или полем O).
Только для общих инструментов в окне настройки автоматически предлагается опция **Применить к копии операций**, в зависимости от того, как она задана в строке состояния.

Если инструменты генерируют новые профили, они открываются при помощи:
- копии исходной подготовки, при наличии
- в противном случае - копии эталонной подготовки (как задано **Индивидуализировать->Технология->Коды по умолчанию** меню Применение).

В режиме грани-детали большая часть инструментов может отключаться, если активно каркасное представление и текущая операция задана на нереальной грани.

Применение инструмента может существенно изменить структуру изменяемых операций при их прямом задании (коды операции, задание параметров в параметрической форме) и/или геометрическом разрешении. Где возможно, сохраняется исходная структура операций, с особым вниманием заданиям параметрического формата, однако это гарантируется не всегда.

10.2 Общие

Центрирование и выравнивание

Группа инструментов перемещает выбранные или текущие операции с центрированием или выравниванием по грани. Эти команды находятся в группе **Общие** вкладки **Инструменты** и отличаются от остальных команд группы тем, что они применяются напрямую, без дополнительного задания, за исключением случая, когда выбрана опция *Применить к копии операций* в строке состояния TpaCAD: в этом случае требуется прямое подтверждение.



Центрировать на грани по оси X: перемещает операцию/операции с центрированием по оси X грани. Позиционирование по оси Y грани не меняется



Центрировать на грани по оси Y: перемещает операцию/операции с центрированием по оси Y грани. Позиционирование по оси X грани не меняется



Центрировать на грани по оси X+Y: перемещает операцию/операции с центрированием на грани (сочетает две предыдущие команды)



Выровнять с гранью с X=0: перемещает операцию/операции, выравнивая минимальный габарит с положением X=0 грани. Позиционирование по оси Y грани не меняется



Выровнять с гранью с X=lf: перемещает операцию/операции, выравнивая максимальный габарит с положением X=lf грани. Позиционирование по оси Y грани не меняется



Выровнять с гранью с Y=0: перемещает операцию/операции, выравнивая минимальный габарит с положением Y=0 грани. Позиционирование по оси X грани не меняется



Выровнять с гранью с Y=hf: перемещает операцию/операции, выравнивая максимальный габарит с положением Y=hf грани. Позиционирование по оси X грани не меняется

Перенос

Переносит выбранные или текущую операцию в заданное положение. Перенос операции, относящейся к профилю, всегда обуславливает перенос всего профиля Команда **Перенести**  находится в группе **Общие** вкладки **Инструменты.**

Это окно предлагает все подключаемые режимы и опции. Прежде всего - выбор системы позиционирования на плоскости xy грани для программирования точки размещения:

- **прямоугольная**: присваивает координаты x и y точки размещения в абсолютном или относительном режиме;
- **полярная**: присваивает координаты x и y центра полярной системы (в абсолютном или относительном режиме), модуль и угол.

При абсолютном позиционировании в указанное положение позиционируется:

- текущая операция

Если активизирована опция **Позиционировать центр габарита**, в указанное положение переносится центр габаритного прямоугольника перенесенных операций.
В качестве альтернативы можно перемещать ребро габаритного прямоугольника, выбирая одну из четырех кнопок:

 переносит ребро на минимальные координаты по X и Y

 переносит ребро на максимальную координату по X и минимальную координату по Y

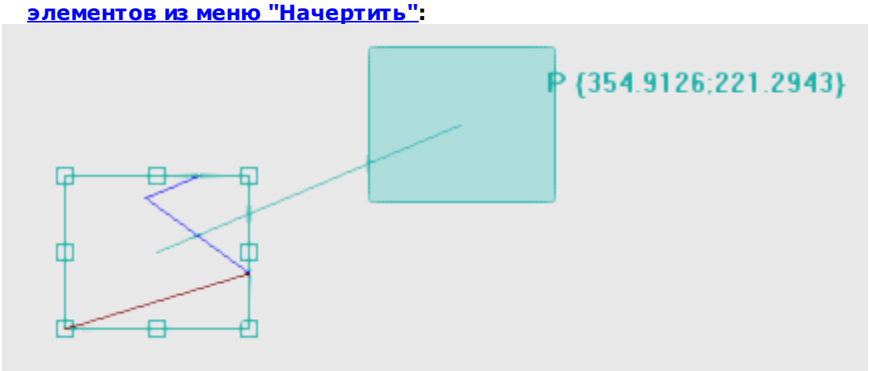
 переносит ребро на максимальные координаты по X и Y

 переносит ребро на минимальную координату по X и максимальную координату по Y

Координаты позиционирования могут вводиться:

- в поля редактирования позиционирование может выражаться в абсолютных и относительных числовых или параметрических значениях;
- мышью в графической зоне при щелчке на значке  (только если активно отображение программы). В этом случае координаты позиционирования X и Y автоматически задаются как абсолютные. При выборе позиционирования в полярной системе координат: интерактивный сбор может касаться положения центра, а также значений модуля и угла. Сообщения, приведенные в зоне Команд, направляют интерактивный режим.

Для интерактивного режима задания положения действительно сказанное для **Ввод геометрических элементов из меню "Начертить":**



- вычерчивается габаритный прямоугольник, соответствующий исходному габариту операций, к которому применяется перенос вместе с указанием исходной точки для переноса (центр габарита (предпочтительнее), текущая операция)
- перемещению мыши соответствует графическое обновление позиционирования габаритного прямоугольника и исходной точки для переноса, в соответствии с текущим положением мыши.

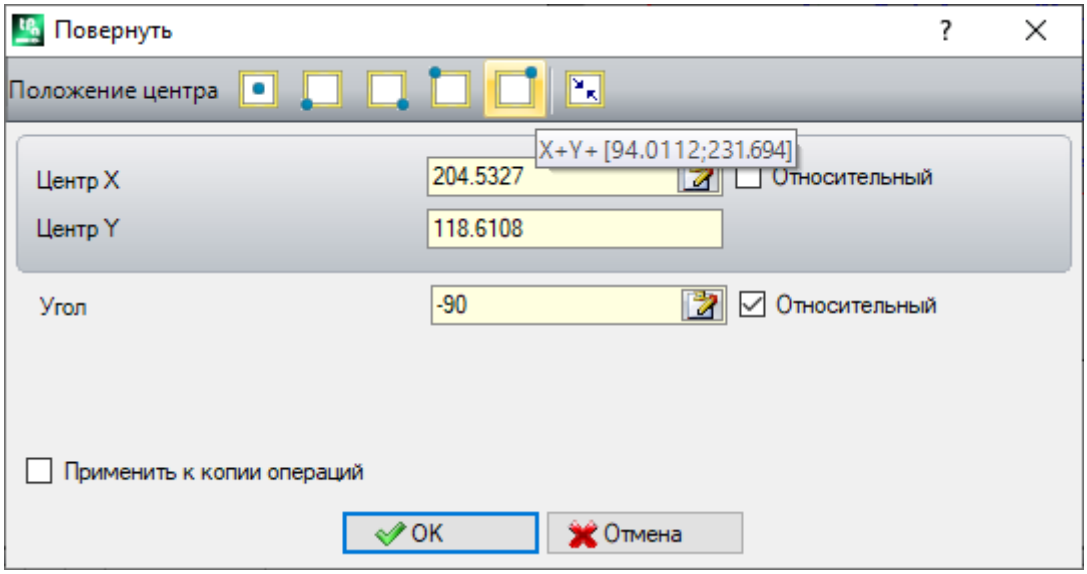
Автоматический указатель
Для точек вершин и медиан на сторонах габаритного прямоугольника активируется автоматический указатель, приводящий положение мыши в рамки, ограничивающие точки. В любом случае, можно отключить этот автоматический указатель, отменяя выбор на соответствующем пункте в локальном меню. Отключение автоматического указателя может, например, позволить выполнить точное позиционирование, выполняемое, например, выбором клавиш направления.

Выход из интерактивной процедуры возвращает к окну прямого задания, в котором можно выполнять интегрирование требуемыми заданиями и выборами.

Выбор опции **Применить к копии операций** применяет этот инструмент к копии операций и не меняет исходные линии.
Позиционирование в полярных координатах или в абсолютных координатах определяет утрату всякой формы параметрического программирования.

Вращение

Поворачивает выбранные или текущую операции. Команда **Повернуть** находится в группе **Общие** вкладки **Инструменты.**



Данные поворота можно ввести:
- в поля редактирования позиционирование может выражаться в абсолютных и относительных числовых или параметрических значениях;
- мышью в графической зоне при щелчке на значке . В этом случае координаты позиционирования X и Y центра поворота автоматически задаются абсолютными. Интерактивное считывание может касаться также угла. Сообщения, приведенные в зоне Команд, направляют интерактивный режим.
- при выборе кнопки из командной строки в окне:
  - первые 5 кнопок задают положение центра на одной из значимых точек габаритного прямоугольника операций, задействованных во вращении (центр или ребро). Положение точек приводится в сообщении (подсказке), отображаемом для каждой кнопки
  - кнопка справа задает центр и угол вращения на такие значения, чтобы максимально уменьшить габаритный прямоугольник.

При относительном позиционировании центра вращения сам центр располагается в относительном режиме относительно:

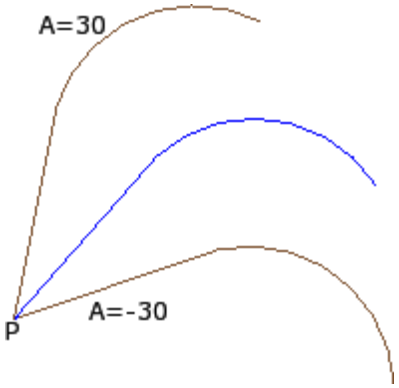
- текущей операции

При подключении из Конфигурации TpaCAD применение этого инструмента к ориентируемой подготовке может предусматривать трансформацию для осей ориентации (только если текущая грань плоская, т. е. не закруглена или задана как поверхность).

Вращение операции, относящейся к профилю, влечет за собой:
- вращение всего профиля, если вращение применяется к операциям, скопированным в Локальный буфер, или к выбранным операциям, или если не выбрана опция **Применить от текущего положения до конца профиля**;
- в противном случае: вращение части профиля между текущей операцией и концом профиля: теперь центр вращения совпадает с начальной точкой текущей операции. Если активизирован выбор опции **Применить инструмент к копии операций**, в любом случае, вставляется копия всего профиля.

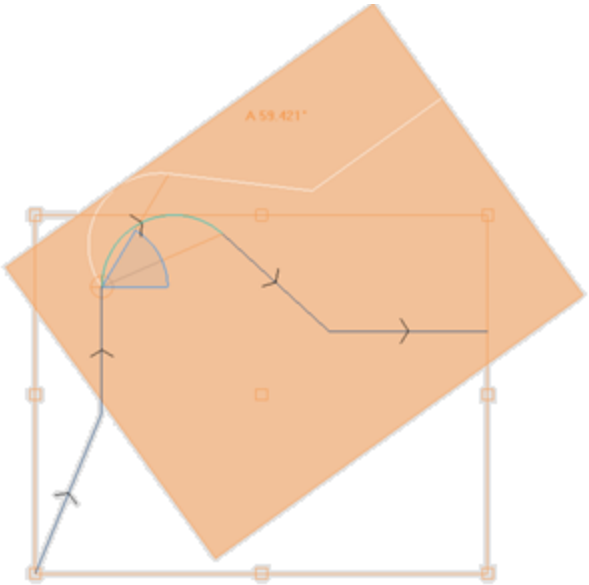
Применение инструмента определяет утрату всякой формы параметрического программирования, которая могла изначально присутствовать на плоскости грани.

Рассмотрим особый пример вращения:



текущая операция в **P**;
центр в относительном режиме на координатах (0;0). Так центр позиционируется в **P**;
относительный угол вращения со значением:
- 30° (для вращения вверх)
- -30° (для вращения вниз).

На рисунке показан пример поворота, примененного к части профиля, с активным интерактивным захватом:



Инструмент вращения применяется не ко всем операциям. Например, исключаются сложные операции, при которых выполняется одно или оба следующих условия:
- они вызывают подпрограмму или макропрограмму, к которым, в соответствии с определенным в базе данных операций, нельзя применить вращение.
- они сами сконфигурированы в базе данных операций как операции, к которым нельзя применить вращение.

Типовыми случаями являются операции прорези, выполняемые неориентируемым инструментом.

Правка (меню "Графическое представление")

Эта команда доступна из контекстуального меню, вызываемого из зоны графического представления нажатием правой кнопки мыши. Эта команда недоступна для функции *Essential*.

Эта команда позволяет применять простые преобразования *Переноса* и *Поворота* полностью интерактивно. Преобразования применяются к выбранным операциям или к текущей операции. Если операция относится к профилю, преобразования всегда применяются внутри профиля.

Как уже описывалось для инструмента **Перенести**, при подключении команды вычерчивается габаритный прямоугольник, соответствующий исходному габариту операций, вместе с центром и точками вершин и средними точками на сторонах прямоугольника. На этих точках может активизироваться автоматический указатель, приводящий положение мыши в рамки, ограничивающие точки.
Исходной точкой для переноса и/или поворота теперь становится центр габарита.
Перемещению мыши соответствует графическое обновление позиционирования габаритного прямоугольника и исходной точки для переноса и/или поворота, в соответствии с текущим положением мыши.

Переключение между двумя возможными трансформациями выполняется при помощи контекстуального меню при выборе пункта:

'R'= Повернуть переключает на инструмент поворота
'M'= Перенести переключает на инструмент переноса

Для подтверждения трансформации щелкнуть левой кнопкой мыши.
[Enter] выполняет закрытие команды, подтверждая выполненные считывания, **[Escape]** выполняет закрытие с отменой.

Симметрично отобразить

Инструменты симметрии представляют выбранные или текущую операции в зеркальном отображении относительно указанной оси.
Команды **Симметрично отобразить** находятся в группе **Общие** вкладки **Инструменты** и все открывают одно окно, в котором можно изменить требуемый тип симметрии.

Можно выбрать 4 типа симметрии:

	Симметрия по вертикальной оси
	Симметрия по горизонтальной оси
	Симметрия горизонтальная+вертикальная (вокруг точки)
	Общая симметрия

В случае **Симметрии по вертикальной оси** или **Симметрии по горизонтальной оси** в этом окне предлагается только значение координаты оси симметрии.
- вертикальная ось параллельна оси Y грани;
- горизонтальная ось параллельна оси X грани.

В случае **Симметрии Горизонтальная+Вертикальная** в этом окне предлагаются координаты X и Y точки симметрии.
В случае **Общей симметрия** в этом окне предлагаются координаты X и Y двух точек для определения общей оси симметрии. Кроме того, это определяет утрату всякой формы параметрического программирования, которая могла изначально присутствовать на плоскости грани.

Если этот инструмент применяется к профилю, инвертируются также настройки коррекции на инструмент (правый или левый) и выбор начального/конечного отрезков профиля в случае задания правой или левой дуги. При подключении из конфигурации TpaCAD применение этого инструмента к профилю может предусматривать зеркальную технологию.
При подключении из Конфигурации TpaCAD применение этого инструмента к ориентируемой подготовке может предусматривать трансформацию для осей ориентации (только если текущая грань плоская, т. е. не закруглена или задана как поверхность).

Симметрия операции, относящейся к профилю, влечет за собой:
- симметрия всего профиля, если симметрия применяется к операциям в Локальном буфере или к выбранным операциям, или если выбирается режим **Симметрии по общей оси**; или если не выбрана опция **Применить от текущего положения до конца профиля**;

- в противном случае: симметрия части профиля между текущей операцией и концом профиля: ось симметрии позиционируется у начальной точки текущей операции. Если активизирован также выбор опции **Применить к копии операций**: в любом случае, вставляется копия всего профиля.

Инструмент симметрии применяется не ко всем операциям. Например, исключаются сложные операции, при которых выполняется одно или оба следующих условия:
- они вызывают подпрограмму или макропрограмму, к которым, в соответствии с определенным в базе данных операций, нельзя применить выбранную симметрию
- они сами сконфигурированы в базе данных операций как операции, к которым нельзя применить выбранную симметрию.

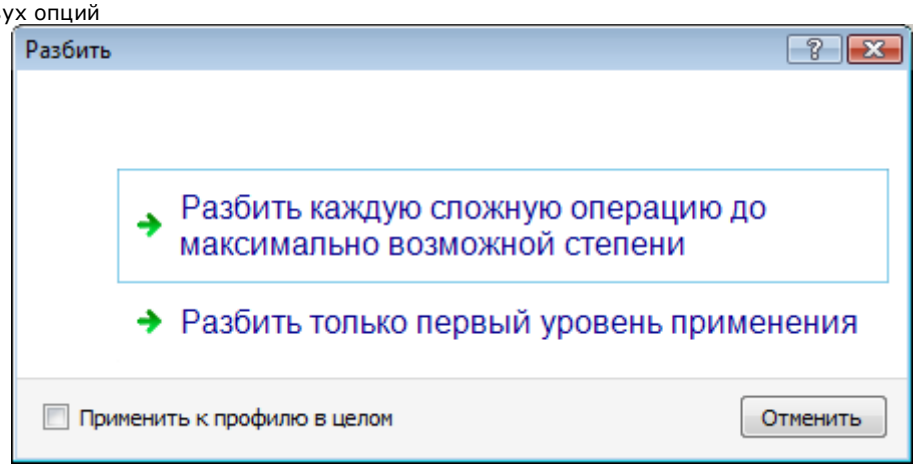
Типовыми случаями являются операции прорези, выполняемые неориентируемым инструментом.

Разбить

Разбивает сложные операции или множественные отрезки профиля на образующие их простые операции.

Команда **Разбить**  находится в группе **Общие** вкладки **Инструменты**.
Этот инструмент применяется к выбранным или текущей операции.
Инструмент "Разбить" может применяться не всегда. Исключаются сложные операции, которые:
- сконфигурированы в базе данных операций как операции, к которым нельзя применить трансформацию.
- в режиме "грань-деталь": решают коды запрограммированного вложенного вызова (SSIDE)

При выборе этой команды, только если она подключена в конфигурации TpaCAD, может открываться окно выбора из двух опций



- первая опция запрашивает максимально возможную разбивку для соответствующих программных строк.
- вторая опция запрашивает минимальную разбивку каждой программной строки, задействованной в применении этой команды.

Если это окно выбора не подключено в конфигурации TpaCAD, всегда применяется первая опция.

Если, например, у нас имеется вызов подпрограммы UNO, которая, в свою очередь, вызывает операцию многократного сверление (“РЯД ОТВЕРСТИЙ ПО X”), для которой не задано никакого предела разбивки:
1. в первом случае разбивка заменяет вызов подпрограммы списком отдельных сверлений операции “РЯД ОТВЕРСТИЙ ПО X”;
2. во втором случае разбивка сохраняет операцию “РЯД ОТВЕРСТИЙ ПО X”.

Если же для операции “РЯД ОТВЕРСТИЙ ПО X” задан предел разбивки, операция “РЯД ОТВЕРСТИЙ ПО X” не может, в любом случае, быть снова приведена к списку отдельных сверлений.

Может предлагаться опция ***Применить к профилю в целом*** в приведенном выше или отдельном окне: выбрать для применения этого инструмента ко всему текущему профилю или профилям, соответствующим выбранным операциям.

При применении команды **Разбить** теряются все развертки, соответствующие вложенным вызовам. В этом случае о ситуации предупреждает сообщение.

Если применение команды Разбить привело соответствующие операции только к простым операциям, гарантируется полное геометрическое соответствие измененной программы изначальной версии.
Если же применение команды оставило вызовы подпрограммы или макропрограммы неразбитыми, сигнал предупреждает, что измененная программа может не полностью соответствовать

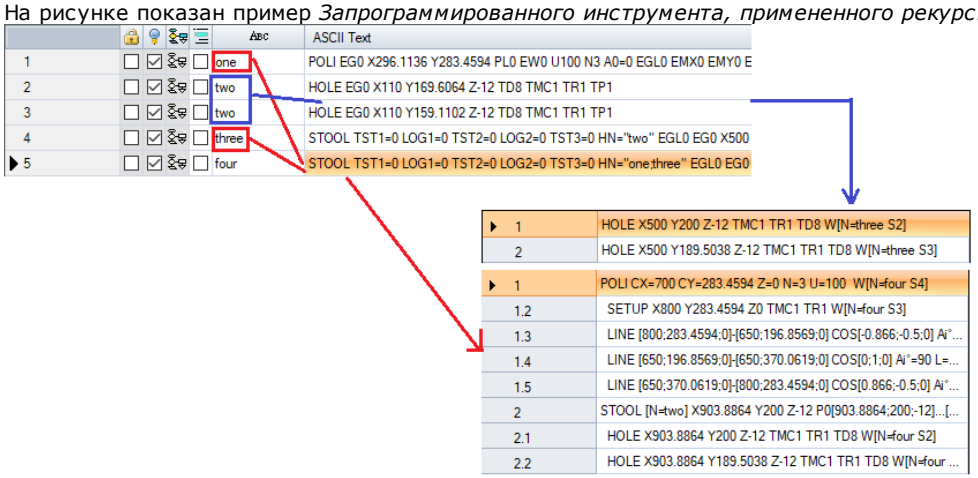
изначальной версии. Это может обуславливаться тем фактом, что не всегда можно применить требуемые трансформации (например, перенос, вращение, зеркальное отображение, пересмотр, инверсия и т.д.) ко всем ветвям развертки внутри операции. Другой критический случай соответствует скоплению трансформаций поворота и симметрий, с учетом того, что порядок применения этих трансформаций меняет конечный результат.

Если применение команды не выводит на уровень запрограммированного списка кодов запрограммированных инструментов, все добавленные строки будут иметь одно и то же *Имя,* заданное исходной программной строке.
В противном случае поле *Имя* добавленных строк может задаваться по другим правилам, направленным на сохранение исходной разработки программы.

Продвинутые положения

Хотим подчеркнуть тот факт, что сейчас мы решаем особую проблему, которая встречается только при определенной конфигурации TpaCAD при продвинутом программировании: возможность запроса минимальной разбивки.

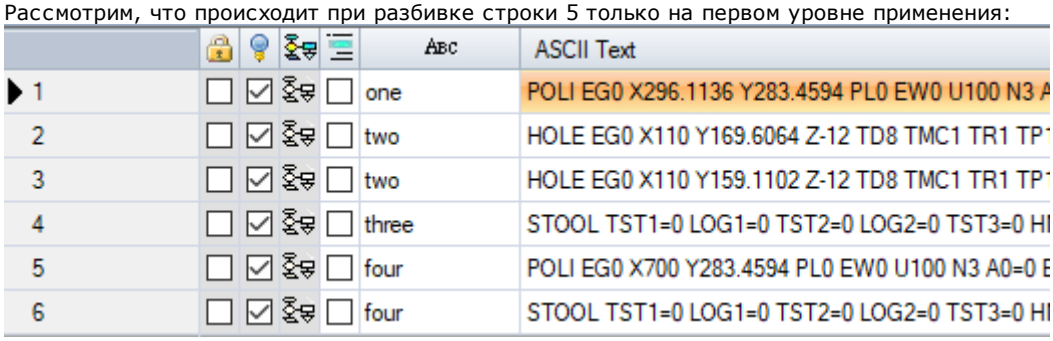
Рассмотрим следующие два особых случая.
На рисунке показан пример *Запрограммированного инструмента, примененного рекурсивно*:



Строка 4 - код STOOL, применяемый к операции с именем two” (операции HOLE строки 2 и 3):
- рядом показано окно, в котором представлен список операций, соответствующих коду STOOL
- строка 4 имеет имя “three”.

Строка 5 - код STOOL применяемый к операции с именем “one” (операция POLI строки 1, в примере это макропрограмма, которая строит: ПОДГОТОВКА + линейные) и one;three” (операция STOOL строки 4):
- рядом показано окно, в котором представлен список операций, соответствующих коду STOOL;
- строка 5 имеет имя “four”.

Рассмотрим, что происходит при разбивке строки 5 только на первом уровне применения:



Строка 5 разбивается на 2 строки:
- [5] с именем “four”: определяется применением операции POLI строки 1
- [6] с именем “four”: определяется применением операции STOOL строки 4.

Имена двух строк соответствуют имени исходной строки.

Попробуем теперь снова вызывать только что рассмотренную программу (назовем ее PRG1) с кодом SUB, а затем разобьем строку вызова только на первом уровне применения:

		ASCII Text
1	xone	POLI EG0 X296.1136 Y283.4594 PL0 EW0 U100 N3 A0=0 EGL0 EMX0 EMY0 EIN\
2	xtwo	HOLE X110 Y169.6064 Z-12 TD8 TMC1 TR1 TP1
3	xtwo	HOLE X110 Y159.1102 Z-12 TD8 TMC1 TR1 TP1
4	xthree	STOOL LOG1=0 TST2=0 LOG2=0 TST3=0 HN="xtwo" EGL0 EG0 X500 Y200 Z-12
▶ 5	xfour	STOOL TST1=0 LOG1=0 TST2=0 LOG2=0 TST3=0 HN="xone;xthree" EGL0 EG0 X

Список операций соответствует исходному тексту программы, только к именам было добавлено 'x'. Если бы строка кода SUB имела заданное имя (например, "yyy"), к именам добавлялось бы имя SUB (например, "yyyone" вместо "xone",..). Имена были изменены также в поле задания кодов STOOL (в строке 4: HN="xtwo"; в строке 5: HN=xone;xthree").
Новая номенклатура, созданная для добавленных операций, стремится снизить к минимуму вероятность того, что:

- код STOOL, приведенный в списке в результате частичной разбивки, может применяться также к существовавшим предусмотренным ранее операциям, заданным с тем же именем, которое используется в подпрограмме (например, "one")
- существовавший ранее расположенный далее код STOOL может также применяться к операциям, введенным в результате разбивки и заданным с тем же именем, которое используется в программе (например, "uno");
- в этом примере было бы неправильно присваивать разбитым строкам имя исходной операции (поле пустое или, например, "yyy"), т. к. два введенных кода STOOL больше не смогли бы найти операции применения, изначально заданные на уровне применения кода SUB.

Более, чем очевидно, что не существует абсолютной уверенности в том, что не возникнет конфликта между именами, созданными в результате частичной разбивки, и исходными именами в программном списке. Однако можно использовать описанный здесь механизм для изменения имен во избежание именно таких конфликтов. Механизм изменения имен должен, естественно, соблюдать максимальную длину имени (16 символов: лишние символы будут удаляться). Поэтому не рекомендуется использовать длинные имена для исключения их автоматического сокращения в серии имен.

Повторения

Свободные повторения

Выполняет копию операции/операций столько число раз, сколько указано в пункте **Повторения,** и помещает их на схему, в которой для каждой скоординированной оси задан сдвиг размещения.

Каждое поле окна может задавать одно числовое или параметрическое значение

Повторение операции, относящейся к профилю, влечет за собой:

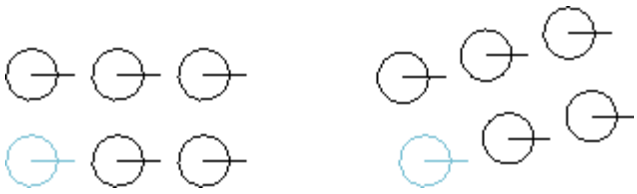
- повторение всего профиля, если этот инструмент применяется к операциям, скопированным в Локальный буфер, или к выбранным операциям, или если не выбрана опция **Применить от текущей операции до конца профиля**;
- в противном случае: выполняется повторение части профиля между текущей операцией и концом профиля: теперь сдвиги размещения определяются автоматически на основе габаритов части профиля, задействованной инструментом.

Прямоугольная серия

Выполняет копию операции/операций с позиционированием по матричной схеме.
Каждое поле окна может задавать числовое или параметрическое значение.

- **Столбцы, строки:** оба поля не могут быть 1, а общее число повторений не может быть больше 100.000. Развертка на строках всегда ассоциируется с осью Y грани, а на столбцах - с осью X грани.
- **Расстояние между столбцами** и **Расстояние между строками:** значимы со знаком.
- **Угол:** угол поворота (относительно оси X грани, плюс при вращении против часовой стрелки).

Пример:



Повторение операции с отмеченным отрезком для **3** столбцов и **2** строк.
На рисунке слева показана серия с **A**=0.
На рисунке справа показана те же серия с **A**#0.

Круговая серия

Выполняет копию операции/операций с позиционированием по круговой схеме.
Каждое поле окна может задавать числовое или параметрическое значение.

- **Центр X, Центр Y:** центр дуги, по которой разворачивается схема. Выбрать значок для считывания положения центра мышью в графической зоне.
- **Элементы серии:** число элементов серии, включая исходный. Задаваемое значение, в т. ч. параметрическое, должно быть больше 1**.**
- **Угол для заполнения, Угол между повторениями**: задаваемые значения, в т. ч. параметрические, должны находиться в пределах от 0.001° до 360°.

Из последних трех параметров должны задаваться два, третий вычисляется автоматически:

- **Элементы серии:** число элементов серии, включая исходный.
- **Угол для заполнения:** угол, который должен быть охвачен повторениями (включая исходные элементы).
- **Угол между повторениями:** угол между последовательными повторениями.

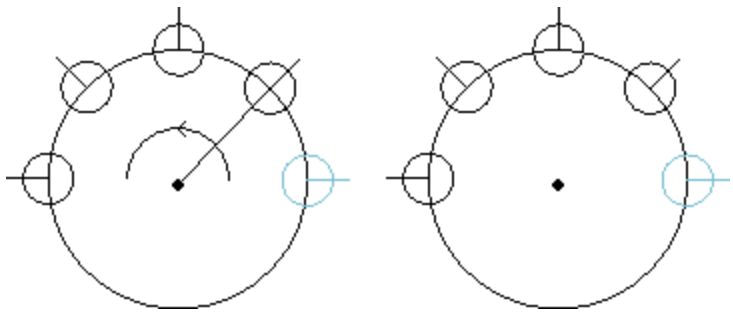
Рассмотрим приоритеты, с которыми оцениваются настройки:

- **Заданные Элементы серии** и **Угол для заполнения**: поле **Угол между повторениями** игнорируется и угол между элементами вычисляется автоматически.
- **Элементы серии** не задано: должны быть заданы оба поля, относящиеся к углам. Число элементов серии вычисляется автоматически.
- **Угол для заполнения** не задан. Должны быть заданы оба других поля.

- **Вращение против часовой стрелки:** выбрать для запроса развертки с вращением повторений против часовой стрелки.
- **Повернуть элементы серии:** выбрать для поворота элементов на каждом отдельном повторении для сохранения без изменений их развертки относительно центра вращения.

Применение инструмента определяет утрату всякой формы параметрического программирования, которая могла изначально присутствовать на плоскости грани.

Пример:



Повторение операции с отмеченным отрезком для **5** элементов, угол для заполнения: **180°**, вращение против часовой стрелки.
На рисунке слева показана серия с невыбранным полем **Повернуть элементы серии**.
Рисунок справа отображает эту же серию с выбранным полем **Повернуть элементы серии**.

Повторения на профиле

Выполняет копию операций столько число раз, сколько указано в пункте **Повторения,** и распределяет их по уже запрограммированному профилю**.**
Повторение операции, относящейся к профилю, обуславливает повторение всего профиля.
Исходные операции не меняются.

- **Повторения:** число элементов серии
- **Операция профиля:** номер по порядку программирования профиля, на котором выполнять распределение (любой отрезок). Выбрать значок для интерактивного считывания профиля. Профиль должен быть простым с дугами только в плоскости xy. и инструмент применяется к выборам, сам профиль распределения не может выбираться.

• Если активизирована опция **Позиционировать центр габарита**, в указанное положение переносится центр габаритного прямоугольника операций.
В качестве альтернативы можно перемещать ребро габаритного прямоугольника, выбирая одну из четырех кнопок:

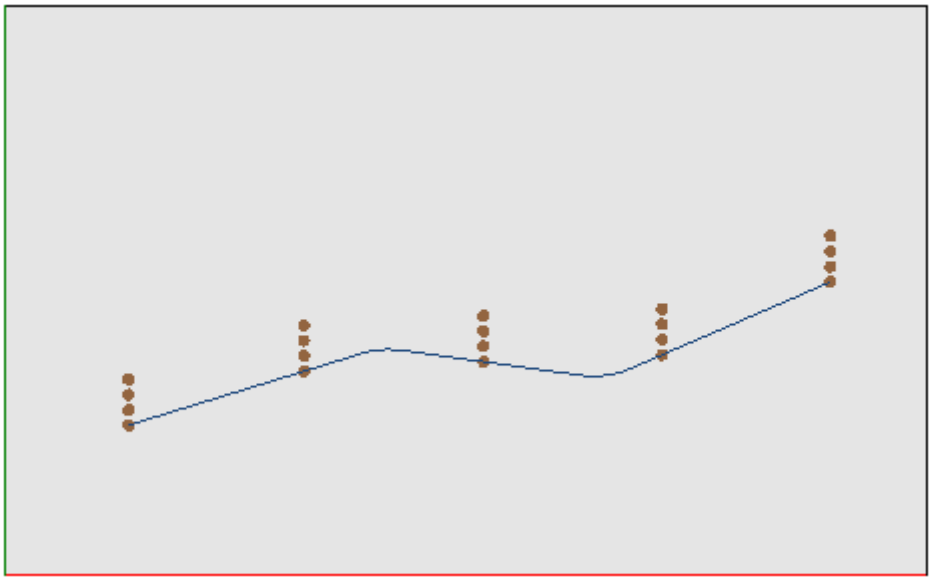
 переносит ребро на минимальные координаты по X и Y

 переносит ребро на максимальную координату по X и минимальную координату по Y

 переносит ребро на максимальные координаты по X и Y

 переносит ребро на минимальную координату по X и максимальную координату по Y

На рисунке показано повторение 4 отверстий, распределенных по вертикали: число повторений - 5.



10.3 Инструменты профиля

Изменить отрезок профиля

Команда **Изменить**  находится в группе **Изменить профили** вкладки **Инструменты.** Меняет отрезок текущего профиля, меняя геометрию, или удлиняет или укорачивает отрезок, меняя конечную точку. Текущий отрезок должен относиться к профилю, быть простым и иметь тип дуги или линии. Линейный отрезок не может иметь нулевую длину.
В режиме "грань-деталь" этот инструмент отключен, если активно каркасное представление, при текущей операции на нереальной грани.

Если отрезок **линейный,** можно задать:
- **Конечная точка:** перемещает конечную точку отрезка на координаты, запрограммированные в полях Координата X, Координата Y (щелкнуть значок  для считывания координат в интерактивном режиме). Перемещение конечной точки меняет направление отрезка. Можно изменить также координату конечной глубины отрезка (Координата Z).
- **Длина отрезка:** определяет Линейную длину отрезка в плоскости грани.
- **Касательная отрезка:** отрезок меняется с заданием касательной с предыдущим отрезком или заданием значения наклона.

Если отрезок - разметка (L24), можно задать:
- **Конечная точка**: перемещает конечную точку отрезка на координаты, запрограммированные в полях Координата X, Координата Y (щелкнуть значок  для считывания координат в интерактивном режиме). Перемещение конечной точки сохраняет без изменения направления начала и конца кривой.
- **Длина отрезка**: определяет расстояние между крайними точками кривой в плоскости грани.
- **Касательная в начале**: кривая меняется с заданием касательной с предыдущим отрезком или прямым заданием значения.
- **Касательная в конце**: кривая меняется с заданием значения касательной на конечной точке.

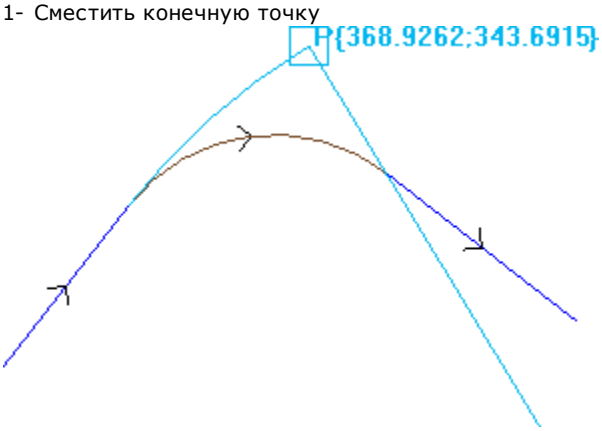
Если выбранный отрезок - **дуга,** заданная в плоскости грани, можно задать:

- **Конечная точка:** перемещает конечную точку отрезка на координаты, запрограммированные в полях Координата X, Координата Y (щелкнуть значок для считывания координат в интерактивном режиме). При перемещении конечной точки остается без изменений *Касательная в начале* дуги и эта точка не может совпадать с начальной точкой дуги. Можно изменить также координату конечной глубины отрезка (Координата Z).
- **Длина отрезка:** определяет Длину дуги в плоскости задания дуги (задаваемое значение ограничивается максимальной длиной круга) или Пройденный угол, выраженный в градусах (задаваемое значение уменьшается до значений в диапазоне от 0 до 360°). Пройденный угол может определяться также в интерактивном режиме.
- **Касательная в начале:** отрезок меняется с заданием касательной с предыдущим отрезком или заданием значения исходного наклона дуги
- **Касательная в конце:** отрезок меняется с заданием значения касательной на конечной точке дуги
- **Сместить центр:** перемещает центр дуги на координаты, запрограммированные в полях Координата X, Координата Y (в т. ч. в интерактивном режиме).
- **Сместить среднюю точку:** перемещает среднюю точку дуги на координаты, запрограммированные в полях Координата X, Координата Y (в т. ч. в интерактивном режиме).

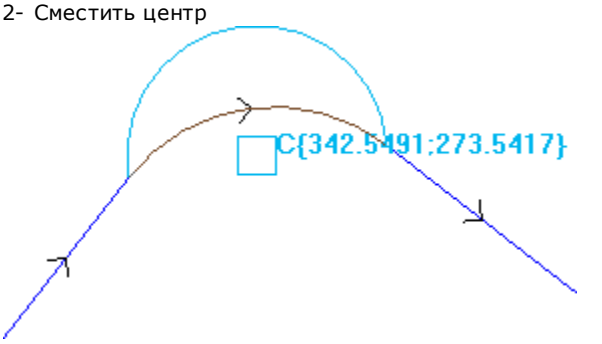
Если выбранный отрезок - **круг,** заданный в плоскости грани, можно задать:
- **Длина отрезка:** определяет Длину дуги в плоскости задания дуги (задаваемое значение ограничивается максимальной длиной круга) или Пройденный угол, выраженный в градусах (задаваемое значение уменьшается до значений в диапазоне от 0 до 360 градусов). Пройденный угол может определяться также в интерактивном режиме.

При перемещении мыши в графической зоне можно увидеть, как меняется дуга.
Рассмотрим три различные ситуации для изменения отрезка типа дуги:

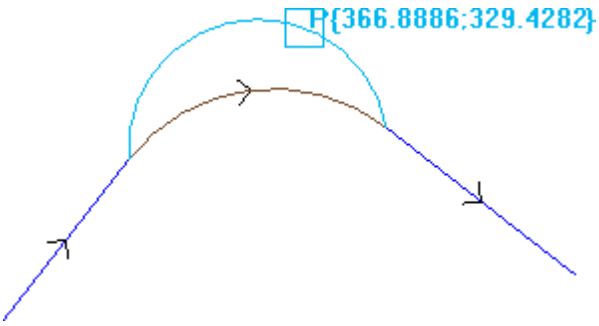
1- Сместить конечную точку



2- Сместить центр



3- Сместить среднюю точку



Если отрезок - **дуга конуса (эллипс),** можно изменить длину отрезка, выраженную как линейная или угловая длина:
- **Длина отрезка:** определяет линейную длину дуги в плоскости грани. Задаваемое значение ограничивается максимальной длиной полного конуса;
- **Пройденный угол:** выражен в градусах (в т. ч. в интерактивном режиме). Заданное значение уменьшается до значений в диапазоне от 0 до 360 градусов.

Если выбранный отрезок - **дуга** или **круг** в плоскости, отличной от плоскости грани, можно задать:
- **Длина отрезка:** определяет линейную длину дуги в плоскости задания дуги. Задаваемое значение ограничивается максимальной длиной круга;
- **Пройденный угол:** выражен в градусах, может задаваться только напрямую. Заданное значение уменьшается до значений в диапазоне от 0 до 360 градусов.

Изменение отрезка может вызывать изменение рабочего кода текущей операции.
Инструмент не работает, если текущая операция не проверяет активные фильтры изображения (варианты выбора, логические условия, уровни, специальные фильтры) или если профиль был заблокирован (это вложенный вызов или имеет заблокированный уровень, конструкцию или поле O).
Инструмент дезактивирован, если:
- нет запрограммированных операций
- текущая операция - недействительного типа.

Заменить ребро на дугу

Команда **Заменить ребро на дугу**  находится в группе **Изменить профили** вкладки **Инструменты**.
Заменить выбранное ребро на дугу.
В режиме "грань-деталь" этот инструмент отключен, если активно каркасное представление, при текущей операции, заданной на нереальной грани.
Инструмент работает на расширенных профилях, но применение возможно только к ребру, определенному двумя простыми линейными отрезками. Инструмент работает непосредственно на текущем профиле. В этом окне отображается плоскость, на которой покоятся три точки, ограничивающие ребро, как плоскость, на которой вычислять дугу. Если на выбранной плоскости не существует геометрических условий для определения дуги, преобразование не выполняется.
Инструмент дезактивирован, если:
- нет запрограммированных операций;
- текущая или следующая операция не являются линейными отрезками;
- три вершины ребра не различны или находятся на одной линии;
- в режиме "грань-деталь", если активно каркасное представление, при текущей операции, заданной на нереальной грани.

Изменить линию в разметку



Команда **Изменить линию в разметку**  имеется в группе **Изменить профили** вкладки **Инструменты**.

Меняет отрезок текущего профиля в операции L24, соответствующей элементу, именуемому **Разметка**.

Отрезок текущего профиля должен быть линейным.

Эта операция описывается в параграфе **Операции->Профиль->Разметка.**

Применить вход к профилю

Команда **Применить вход к профилю**  находится в группе **Изменить профили** вкладки **Инструменты.** В режиме "грань-деталь" этот инструмент отключен, если активно каркасное представление, при текущей операции, заданной на нереальной грани. Добавляет линейный отрезок или часть окружности открытия в текущий профиль. Если текущий профиль открытый, добавляется операция setup в качестве начальной точки добавленного отрезка, или же подготовка смещается в новую начальную точку профиля.
В режиме "грань-деталь" этот инструмент отключен, если активно каркасное представление, при текущей операции, заданной на нереальной грани.

Можно выбрать различные опции ввода:

- **Вставить линейный отрезок при заданных координатах:** вставляет линейный отрезок, который начинается из точки, определенной запрограммированными координатами x и y (в т. ч. в интерактивном режиме) и завершается в начальной точке исходного профиля. При выборе соответствующего режима значения суммируются с начальным положением исходного профиля. При выборе пункта **Вставляет два отрезка перемещения (Z + XY)** запрограммированное перемещение делится на два линейных отрезка:
  - подготовка профиля перемещается в запрограммированное положение
  - первый линейный отрезок выполняет перемещение по оси Z до начальной координаты Z исходного профиля
  - второй линейный отрезок выполняет перемещение по XY до достижения изначального положения подготовки.

  Начальная глубина дополнительного отрезка может задаваться в поле **Координата Z**: при выборе относительного режима это значение применяется с учетом начальной глубины исходного профиля.
- **Вставить линейный отрезок по касательной:** вставляет линейный отрезок длины, определенной **Модулем,** а направление отрезка задается так, чтобы сохранить исходное направление исходного профиля При выбранном пункте **Применить в трехмерном виде** обуславливает непрерывность касательной в пространстве при запуске профиля: направление отрезка и исходная глубина дополнительного отрезка определяются автоматически с первого отрезка исходного профиля. Если пункт **Применить в трехмерном виде** не выбран: начальная глубина дополнительного отрезка может задаваться в поле **Координата Z (**при выборе относительного режима это значение применяется с учетом начальной глубины исходного профиля). При выборе пункта **Вставляет два отрезка перемещения (Z + XY)** запрограммированное перемещение делится на два линейных отрезка, как уже отмечалось для предыдущего случая.
- **Вставить круговой отрезок по касательной:** вставляет дугу в плоскость xy грани, а направление отрезка задается так, чтобы сохранить исходное направление исходного профиля. Параметры Координата X и Координата Y представляют абсолютные или относительные координаты начальной точки дуги (в т. ч. в интерактивном режиме). Начальная глубина отрезка определяется параметром Координата Z (при задании в относительном режиме это значение применяется с учетом начальной глубины исходного профиля).
- **Вставить трехмерную дугу по касательной:** вставляет часть окружности, определенную **Радиусом** и шириной Пройденного угла, выраженной в градусах, с направлением, заданным в пространстве так, чтобы сохранить исходное направление исходного профиля. Значение ширины угла должно находиться в пределах от 1.0° до 90°. Если невозможно определить дугу, определяется линейный отрезок длины, равной **Радиусу**, с непрерывностью касательной при запуске профиля. Решение отрезка аналогично применяемому в операциях подготовки для программирования начального отрезка профиля.
- **Вставить отрезок с перекрытием**: вставляет отрезок заданной длины, дублирующий геометрию последнего и/или первого отрезка исходного профиля. Выбор отрезка с перекрытием возможен только если исходный профиль закрытый и завершается отрезком профиля. Определяются следующие параметры:
  - **Применить перекрытие в начале:** при активизации требует вставки в начало профиля отрезка с перекрытием (полным или частичным) последнего отрезка исходного профиля. Задать поля:
    - **Модуль**: длина дополнительного отрезка. Это поле инициализируется на значении длины последнего отрезка профиля. При задании значения, равного нулю или больше инициализированного значения, перекрытие будет полным.
    - **Координата Z**: начальная глубина отрезка. Если координата находится в относительном режиме, она применяется относительно начальной глубины исходного профиля. Координата глубины не учитывается в случае, когда отрезок перекрытия - типа дуги с разверткой на плоскости, отличной от xy.
  - **Применить перекрытие в конце:** при активизации требует вставки в конец профиля отрезка с перекрытием (полным или частичным) первого отрезка исходного профиля. Эта опция может не выбираться, если геометрия профиля не позволяет вставлять отрезок с перекрытием в конец. Задать поля:
    - **Модуль**: длина дополнительного отрезка. Это поле инициализируется на значении длины первого отрезка профиля. При задании значения, равного нулю или больше инициализированного значения, перекрытие будет полным.
    - **Координата Z**: конечная глубина отрезка. Если координата находится в относительном режиме, она применяется относительно конечной глубины исходного профиля. Координата глубины не

учитывается в случае, когда отрезок перекрытия - типа дуги с разверткой на плоскости, отличной от xy.

Применить выход к профилю

Команда **Применить выход к профилю**  находится в группе **Изменить профили** вкладки **Инструменты.** Вставляет линейный или круговой отрезок закрытия в текущий профиль. В режиме "грань-деталь" этот инструмент отключен, если активно каркасное представление, при текущей операции, заданной на нереальной грани.

Можно выбрать различные опции ввода:

- **Вставить линейный отрезок при заданных координатах:** вставляет линейный отрезок от конечной точки профиля до точки, определенной запрограммированными координатами x и y (в т. ч. в интерактивном режиме). При выборе соответствующего режима значения суммируются с конечным положением исходного профиля. При выборе пункта **Вставляет два отрезка перемещения (XY + Z)** запрограммированное перемещение делится на два линейных отрезка:
  - конечная точка профиля перемещается в запрограммированное положение
  - Первый линейный отрезок выполняет перемещение по XY начиная с конечного положения исходного профиля до достижения запрограммированного положения
  - второй линейный отрезок выполняет перемещение по оси Z до запрограммированной конечной координаты Z

  Конечная глубина дополнительного отрезка может задаваться в поле **Координата Z**: при выборе относительного режима это значение применяется с учетом конечной глубины исходного профиля.
- **Вставить линейный отрезок по касательной:** вставляет линейный отрезок длины, определенной **Модулем,** а направление отрезка задается так, чтобы сохранить направление закрытия исходного профиля. При выбранном пункте **Применить в трехмерном виде** обуславливает непрерывность касательной в пространстве при закрытии профиля: направление отрезка и конечная глубина дополнительного отрезка определяются автоматически с последнего отрезка исходного профиля. Если пункт **Применить в трехмерном виде** не выбран, конечная глубина дополнительного отрезка может задаваться в поле **Координата Z (**при выборе относительного режима это значение применяется с учетом конечной глубины исходного профиля). При выборе пункта **Вставляет два отрезка перемещения (XY + Z)** запрограммированное перемещение делится на два линейных отрезка, как и в указанном выше случае.
- **Вставить круговой отрезок по касательной:** вставляет часть окружности в плоскость xy грани, а направление отрезка задается так, чтобы сохранить направление закрытия исходного профиля. Параметры Координата X и Координата Y представляют абсолютные или относительные координаты конечной точки дуги (в т. ч. в интерактивном режиме). Конечная глубина отрезка определяется параметром Координата Z (при задании в относительном режиме это значение применяется с учетом конечной глубины исходного профиля).
- **Вставить трехмерную дугу по касательной:** вставляет часть окружности, определенную **Радиусом** и шириной **Пройденного угла,** выраженной в градусах, с направлением, заданным в пространстве так, чтобы сохранить направление закрытия исходного профиля. Значение ширины угла должно находиться в пределах от 1.0° до 90°. Если невозможно определить дугу, определяется линейный отрезок длины, равной **Радиусу**, с непрерывностью касательной при закрытии исходного профиля. Решение отрезка аналогично применяемому в операциях подготовки в части программирования выходного отрезка профиля.
- **Вставить отрезок с перекрытием**: вставляет отрезок заданной длины, дублирующий геометрию первого отрезка исходного профиля. Выбор отрезка с перекрытием возможен только если исходный профиль закрытый и завершается отрезком профиля. Определяются следующие параметры:
  - **Модуль**: длина дополнительного отрезка. Это поле инициализируется на значении длины первого отрезка профиля. При задании значения, равного нулю или больше инициализированного значения, перекрытие будет полным.
  - **Координата Z**: конечная глубина отрезка. При задании в относительном режиме значение применяется с учетом начальной глубины профиля. Координата глубины не учитывается в случае, когда отрезок перекрытия - типа дуги с разверткой на плоскости, отличной от xy.

Замкнуть профиль

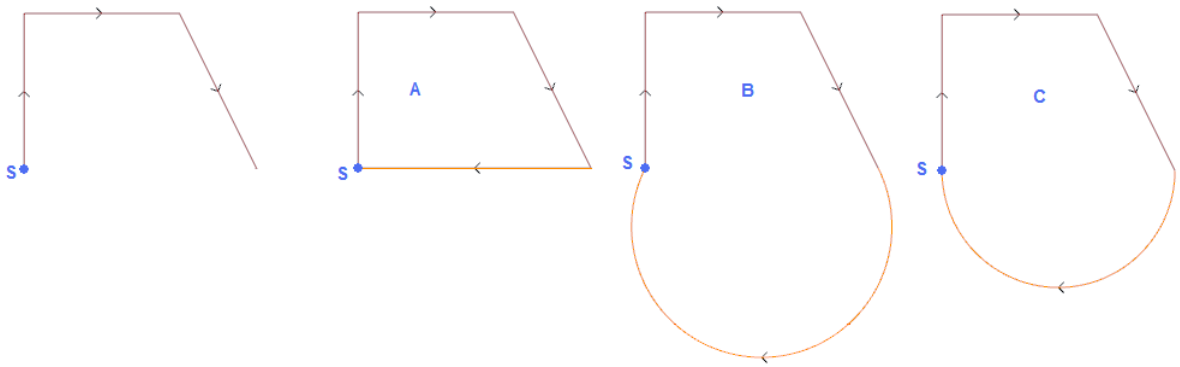
Применяет к текущему профилю линейный или круговой отрезок закрытия. Исходный профиль не может быть замкнут. Команда **Замкнуть профиль**  находится в группе **Изменить профили** вкладки **Инструменты.**

Тип вставляемого отрезка выбирается из трех опций:

- **Замкнуть профиль линейным отрезком:** замыкает профиль, вводя линейный отрезок, соединяющий последнюю точку профиля с точкой подготовки.
- **Замкнуть профиль дугой, касательной к концу последнего отрезка:** замыкает профиль, вводя дугу непрерывно по касательной с последним отрезком исходного профиля

- **Замкнуть профиль дугой, касательной к началу первого отрезка:** замыкает профиль, вводя дугу непрерывно по касательной с первым отрезком исходного профиля.

Рассмотрим пример замыкания простого профиля с тремя возможными опциями:
- слева показан исходный незамкнутый профиль: S - точка начала (точка подготовки), движение осуществляется против часовой стрелки;
- фигура A: профиль замкнут линейным отрезком
- фигура B: профиль замкнут дугой, касательной к последнему отрезку исходного профиля
- фигура C: профиль замкнут дугой, касательной к первому отрезку профиля



Инвертировать профиль

Инвертирует направление прохождения выбранных или текущего профилей. Команда **Инвертировать профиль**  находится в группе **Изменить профили** вкладки **Инструменты.**
Этот инструмент применим:
- ко всем профилям, имеющим по меньшей мере один выбранный элемент
- к текущему профилю.

При выполнении команды можно применять изменения непосредственно к исходным профилям или копии.
Инструмент инвертирует также настройки:
- коррекции на инструмент (правый или левый);
- выбора начального/конечного отрезков профиля в случае задания правой или левой дуги.

При подключении из конфигурации TpaCAD применение этого инструмента может предусматривать зеркальную технологию.
При подключении из Конфигурации TpaCAD применение этого инструмента к ориентируемой подготовке может предусматривать трансформацию для осей ориентации (только если текущая грань плоская, т. е. не закруглена или задана как поверхность).

Этот инструмент обнуляет изменения коррекции на инструмент на маршруте (прерывания, приостановки и возобновления и/или изменения стороны).

Масштабировать профиль

Применяет масштабный коэффициент к одному или нескольким профилям. Команда **Масштабировать профиль**  находится в группе **Изменить профили** вкладки **Инструменты**.
Применяется к:
- ко всем профилям, имеющим по меньшей мере один выбранный элемент
- к текущему профилю

Выбрать порядок задания масштаба:
- **Абсолютная шкала:** напрямую задает **масштабный коэффициент**. Значение больше 1.0 увеличивает профиль, значение от 0.01 до 1.0 (не включая его) уменьшает его.
- **Относительная шкала:** масштабный коэффициент определяется заданием двух значений:
  - **Исходная длина**: текущее эталонное значение
  - **Новая длина**: значение, измененное после шкалы.

Например, при задании для этих двух значений, соответственно, 5.0 и 10.0, отрезок длиной 5.0 увеличивается с применением абсолютного масштабного коэффициента, равного 2.0, полученного как отношение между 10.0 и 5.0.

- **Масштаб по габаритному прямоугольнику**: выбрать для задания базовой точки в центре габаритного прямоугольника профилей, участвующих в трансформации. Если этот пункт не выбран, задать базовую точку в:

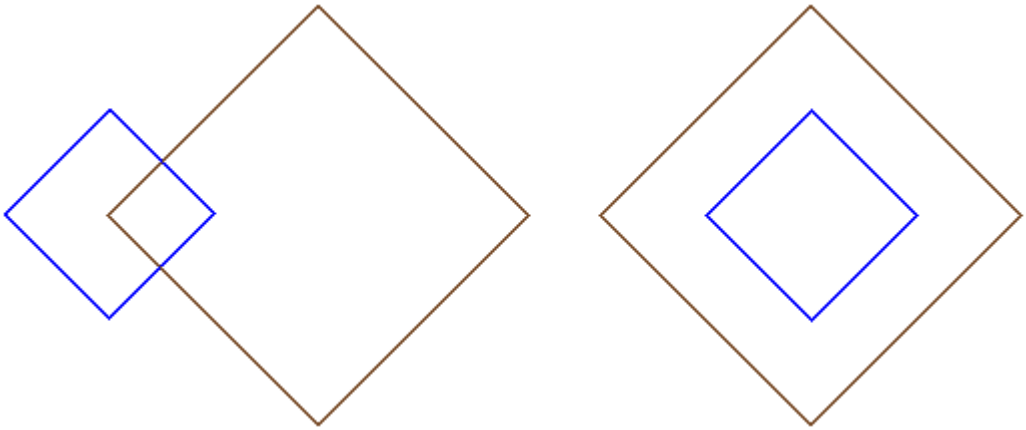
- **Координата X Координата Y** : координаты программируются в полях Координата X, Координата Y (в т. ч. в интерактивном режиме, при щелчке на значке).

- **Применить в трехмерном виде**: подключить для применения масштаба также на глубине грани (в противном случае - только в плоскости xy). Этот выбор обязателен, если соответствующие профили выполняют дуги в плоскостях #xy.

Для поля **Масштабный коэффициент** можно подключить интерактивный режим (щелчком на значке): графика содержит габаритный прямоугольник профилей, задействованных в трансформации, масштабированный относительно базовой точки. Масштабный коэффициент можно изменить при помощи колесика мыши или кнопок прибавления (+) или вычитания (-). В интерактивном режиме масштабный коэффициент можно менять в пределах крайних значений (0.2; 2.0);

Применение к профилям вызывает:
- изменение всего профиля, если инструмент применяется к выбранным операциям, или если не выбрана опция **Применить от текущего положения до конца профиля**;
- в противном случае: изменение части профиля между текущей операцией и концом профиля: теперь базовая точка совпадает с начальной точкой текущей операции. Если активизирован выбор опции **Применить к копии операций**, в любом случае, вставляется копия всего профиля.

Выбор опции *Применить к копии операций* применяет этот инструмент к копии операций и не меняет исходные линии.



Рассмотрим пример:

Требуемый масштабный коэффициент: **0,5**
Изменить задание **базовой точки:**
- Рисунок слева: точка (X, Y...), указанная слева от профилей;
- Рисунок справа: центровка на габаритном прямоугольнике.

Выполнение инструмента уменьшает в два раза размер каждого отрезка профиля, а каждый профиль уменьшает в два раза свое расстояние от указанной базовой точки. Если профилям задан также начальный и/или конечный отрезок, масштабный коэффициент применяется и к ним.

Растянуть профиль

Применяет масштабный коэффициент к одному или нескольким профилям с возможностью дифференцирования масштаба, применяемого по x и по y. Команда **Растянуть профиль** находится в группе **Изменить профили** вкладки **Инструменты.**

- **Масштабный коэффициент по X, Масштабный коэффициент по Y**: масштабный коэффициент задается с абсолютными значениями, дифференцируемыми по направлению X и Y.

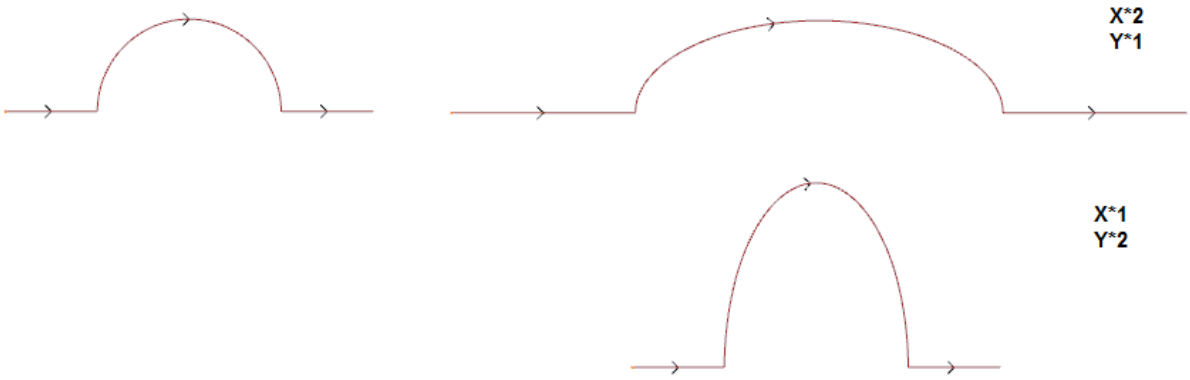
- **Масштаб по габаритному прямоугольнику**: выбрать для автоматического задания базовой точки в центре габаритного прямоугольника профилей, участвующих в трансформации. Если этот пункт не выбран, задать базовую точку в:
- **Координата X Координата Y**: координаты программируются в полях Координата X, Координата Y (в т. ч. в интерактивном режиме, при щелчке на значке).

Для полей **Масштабный коэффициент по X/Y** можно подключить интерактивный режим (щелчком на значке): графика содержит габаритный прямоугольник профилей, задействованных в трансформации, масштабированный относительно базовой точки и заданного масштаба. Отдельный масштабный коэффициент можно изменить при помощи колесика мыши или кнопок прибавления (+) или вычитания (-). В интерактивном режиме масштабный коэффициент можно менять в пределах крайних значений (0.2; 2.0);

Применение к профилям вызывает:
- изменение всего профиля, если инструмент применяется к выбранным операциям или если не выбрана опция **Применить от текущего положения до конца профиля**;
- в противном случае: изменение части профиля между текущей операцией и концом профиля: теперь базовая точка совпадает с начальной точкой текущей операции. Если активизирован также выбор опции **Применить к копии операций**: в любом случае, вставляется копия всего профиля.

Выбор опции *Применить к копии операций* применяет этот инструмент к копии операций и не меняет исходные линии.

Растягивание дуги автоматически создает дугу эллипса. В случае профилей с заданными начальными и/или конечными отрезками масштаб применяется также к геометрии отрезков только если масштабные коэффициенты одинаковы.



- слева - исходный профиль
- справа - результирующий профиль, с примененными масштабными коэффициентами как указано: в обоих случаях, исходный полукруг был изменен полуэллипсом.

Разделить профиль

Команда **Разделить профиль** находится в группе **Изменить профили** вкладки **Инструменты.** Этот инструмент позволяет удалять часть текущего профиля или выполнять фрагментацию отрезка профиля на два отдельных отрезка.

- **Точка резки**: присваивает положение (X, Y) точки резки на текущем профиле (щелкнуть значок для позиционирования мышью в графической зоне);

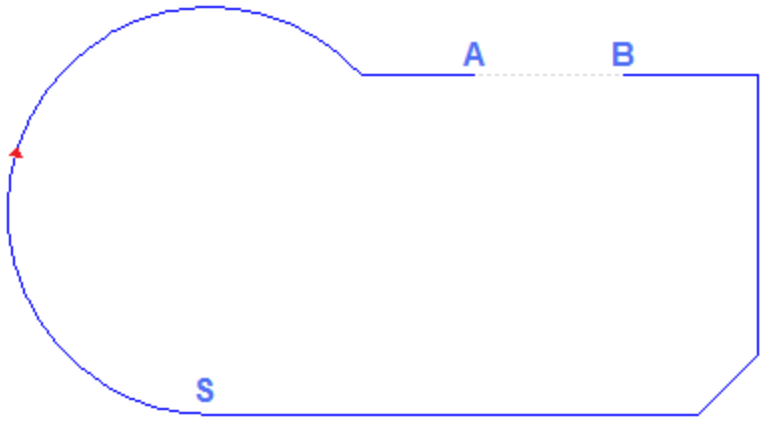
- **Указать вторую точку резки**: выбрать для задания второй точки на профиле и удаления части профиля между этими двумя точками.

Если этот пункт не выбран: инструмент разделяет профиль в первой заданной точке резки. В этом случае выделяются два случая, в зависимости от состояния пункта **Резать профиль**:
- при выборе профиль режется на точке и получаются два профиля: первый завершается на точке резки, а второй начинается из точки резки и содержит конечную часть исходного профиля.
- если этот пункт не выбирается, отрезок профиля, на котором оказывается точка резки, делится на два, но профиль остается единым.

Если пункт **Указать вторую точку резки** выбран, задать напрямую или интерактивно вторую точку резки.

При определении положения точек резки мышью координаты задаются так, чтобы соответствовать точке вдоль профиля. При прямом задании или изменении координат осуществляется поиск точки вдоль профиля, ближе всего расположенной к заданной.
На рисунке представлен пример профиля:

- (S) означает точку начала профиля
- стрелка указывает на перемещение против часовой стрелки
- профиль в примере замкнут



На профиле были указаны 2 точки резки: (A) и (B) (эти две точки могут выпадать на один отрезок или на отдельные отрезки). Часть профиля между двумя точками удаляется (**ВНИМАНИЕ**: в изначальном направлении перемещения).
Таким образом, исходный профиль будет разбит инструментом на 2 профиля:
- 1-й профиль начинается в (S) и приходит в (A);
- 2-й профиль начинается в (B) и приходит в (S);

Для разбивания данного линейного отрезка в одной точке (например, (A)), достаточно не указывать вторую точку резки.
В этом случае профиль останется единым, но только с еще одним отрезком.
Для разделения на два отдельных профиля на точке (A) достаточно не указывать вторую точку резки и выбрать пункт ***Резать профиль***

Отсоединить каждый отрезок профиля

Команда **Отсоединить каждый отрезок профиля**  находится в группе **Изменить профили** вкладки **Инструменты.** Этот инструмент меняет текущий профиль или его копию, разделяя его на каждый отдельный отрезок, определяя столько же отдельных профилей.

Если исходный профиль начинается с подготовки, можно начинать любой новый отделенный профиль его копией или, если возможно, - применить координаты начальной точки непосредственно на каждом отдельном отрезке.

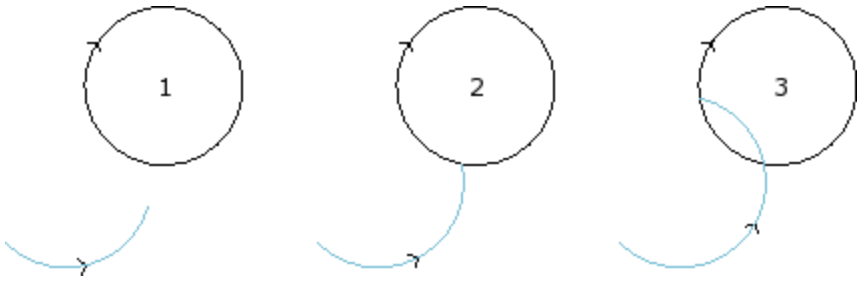
Продлить

Команда **Продлить**  находится в группе **Изменить профили** вкладки **Инструменты.** Этот инструмент расширяет отрезок текущего профиля (текущий отрезок или последний отрезок профиля) до пересечения с выбранным ограничивающим элементом.
Текущий отрезок должен относиться к профилю и быть типа:
- дуги, но не может быть кругом
- или линии ненулевой длины
- если элемент разметки (L24): он рассматривается как линейный отрезок.

Возможные варианты выбора ограничивающего элемента:
- **продолжить до программного элемента:** ограничивающий элемент определяется запрограммированной операцией профиля, за исключением точечных и подготовительных операций. Элемент определяется заданием точки, ближайшей к отрезку: координаты X и Y программируются в полях редактирования (в т. ч. в интерактивном режиме при щелчке на значке ). Отрезок удлиняется до пересечения с профилем, ближе всего расположенным к точке выбора. При нахождении нескольких решений пересечения действительно решение, ближайшее к начальной точке.
- **продолжить до вертикальной прямой:** ограничивающий элемент определяется вертикальной прямой. Координата вертикальной оси программируется в поле редактирования (также в интерактивном режиме при щелчке на значке ).

- **продолжить до горизонтальной прямой:** ограничивающий элемент определяется горизонтальной прямой. Координата горизонтальной оси программируется в поле редактирования (также в интерактивном режиме при щелчке на значке).
- **продолжить до пересечения с гранью:** ограничивающий элемент определяется габаритным прямоугольником грани. Отрезок удлиняется до пересечения со стороной грани.
- **продолжает до пересечения с прямоугольником:** ограничивающий элемент задается прямоугольником. Габарит прямоугольника программируется в поле редактирования (также в интерактивном режиме при щелчке на значке).

На рисунке показаны случаи применения инструмента.



Дуга представляет отрезок, который должен продолжаться до точки пересечения с окружностью

- рисунок **1** соответствует исходной ситуации
- рисунок **2** соответствует первому применению продолжения
- рисунок **3** соответствует второму применению продолжения

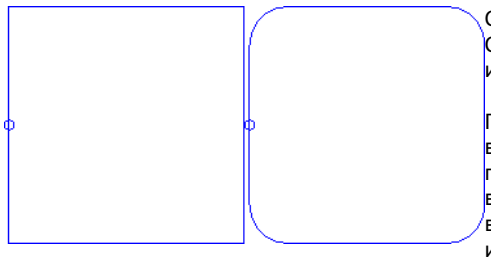
Скруглить профиль

Команда **Скруглить профиль** находится в группе **Изменить профили** вкладки **Инструменты.** Этот инструмент вставляет дуги скругления на ребра профиля и каждое скругление определяется так, чтобы обеспечить непрерывность касательной с исходными отрезками ребер.
Исходные ребра могут быть ограничены: двумя линейными отрезками, линейным отрезом и дугой, двумя дугами.
Этот инструмент не разбивает сложные операции или множественные отрезки, относящиеся к профилю: при необходимости применить вначале инструмент **Разбить** (группа **Общие** вкладки **Инструменты)**.
Этот инструмент применяется напрямую только к текущему профилю.

- **Радиус скругления:** радиус дуги скругления, вставленной на ребро.

Применить к:

- **применить только к текущему положению**: вставляет скругление на ребро, образованное текущей операцией с последующей операцией
- **применить от текущего положения до конца профиля**: вставляет скругления на каждое ребро профиля начиная с текущей операции
- **применить к профилю в целом**: вставляет скругления на все ребра профиля



Слева представлен прямоугольник с острыми ребрами. Справа представлена конечная ситуация после применения инструмента ко всему профилю.

Применение этого инструмента возможно только к ребрам с габаритом, позволяющим ввести дугу скругления внутри (а не снаружи) исходных отрезков.

PROFESSIONAL

Можно генерировать профили с вводом дуг скругления в точках ребер профиля также в форме сложной операции, вызывая операцию типа Запрограммированные инструменты в списке операций. В группе ИНСТРУМЕНТЫ выбирается операция STOOL:СКРУГЛИТЬ ПРОФИЛЬ:

- Поле **Операции** задает имена, заданные запрограммированным ранее операциям, соответствующим исходным профилям.

Профили могут быть получены также в результате применения других сложных кодов и выполнение операции соответствует только измененному профилю и не включает исходные профили. Операции, не используемые для целей требуемых функций (например, точечные или логические операции или сложные неразбиваемые операции), игнорируются.

В операции задаются:
- Типичные параметры сложной операции (см. информацию, приведенную об общем коде подпрограммы):
  - **Qx,Qy Zp**: начальные координаты размещения разработанных операций
  - **...**
  - **Свойства операции**: задает свойства, присвоенные операции

- Особые параметры функции операции со значением, аналогичным полям, определенным в окне инструмента:
  - **Радиус скругления:** радиус дуги скругления, вставленной на ребрах
  - **Применить к острым углам:** при выборе активирует применение скругления только к ребрам, лежащим внутри прямого угла (<90°)
  - **Применить только к вершинам с дугой:** при выборе активирует применение скругления только к ребрам, заданным между линией и дугой, дугой и линией, дугой и дугой. Следовательно, при выборе исключаются ребра, заданные между линией и линией.

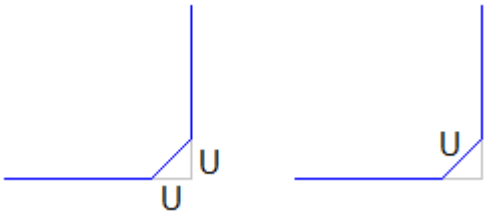
Основное преимущество использования операцииSTOOL:СКРУГЛИТЬ состоит в том, что генерированные профили адаптируются к изменениям исходных профилей, кроме того, можно работать в общей сложности на нескольких профилях, в т. ч. сложных.

Выполнить фаску на профиле

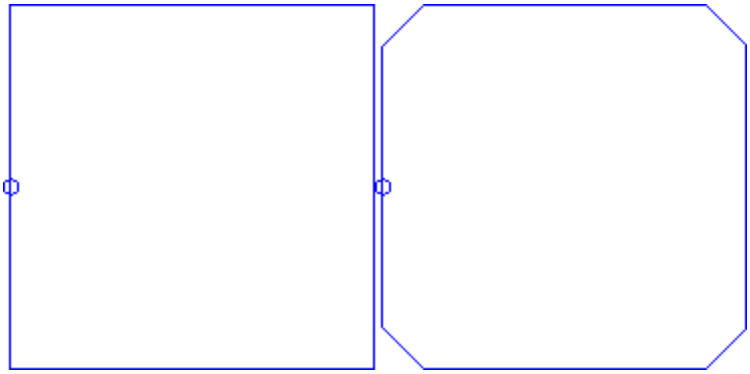
Команда **Выполнить фаску на профиле**  находится в группе **Изменить профили** вкладки **Инструменты.** Этот инструмент вставляет фаски на ребра профиля: могут быть изменены только ребра, ограниченные двумя линейными отрезкам. Этот инструмент не разбивает сложные операции или множественные отрезки, относящиеся к профилю: при необходимости применить вначале инструмент **Разбить** (группа **Общие** вкладки **Инструменты)**.
Этот инструмент применяется напрямую только к текущему профилю.

- **Вектор:** длина, заданная для определения фаски.

Имеется две опции:
- **Вектор задает расстояние от ребра**: поле *Вектор* задает расстояние выступа от крайних точек фаски до исходного ребра (рисунок слева: U указывает заданное значение).
- **Вектор задает длину фаски:** поле *Вектор* задает длину фаски (рисунок справа).



Применить к:
- **применить только к текущему положению**: вставляет фаску на ребро, образованное текущей операцией с последующей операцией
- **применить от текущего положения до конца профиля**: вставляет фаски на каждое ребро профиля начиная с текущей операции
- **применить к профилю в целом**: вставляет фаски на все ребра профиля



Слева представлен прямоугольник с острыми ребрами.
Справа представлена конечная ситуация после применения инструмента ко всему профилю.

Применение этого инструмента возможно только к ребрам с габаритом, позволяющим ввести фаску внутри (а не снаружи) исходных отрезков.

PROFESSIONAL

Можно генерировать профили с вводом фасок в точках ребер профиля также в форме сложной операции, вызывая операцию типа Запрограммированные инструменты в списке операций. В группе ИНСТРУМЕНТЫ выбирается операция STOOL:ВЫПОЛНИТЬ ФАСКУ НА ПРОФИЛЕ:

- Поле **Операции** задает имена, заданные запрограммированным ранее операциям, соответствующим исходным профилям.

Профили могут быть получены также в результате применения других сложных кодов и выполнение операции соответствует только измененному профилю и не включает исходные профили. Операции, не используемые для целей требуемых функций (например, точечные или логические операции или сложные неразбиваемые операции), игнорируются.

В операции задаются:

- Типичные параметры сложной операции (см. информацию, приведенную об общем коде подпрограммы):
  - **Qx,Qy Zp**: начальные координаты размещения разработанных операций
  - **...**
  - **Свойства операции**: задает свойства, присвоенные операции

- Особые параметры функции операции со значением, аналогичным полям, определенным в окне инструмента:
  - **Фаска:** длина, заданная для определения фаски
  - **Тип:** присваивает вид выполняемой фаски
    - **Фаска**= значение, заданное в параметре **Фаска** – это длина отрезка фаски
    - **Отрезки от вершины=** значение, заданное в параметре **Фаска** – это длина линейных отрезков, пересекаемых на двух линиях, начиная от ребра, на котором требуется выполнить фаску
  - **Применить к острым углам:** при выборе активирует применение скругления только к ребрам, лежащим внутри прямого угла (<90°).

Основное преимущество использования операции STOOL:ВЫПОЛНИТЬ ФАСКУ НА ПРОФИЛЕ состоит в том, что генерированные профили адаптируются к изменениям исходных профилей, кроме того, можно работать в общей сложности на нескольких профилях, в т. ч. сложных.

Минимизировать профиль

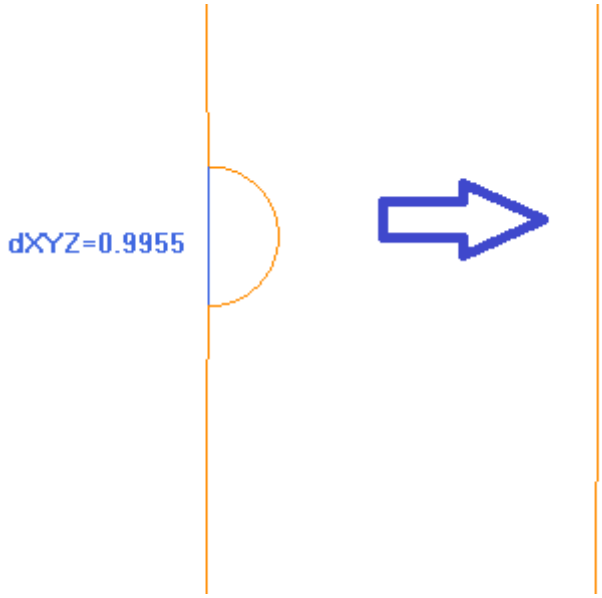
Команда **Минимизировать профиль**  находится в группе **Изменить профили** вкладки **Инструменты.** Этот инструмент позволяет уменьшить число образующих профиль отрезков. Уменьшение обеспечивается за счет объединения последовательных линейных и/или закругленных отрезков, для которых удовлетворяется условие геометрической непрерывности, с выбором из различных критериев.

- **Угол сокращения**: выражен в градусах, задает максимальный угловой конус, в пределах которого объединяются последовательные линейные отрезки. Заданное значение должно находиться в пределах от 0.0º до 90°: значение 0.0º соответствует неприменению никакого углового сокращения.
- **Максимизировать угловое сокращение**: при выборе требует итерации процедуры углового сокращения вплоть до обнуления применения.
- **Линейное сокращение**: при выборе требует сокращения, оценивающего длину отдельных отрезков. Выбор поля подключает применение приведенных ниже настроек.
- **Минимальная длина отрезков**: минимальная длина отрезков, оцененная как линейное расстояние

- **Применить к длине дуг**: при выборе применяет минимальную длину также к дугам, оценивая линейную длину дуги
- **Применить к хорде дуг**: при выборе применяет минимальную длину также к дугам, оценивая длину хорды дуги.

Этот инструмент применяет:
- предварительное сокращение профиля, объединяя линейные отрезки минимальной длины меньше эпсилон
- если требуется, угловое сокращение линейных отрезков, с возможной итерацией для повышения эффективности
- сокращение последовательных дуг, приводимых к одной разделенной дуге
- если требуется, сокращение оценивается на требуемом линейном расстоянии.

Очевидно, что чем больше требуется этапов, тем сложнее связанная с этим обработка, также с учетом размеров обрабатываемых профилей.
На рисунке показан пример линейного сокращения, примененного к профилю, генерированному с дефектами маршрута. Применяется очень сильный зум:



Слева отображается исходный маршрут, более светлый; очевидна вдающаяся внутрь зона, а синий отрезок выделяет порядок величины менее 1 мм;
справа отображается результат, получаемый с линейным сокращением, например, в 0.3 мм.

Выбор опции *Применить к копии операций* применяет этот инструмент к копии операций и не меняет исходные линии.
Этот инструмент доступен также как общий инструмент программы.

Фрагментировать профиль

Команда **Фрагментировать профиль**  находится в группе **Изменить профили** вкладки **Инструменты.** Инструмент разбивает отрезки профиля на несколько отрезков с выбором среди различных критериев фрагментации.

- **Максимальная длина отрезков:** максимальная длина отрезков, на которые разбит профиль. Минимальное значение положительно и равно 10.0*эпсилон разрешения координат.
- **Применить ошибку хорды к дугам:** при подключении выполняет фрагментацию дуг, задавая заданную Ошибку хорды (см. ниже). В этом случае значение **Максимальная длина отрезков** применяется только к фрагментации линейных отрезков.
- **Распределить оставшиеся:** при выборе на каждом отдельном исходном отрезке вычисляет целое число отрезков, на которые фрагментировать его, и на них распределяет оставшиеся.
  Например, если линейный отрезок имеет длину 52 мм и задается длина отрезков, равная 10 мм:
  - если этот пункт не выбран, линейный отрезок будет фрагментирован на 6 отрезков: 5 длиной по 10 мм и один (последний) длиной 2 мм.
  - если же этот пункт выбран, линейный отрезок будет фрагментирован на 6 отрезков, все одинаковой длины. Длина пересчитывается равной (52/6) =8.6666
- **Фрагментировать только дуги:** инструмент применяется только к закругленным отрезкам. Этот выбор не применяется при выборе пункта **Применить только к текущему положению**.

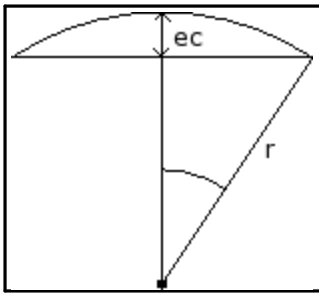
- **Линеаризировать дуги:** разбивает дуги на отрезки, которые преобразуются в линейные отрезки
- **Применить в трехмерном виде:** максимальная длина отрезков применяется также к компоненту глубины

Применить к: предусматриваются три опции:

- **Применить только к текущему положению:** разбивает только текущий отрезок
- **Применить от текущего положения до конца профиля:** разбивает от текущего отрезка до конца профиля
- **Применить к профилю в целом:**разбивает весь профиль
- **Разбить на число отрезков:** при выбранном пункте **Применить только к текущему положению** позволяет фрагментировать элемент на заданное число отрезков. В этом случае задание **Максимальная длина отрезков** игнорируется и вычисляется автоматически. Это поле принимает одно числовое значение от 2 до 99. Этот пункт активизирован, только если:
  - этот инструмент применяется к текущей операции
  - текущая операция - операция типа линии или дуги и выполняет один геометрический отрезок
- **Максимальный радиус фрагментирования:** позволяет фрагментировать только дуги с заданным максимальным радиусом. Это поле принимает положительное числовое значение. Например, выбрать клетку управления и задать значение 4.0: инструмент разбивает дуги, имеющие радиус, меньший или равный 4.0 мм. Этот выбор применяется только если не выбран пункт **Применить только к текущему положению.**

Выбор опции *Применить к копии операций* применяет этот инструмент к копии операций и не меняет исходные линии.
Если в подготовке исходного профиля заданы начальный и/или конечный отрезки, в подготовке они остаются присвоенными напрямую и не подлежат фрагментации.
Этот инструмент доступен также как общий инструмент программы.

Фрагментация дуги



На рисунке показано геометрическое значение, присвоенное заданному значению ошибки хорды (заданное значение ошибки хорды может быть 0.05 мм).
Фрагментация дуги по критерию ошибки хорды определяет выборки длины, зависящей от радиуса дуги. При увеличении радиуса увеличивается длина отрезков.
Фрагментация дуги по критерию ошибки хорды сохраняет одну точность выборки для всех дуг, т. к. связана с изгибом дуги.

Кроме того:

- для каждой фрагментации допускается максимальное значение ошибки хорды, равное 50% радиуса дуги
- для дуги, в любом случае, решается число выборок не менее целых долей 45° (амплитуды дуги)
- для каждой фрагментации допускаются точные пределы вычисляемого угла выборки. Минимальное значение составляет 1°, максимальное - 45°

PROFESSIONAL

Можно генерировать профили с фрагментацией и линеаризацией профиля также в форме сложной операции, вызывая операцию типа Запрограммированные инструменты в списке операций. В группе ИНСТРУМЕНТЫ выбирается операция STOOL: ФРАГМЕНТИРОВАТЬ И ЛИНЕАРИЗОВАТЬ:

- Поле **Операции** задает имена, заданные запрограммированным ранее операциям, соответствующим исходным профилям.

Профили могут быть получены также в результате применения других сложных кодов и выполнение операции соответствует только измененному профилю и не включает исходные профили. Операции, не используемые для целей требуемых функций (например, точечные или логические операции или сложные неразбиваемые операции), игнорируются.

В операции задаются:

- Типичные параметры сложной операции (см. информацию, приведенную об общем коде подпрограммы):
  - **Qx,Qy Zp**: начальные координаты размещения разработанных операций
  - **...**
  - **Свойства операции**: задает свойства, присвоенные операции

- Особые параметры функции операции со значением, аналогичным полям, определенным в окне инструмента:
  - **Максимальная длина отрезков:** максимальная длина отрезков, на которые разбит профиль.

- **Применить ошибку хорды к дугам:** при активизации выполняет фрагментацию дуг, присваивая Погрешность хорды, заданную в следующем поле. В этом случае значение **Максимальная длина отрезков** применяется только к фрагментации линейных отрезков.
- **Погрешность хорды:** задает ошибку хорды
- **Распределить оставшиеся:** при выборе на каждом отдельном исходном отрезке вычисляет целое число отрезков, на которые фрагментировать его, и на них распределяет оставшиеся.
- **Фрагментировать только дуги:** инструмент применяется только к закругленным отрезкам.
- **Линеаризировать дуги:** разбивает дуги на отрезки, которые преобразуются в линейные отрезки.
- **Применить в трехмерном виде:** максимальная длина отрезков применяется также к компоненту глубины

Основное преимущество использования операции STOOL: ФРАГМЕНТИРОВАТЬ и ЛИНЕАРИЗОВАТЬ состоит в том, что генерированные профили адаптируются к изменениям исходных профилей, кроме того, можно работать в общей сложности на нескольких профилях, в т. ч. сложных.

Линеаризировать по Z

Команда **Линеаризовать глубину**  находится в группе **Изменить профили** вкладки **Инструменты.** Инструмент линеаризирует изменение глубины профиля. Инструмент действует на простые профили только дугами на плоскости xy.

Выбрать из двух предложенных опций:
- **Читать изменение глубины из профиля**: считывает крайние координаты Z (начальную и конечную), как они заданы на профиле, и делает недоступными поля ввода данных
- **Задать изменение глубины**: задать непосредственно в полях начальную координату Z и конечную координату Z. Если конечная Z не задана (поле пустое), она принимает то же значение, что и начальная Z, обеспечивая равномерность глубины по всему профилю.

При двух координатах Z, как задано на рисунке: изменяет конечную глубину применения каждого отрезка профиля так, чтобы получить постепенное изменение глубины по всему профилю, от координаты Z=-5 до конечной координаты Z=-12.
Этот инструмент применяется напрямую только к текущему профилю.



Можно генерировать профили с линеаризацией глубины также в форме сложной операции, вызывая операцию типа Запрограммированные инструменты в списке операций. В группе ИНСТРУМЕНТЫ выбирается операция STOOL:ЛИНЕАРИЗОВАТЬ ПО Z:
- Поле **Операции** задает имена, заданные запрограммированным ранее операциям, соответствующим исходным профилям.

Профили могут быть получены также в результате применения других сложных кодов и выполнение операции соответствует только измененному профилю/профилям и не включает исходные профили. Операции, не используемые для целей требуемых функций (например, точечные или логические операции или сложные неразбиваемые операции), игнорируются.

Операция задает:
- Типичные параметры сложной операции (см. информацию, приведенную об общем коде подпрограммы):
  - **Qx,Qy Zp**: начальные координаты размещения разработанных операций
  - **...**
  - **Свойства операции**: задает свойства, присвоенные операции

- Это особые параметры функции операции со значением, аналогичным полям в окне инструмента:
  - **Задать глубину:** подключает задание крайних координат
  - **Z начальная, Z конечная:** задают начальную и конечную координату Z при выбранной опции **Задать глубину**.

Основное преимущество использования операции STOOL: ЛИНЕАРИЗОВАТЬ ПО Z состоит в том, что генерированные профили адаптируются к изменениям исходных профилей, кроме того, можно работать в общей сложности на нескольких профилях, в т. ч. сложных.

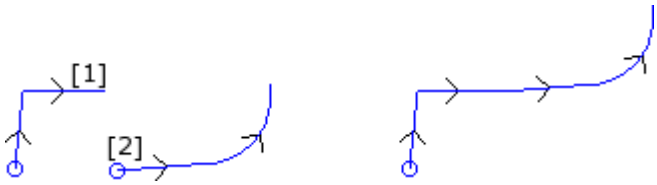
Объединение профилей

Переносом

Команда **Объединить профили переносом**  находится в группе **Изменить профили** вкладки **Инструменты.**

Этот инструмент объединяет два или более профиля, перенося их таким образом, чтобы начальная точка второго профиля совпала с конечной точкой первого профиля. Этот инструмент действует также на расширенные профили.
В грани-детали этот инструмент активизирован только при активизированном двухмерном или каркасном представлении и действует только на профили, примененные к текущей грани.
При выборе этой команды предоставляется запрос, применять ли этот инструмент к копии операций.

Отдельные профили определяются при их выборе в графической зоне мышью и нумеруются для выделения порядка, в котором выполняется объединение. Указания к выполнению приводятся в зоне "Команды" и можно выбрать максимум 99 профилей.
Для отмены последнего выбора нажать кнопку "Z". Недействительные или дублированные выборы сигнализируются сообщениями.
Завершить выбор, нажимая **[Enter]** для подтверждения применения инструмента или **[Escape]** для отмены выбора.



Слева представлена исходная ситуация с двумя отдельными профилями. Профиль **(1)** - это первый выбранный профиль, а профиль **(2)** - это второй выбранный профиль.
Справа представлена ситуация после применения инструмента. Профиль **(1)** остается в исходном положении, а профиль **(2)** переносится к конечной точке отрезка **(1)**. В результате получается единый профиль с подготовкой в начальной точке профиля **(1)**.

Соединительным отрезком

Команда **Соединить профили соединительным отрезком**  находится в группе **Изменить профили** вкладки **Инструменты.**
Объединяет профили без изменения их положения, вводя два соединительных линейных отрезка. Действительны положения, приведенные для предыдущего инструмента.



Слева представлена исходная ситуация с двумя отдельными профилями. Профиль **(1)** - это первый выбранный профиль, а профиль **(2)** - это второй выбранный профиль.
Справа представлена ситуация после применения инструмента. Профили остаются в своем исходном положении и между конечной точкой отрезка **(1)** и начальной точкой отрезка **(2)** вводится соединительный линейный отрезок. В результате получается единый профиль с подготовкой в начальной точке профиля **(1)**.

Соединить соседние профили

Объединяет профили, в которых конечная или начальная точка одного профиля совпадает с начальной или конечной точкой следующего профиля. Команда **Соединить соседние профили**  находится в группе **Изменить профили** вкладки **Инструменты.**

- **Расстояние соединения:** это максимальное допустимое расстояние между двумя профилями, определенное между конечной точкой первого и начальной точкой второго профиля, для подключения возможности выполнения автоматического соединения. Принимается значение от *эпсилон* системы до значения (100 * *эпсилон*). Если заданное здесь значение больше *эпсилон* системы, соединение перемещает второй профиль так, чтобы обеспечить геометрическую непрерывность с предыдущим.
- **Управлять инверсией профилей:** выбрать для подключения инверсии профилей, следующих за первым, для более лучшей оценки возможности объединения;
- **Применить в трехмерном виде:** выбрать для оценки расстояния между профилями также на составляющей глубины (ось Z).

Выбор опции *Применить к копии операций* применяет этот инструмент к копии операций и не меняет исходные линии.

Закрыть окно, нажимая OK для подтверждения настроек и продолжения работы с инструментом. Затем представляется запрос **указать первый профиль** указателем мыши.
После выполнения выбора представляется запрос, желаете ли вы **Активировать автоматический поиск соединений:**
- Если ответ положительный, выполняется соединение всех соседних профилей автоматически;
- если ответ отрицателен, продолжать работу, выбирая соединяемые профили мышью (максимум 99 профилей), как уже рассматривалось для двух предыдущих инструментов.



Можно генерировать профили соединением также в форме сложной операции, вызывая операцию типа *Запрограммированные инструменты* в списке операций. В группе ИНСТРУМЕНТЫ выбирается операция STOOL:СОЕДИНИТЬ ПРОФИЛИ:
- Поле **Операции** задает имена, заданные запрограммированным ранее операциям, соответствующим исходным профилям.

Профили могут быть получены также в результате применения сложных кодов и выполнение операции соответствует только измененному профилю/профилям и не включает исходные профили. Операции, не используемые для целей требуемых функций (например, точечные или логические операции или сложные неразбиваемые операции), игнорируются.

Операция задает:
- Типичные параметры сложной операции (см. информацию, приведенную об общем коде подпрограммы):
  - **Qx,Qy Zp**: начальные координаты размещения разработанных операций
  - **...**
  - **Свойства операции**: задает свойства, присвоенные операции
- Это особые параметры функции операции со значением, аналогичным полям в окне инструмента:
  - **Переместить профили**: выбрать для подключения перемещения профилей (см. инструмент: *Объединить профили переносом*). Если этот пункт не выбран: объединение выполняется путем ввода соединительных линейных отрезков (см. инструмент: *Соединить профили соединительными отрезками*).

Объединение профилей выполняется без изменения их порядка или изначального направления профилей. Основное преимущество использования операции STOOL:СОЕДИНИТЬ ПРОФИЛИ состоит в том, что вставленные профили адаптируются к изменениям исходных профилей.

Сместить подготовку на замкнутый профиль

Команда **Сместить подготовку на замкнутый профиль**  находится в группе **Изменить профили** вкладки **Инструменты.** Этот инструмент смещает подготовку текущего профиля в другую точку этого профиля. Профиль должен быть замкнут, поэтому начальная и конечная точки должны совпадать во всех координатах (x, y, z).

Координата X, Координата Y: задают новое положение подготовки. Щелкнуть значок  для выбора мышью в графической зоне. При определении положения точки мышью координаты задаются так, чтобы соответствовать точке вдоль профиля. При прямом задании или изменении координат осуществляется поиск точки вдоль профиля, ближе всего расположенной к заданной.

Профиль изменяется при сохранении без изменения направления обхода. Если исходный профиль не начинается с операции setup, новая точка setup задается с копией эталонной подготовки (как задано **Индивидуализировать->Технология->Коды по умолчанию** меню Применение).

Выбор опции *Применить к копии операций* применяет этот инструмент к копии операций и не меняет исходные линии.

Применить подготовку к профилю

Команда **Применить подготовку**  находится в группе **Изменить профили** вкладки **Инструменты.** Этот инструмент применяет технологическую подготовку к выбранным профилям (профилям, имеющим по меньшей мере один выбранный элемент) или к текущему профилю.
Режим применения инструмента рассматривается в параграфе **Порядок задания технологии профилю**.
Этот инструмент применяется непосредственно к исходным профилям или копии.

Применить многократную подготовку

Команда **Применить многократную подготовку**  находится в группе **Изменить профили** вкладки **Инструменты.** Применить многократную подготовку к выбранным профилям (профилям, имеющим как минимум один выбранный элемент) или к текущему профилю.
Режим применения инструмента рассматривается в параграфе **Порядок задания технологии профилю**.
Этот инструмент применяется непосредственно к исходным профилям или копии.

10.4 Построения

Исправленный профиль

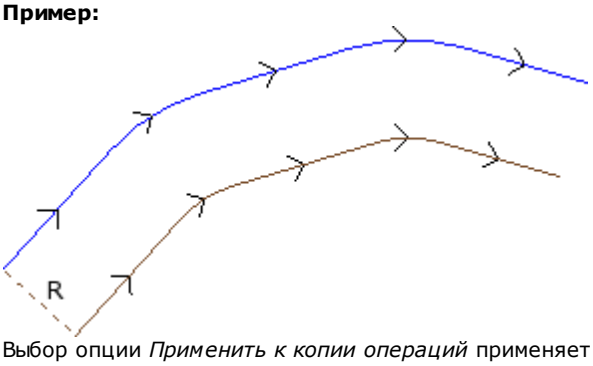
Вставляет новый профиль, полученный для коррекции на инструмент из текущего профиля. Команда **Исправленный профиль**  находится в группе **Построения** вкладки **Инструменты.** Этот инструмент подключен, если текущая операция относится к профилю.
Эта команда может быть доступна также для функции *Essential*.

- **Радиус коррекции:** значение коррекции
- **Разрешить сокращение профиля:** активизирует удаление отрезков в исправленном профиле по сравнению с изначальным с учетом геометрических условий габарита, превышающих саму коррекцию.

На внешней коррекции ребра:

- **Вставить скругления:** выбрать для вставки скругления;
- **Сократить скругления до пересечений:** выбрать для решения ребра.
- **Сторона коррекции:** выбирает сторону коррекции (левую или правую)

Пример:



Исходный профиль наружный. Исправленный профиль внутренний, т. е. требуемая **Сторона коррекции** - **Правая**.
R - радиус коррекции.
Если исходный профиль имеет в подготовке заданный начальный и/или конечный отрезок, он остается заданным также в настройке исправленного профиля.

Выбор опции *Применить к копии операций* применяет этот инструмент к копии операций и не меняет исходные линии.
Исправленный профиль генерируется с сокращением профиля до элементарных кодов и только с цифровыми заданиями (т. е. определяет в цифровом формате любой параметр, используемый при задании исходного профиля).
ВНИМАНИЕ: элементы разметки (L24) разбиваются на микроотрезки, задающие кривую.



Можно генерировать исправленные профили также в форме сложной операции, вызывая операцию типа Запрограммированные инструменты в списке операций. В группе ИНСТРУМЕНТЫ выбирается операция STOOL:ИСПРАВЛЕННЫЙ ПРОФИЛЬ:

- Поле **Операции** задает имена, заданные запрограммированным ранее операциям, соответствующим исходным профилям.

Профили могут быть получены также в результате применения сложных кодов и выполнение операции соответствует только измененному профилю/профилям и не включает исходные профили. Операции, не используемые для целей требуемых функций (например, точечные или логические операции или сложные неразбиваемые операции), игнорируются.

Операция задает:

- Типичные параметры сложной операции (см. информацию, приведенную об общем коде подпрограммы):
  - **Qx,Qy Zp**: начальные координаты размещения разработанных операций.
  - ...
  - **Свойства операции**: задает свойства, присвоенные операции.

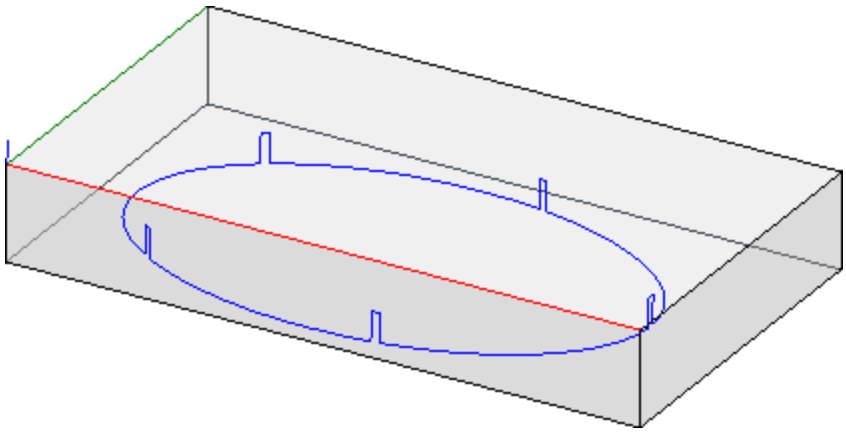
- Это особые параметры функции операции со значением, аналогичным полям в окне инструмента:
  - **Радиус коррекции:** значение коррекции.
  - **Коррекция:** выбирает сторону коррекции (левую или правую).
  - **Контурирование**: выбирает режим коррекции на ребрах. В списке имеется два пункта:
    - **Скругления** активизирует коррекцию с введением скруглений.
    - **Ребра** активизирует коррекцию на поиске пересечений.
  - **Сократить профиль:** активизирует удаление отрезков в исправленном профиле по сравнению с изначальным с учетом геометрических условий габарита, превышающих саму коррекцию

Основное преимущество использования операции STOOL: ИСПРАВЛЕННЫЙ ПРОФИЛЬ состоит в том, что вставленные профили адаптируются к изменениям исходных профилей, кроме того, можно работать в общей сложности на нескольких профилях.

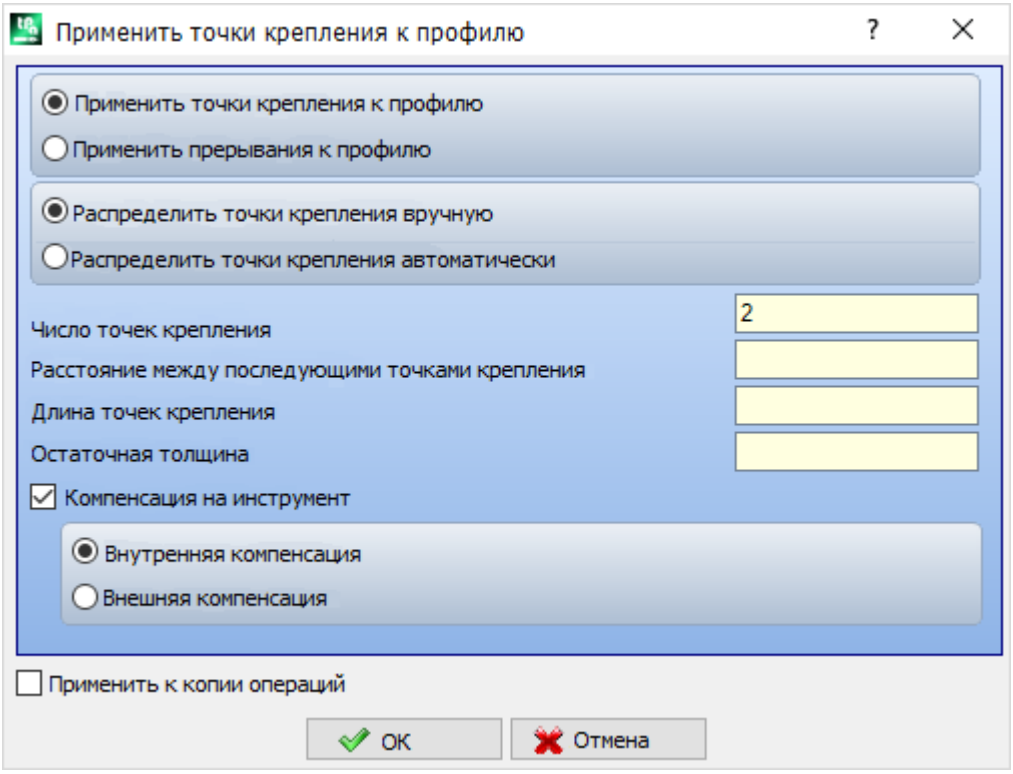
Применить точки крепления к профилю

Команда **Применить точки крепления к профилю**  находится в группе **Построения** вкладки **Инструменты.** Этот инструмент имеет двойную функцию с возможностью применения точек крепления или прерываний.
Эта команда может быть доступна также для функции *Essential*.

Точка крепления состоит в фрагментации изначального отрезка профиля на коротком отрезке, выполненном на такой глубине, чтобы оставить остаточную толщину в детали.
Прерывание состоит в фрагментации исходного отрезка профиля на короткий отрезок, который затем удаляется, прерывая исходный профиль на несколько профилей.
Этот инструмент подключен, если текущая операция относится к профилю.
При применении точек крепления этот инструмент обычно используется при наличии замкнутых профилей с глубиной фрезерования, превышающей толщину детали (сквозной профиль). В этих случаях прямое выполнение профиля вызвало бы отсоединение с опасностью падения части детали (внутри или снаружи фрезерования) при обработке детали.
Ввод точек крепления позволяет оставить точки крепления в детали, не допуская отсоединения ее части.
На рисунке показан пример сквозного замкнутого эллипса с примененными 5 точками крепления.



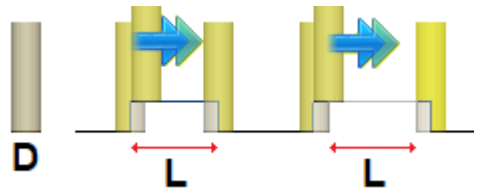
При применении прерываний обычно используется инструмент с технологиями резки, отличными от традиционного инструмента: плазменной, лазерной резки и пр.
Распределение точек крепления (или прерываний) на профиле может выполняться следующими способами:
- **Распределить точки крепления вручную:** мышью выбираются точки на профиле для ввода точки крепления. Процедура считывания позиций запускается после закрытия окна с подтверждением.
- **Распределить точки крепления автоматически:** точки крепления, в указанном количестве, распределяются по профилю наиболее равномерно.



- **Число точек крепления**: задает число точек крепления для автоматического распределения. Это поле принимает одно целое числовое значение от 1 до 255.
- **Расстояние между креплениями**: задает линейное расстояние между последовательными креплениями и значимо, если больше (эпсилон*10.0).
  Эта настройка альтернативна **Числу точек крепления**, если заданное число креплений меньше 1. При **Числе точек крепления** больше 1 дополняет их использование: заданное здесь расстояние можно переоценивать так, чтобы распределить по меньшей мере требуемое число точек крепления. В любом случае, минимальное число распределенных креплений - 1.
- **Длина креплений**: задает длину крепления (в плоскости xy грани). Это значение не может быть меньше эпсилон разрешения, заданного в конфигурации приложения производителем станка.
- **Остаточная толщина**: задает толщину, которую инструмент оставляет в детали при выполнении крепления. Принимает положительные значения, равные как минимум эпсилон разрешения (задается в конфигурации приложения производителем станка). Этот параметр не значим при применении прерываний.
- **Компенсация на инструмент**: выбрать, если требуется изменить фактическую длину крепления с учетом габарита инструмента. При активизированном выборе крепление генерируется более узким или широким, позволяя инструменту выполнить его нужной длины. Критерий изменения точки крепления определяется следующим выбором:
  - **Внутренняя компенсация**: крепление создается уже диаметра инструмента
  - **Наружная компенсация**: крепление создается шире диаметра инструмента.

При внутренней компенсации заданная для всех креплений длина должна быть как минимум равна диаметру инструмента

На рисунке показан типичный случай **наружной** компенсации:

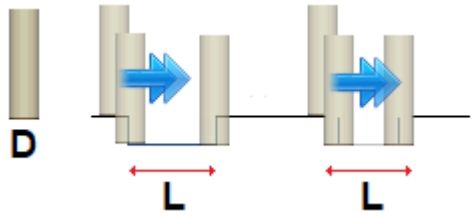


- **L** - Запрограммированная **Длина креплений**
- **D** - Диаметр инструмента

Слева выполняется крепление выполненное без Компенсации на инструмент и справа показано то же самое крепление при Наружной компенсацией:

- без компенсации реальная длина крепления снижается на значение **D**
- при компенсацией реальная длина крепления - **L**.

На рисунке показан типичный случай **внутренней** компенсации:



Слева выполняется крепление выполненное без Компенсации на инструмент и справа показано то же самое крепление при внутренней Компенсацией:

- без компенсации реальная длина крепления увеличивается на значение **D**
- при компенсацией реальная длина крепления - **L**.

В представленном на рисунке примере эллипса предположим:
- толщина детали 65 мм;
- запрограммированная глубина для эллипса: -70 мм
- диаметр фрезы: 9 мм.

В этом окне зададим:
- число точек крепления: 5
- длина точек крепления: 20 мм
- толщина остатков: 2 мм

По эллипсу распределяется 5 точек крепления, выполненные со следующими параметрами:
обратный подъем от Z=-70 мм до Z=-(65-2)=-63 мм
выполнение дуги эллипса длиной (20+/-9)=29/11 мм
опускание до Z=-70 мм, с продолжением запрограммированной траектории до следующей точки крепления.

Если в подготовке исходного профиля заданы начальный и/или конечный отрезки, в подготовке они остаются присвоенными напрямую.
Выбор опции *Применить к копии операций* применяет этот инструмент к копии операций и не меняет исходные линии.
Профиль с точками крепления генерируется только с цифровыми заданиями: определяет в числовом формате любой параметр, использованный при задании исходного профиля.

PROFESSIONAL

Можно генерировать профили, измененные с применением креплений также в форме сложной операции, вызывая операцию типа *Запрограммированные инструменты* в списке операций. В группе ИНСТРУМЕНТЫ выбирается операция STOOL:ПРИМЕНИТЬ ТОЧКИ КРЕПЛЕНИЯ:
- Поле **Операции** задает имена, заданные запрограммированным ранее операциям, соответствующим исходным профилям.

Профили могут быть получены также в результате применения сложных кодов и выполнение операции STOOL: ПРИМЕНИТЬ ТОЧКИ КРЕПЛЕНИЯ относится только к созданному скорректированному профилю/профилям и не включает исходные профили. Операции, не используемые для целей требуемых функций (например, точечные или логические операции или сложные неразбиваемые операции), игнорируются.

Операция задает:
- Типичные параметры сложной операции (см. информацию, приведенную об общем коде подпрограммы):
  - **Qx,Qy Zp**: начальные координаты размещения разработанных операций
  - **..**
  - **Свойства операции**: задает свойства, присвоенные операции

- Это особые параметры функции операции со значением, аналогичным полям в окне инструмента:
  - **Применить прерывания:** выбрать для применения прерывания
  - **Число точек крепления**: число распределяемых точек крепления. Поле принимает значения от 1 до 255.
  - **Расстояние между креплениями**: задает линейное расстояние между последовательными креплениями и значимо, если больше (эпсилон*10.0). Эта настройка альтернативна **Числу точек крепления**, если заданное число креплений меньше 1, или **Число точек крепления** больше 1 дополняет их использование: заданное здесь расстояние можно переоценивать так, чтобы распределить по меньшей мере требуемое число креплений. В любом случае, минимальное число распределенных креплений - 1.
  - **Длина креплений**: длина крепления (в плоскости xy грани)
  - **Толщина остатков:** толщина, которую инструмент оставляет в детали при выполнении точки крепления.

- **Компенсация на инструмент**: при выборе меняет фактическую длину крепления так, чтобы учитывать габарит инструмента. Выбор осуществляется из 3 пунктов: Не требуется; Внутренняя; Внешняя.

Фактическое число точек крепления, распределенное вдоль каждого профиля, зависит от общей длины профиля (общая длина и ее фрагментация) и поэтому может быть меньше заданного значения.
Основное преимущество использования операции STOOL: ПРИМЕНИТЬ ТОЧКИ КРЕПЛЕНИЯ состоит в том, что исправленные профили адаптируются к изменениям исходных профилей, кроме того, можно работать в общей сложности на нескольких профилях.

Применить подачу глубины

Изменяет текущий профиль с вводом последующих проходов до достижения конечной заданной глубины.

Команда **Применить подачу по Z**  находится в группе **Построения** вкладки **Инструменты.**
Эта команда может быть доступна также для функции *Essential*.

Этот инструмент подключен, если текущая операция относится к профилю, и работает только на простых профилях. Обычно используется в профилях, которые должны выполняться на координате глубины, не доступной за один проход.

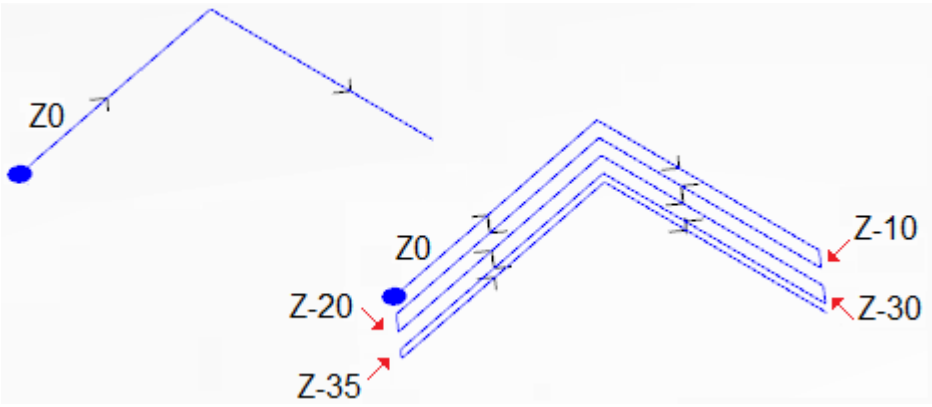
- **Z начальная**: содержит исходную глубину, считываемую профилем. Это поле не может быть изменено.
- **Z конечная**: воспроизводит глубину, считанную профилем на последнем запрограммированном отрезке (конечная глубина). Это поле не может быть изменено.
- **Координата Z**: задает конечную глубину, требуемую в применении рекурсивной развертки профиля. Это координата, на которую необходимо выйти исходя из заданного ниже шага. Это значение должно находиться за пределами интервала от **Z начальная** до **Z конечная**
- **Шаг по Z**: шаг подачи глубины, применимый при каждой развертке. Задание значимо без знака: процедура применяет необходимую подачу (+/-) для достижения конечной глубины. Если заданное значение равно нулю или превышает максимально допустимое, автоматически определяется значение, позволяющее достигнуть требуемую **Координату Z** одним добавленным выполнением.
- **Управлять инверсией профилей**: эта опция активирована только если эта команда применяется к замкнутому профилю. При выборе выполняет инверсию исполнения профиля при каждом изменении глубины. Если этот пункт не выбран, профиль всегда выполняется в изначальном направлении. Если профиль не замкнут, выполнение обязательно инвертируется при каждой смене глубины.
- 

Разрешить изменение стороны коррекции: при подключении при каждом дополнительном проходе задается инверсия стороны коррекции (справа налево или наоборот). Подключение незначимо в следующих случаях:
  - профиль замкнут, где не требуется инверсия профиля
  - профиль без запроса коррекции на инструмент.

  Управление этой опцией возможно только при ее активизации на этапе конфигурации TpaCAD производителем станка. Эта опция доступна только в режиме **Профессиональный**.

Если исходный профиль не имеет изменений глубины, заданная **Координата Z** также является глубиной, достигнутой на всей развертке профиля. Если же исходный профиль имеет изменения глубины, контроль достижения **Координаты Z** выполняется на развертке каждой подачи, вычисленной как амплитуда между **Z начальной** и **Z конечной**.
Значение последнего шага подачи может корректироваться в нижнюю сторону для обеспечения того, чтобы оно не превышало заданную **Координату Z** при выполнении между **Z начальная** и **Z конечная.**
При применении инструмента не учитываются изменения запрограммированных промежуточных глубин профиля.

Если в подготовке исходного профиля заданы начальный и/или конечный отрезки, в подготовке они остаются присвоенными напрямую и не учитываются при применении инструмента.
На дополнительных проходах обнуляются все изменения коррекции на инструмент на маршруте (прерывания, приостановки и возобновления и/или изменения стороны).
На рисунке представлен пример применения инструмента:



Слева представлен исходный профиль, заданный двумя линейными отрезками, выполненными на координате глубины Z=0.0.
Справа - профиль после изменения инструментом при применении заданий:

- Координата Z=-35.0
- Шаг Z=10.0

Исходный профиль не замкнут, поэтому каждое повторение инвертирует исполнение.
На рисунке показаны различные положения глубины, как они вычислены для каждого из проходов: последний проход достигает Координаты Z, заданной при уменьшении шага подачи с 10.0 до 5.0.
В примере максимальное значение **Шага Z** равно 35.0.

PROFESSIONAL

Можно генерировать профили, измененные с подачей глубины, также в форме сложной операции, вызывая операцию типа *Запрограммированные инструменты* в списке операций. В группе ИНСТРУМЕНТЫ выбирается операция STOOL:ПОДАЧА ПО Z:

- Поле **Операции** задает имена, заданные запрограммированным ранее операциям, соответствующим исходным профилям.

Профили могут быть получены также в результате применения других сложных кодов и выполнение операции STOOL: ПОДАЧА ПО Z относится только к измененным профилям и не включает исходные профили. Операции, не используемые для целей требуемых функций (например, точечные или логические операции или сложные неразбиваемые операции), игнорируются.
Дополнительно к инструменту эта операция позволяет задавать ось развертки на одной из трех осей координат грани.

Операция задает:

- Типичные параметры сложной операции (см. информацию, приведенную об общем коде подпрограммы):
  - **Qx,Qy Zp**: начальные координаты размещения разработанных операций
  - **..**
  - **Свойства операции**: задает свойства, присвоенные операции

- Это особые параметры функции операции со значением, аналогичным полям в окне инструмента:
  - **Ось развертки**: выбирает ось развертки на одной из трех осей координат грани (Z, X, Y)
  - **Присвоить число проходов**: при выборе эта опция запрашивает задание числа проходов. В противном случае запрашивает задание конечной глубины, на которой выполняется последняя подача.
  - **Конечная координата**: задает требуемую конечную глубину по оси развертки. Это задание игнорируется, если выбрана предыдущая опция;
  - **Число проходов**: требуемое число проходов (действительное задание: 1-1000). Задание значимо, если выбрана опция *Присвоить число проходов*;
  - **Шаг по Z**: шаг подачи глубины, применимый при каждой развертке. При задании конечной глубины задание значимо без знака: процедура применяет необходимую подачу (+/-) для достижения конечной глубины.
    Это задание значимо со знаком при задании числа проходов.
  - **Управлять инверсией профилей**: управляет инверсией выполнения (см. инструмент).
  - **Изменить сторону коррекции:** управляет сменой стороны коррекции на инструмент (см. инструмент). Задание значимо при развертке по оси Z.

Основное преимущество использования операции STOOL: ПОДАЧА ПО Z состоит в том, что исправленные профили адаптируются к изменениям исходных профилей, кроме того, можно работать в общей сложности на нескольких профилях, в т. ч. сложных.

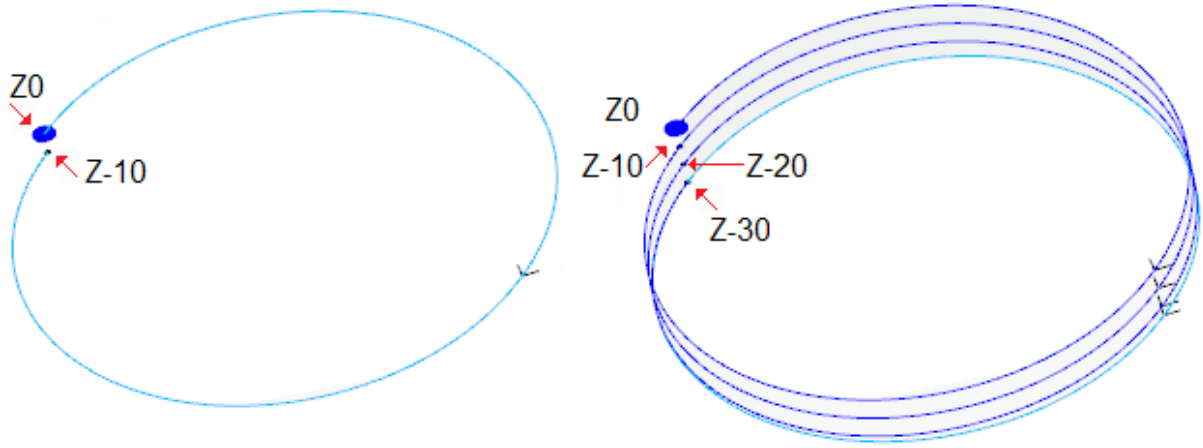
Применить повторение профиля

Изменяет текущий профиль, вводя последующие повторения до достижения или приближения к заданной конечной глубине с возможным выполнением заключительного прохода на постоянной глубине. Команда **Повторение профиля** находится в группе **Построения** вкладки **Инструменты.**
Эта команда может быть доступна также для функции *Essential*.
Этот инструмент подключен, если текущая операция относится к профилю, и работает только на простых профилях.
Кроме того, профиль должен замыкаться на плоскости XY и выполнять изменение глубины между его началом и концом.
Основное отличие от инструмента **Применить подачу по Z** состоит в том, что повторения профиля определяются не введением вертикальных отрезков, а самой структурой исходного профиля.

- **Z начальная**: содержит исходную глубину, считываемую профилем. Это поле не может быть изменено.
- **Z конечная**: воспроизводит глубину, считанную профилем на последнем запрограммированном отрезке (конечная глубина). Это поле не может быть изменено.
- **Координата Z**: задает конечную глубину, требуемую в применении повторений профиля. Это предельное значение, которое не может быть достигнуто, если это не позволяет изменение глубины исходного профиля. Это значение должно находиться за пределами интервала от **Z начальная** до **Z конечная**.
- **Завершает проходом при постоянном Z**: эта опция значима, если профиль не выполняет дуги на плоскости, отличной от xy. При выборе завершает изменение профиля добавлением прохода на постоянной глубине. Если значение **Координата Z** равно **Z конечная**: инструмент меняет профиль, добавляя только проход с постоянной глубиной.

При применении инструмента не учитываются изменения запрограммированных промежуточных глубин профиля.

Если в подготовке исходного профиля заданы начальный и/или конечный отрезки, в подготовке они остаются присвоенными напрямую и не учитываются при применении инструмента.
На дополнительных проходах обнуляются все изменения коррекции на инструмент на маршруте (прерывания, приостановки и возобновления и/или изменения стороны).
На рисунке представлен пример типового применения инструмента:



Слева представлен исходный профиль, заданный кругом, выполненным на глубине Z от 0.0 до -10.0.
Справа - профиль, как он был изменен инструментом при применении заданий:
- Координата Z=-30.0

На рисунке показаны различные положения глубины как они вычислены для каждого из повторения:
- первое добавленное повторение начинается на Z=-10.0 и завершается на Z=-20.0
- второе добавленное повторение начинается на Z=-20.0 и завершается на Z=-30.0
- на рисунке добавлено также последнее повторение, выполненное с постоянной глубиной Z=-30.0.

PROFESSIONAL

Можно генерировать профили, измененные повторением, также в форме сложной операции, вызывая операцию типа **Запрограммированные инструменты** в списке операций. В группе ИНСТРУМЕНТЫ выбирается операция STOOL: ПОВТОРЕНИЕ ПРОФИЛЯ:

- Поле **Операции** задает имена, заданные запрограммированным ранее операциям, соответствующим исходным профилям.

Профили могут быть получены также в результате применения других сложных кодов и выполнение операции STOOL: ПОВТОРЕНИЕ ПРОФИЛЯ относится только к измененным профилям и не включает исходные профили. Операции, не используемые для целей требуемых функций (например, точечные или логические операции или сложные неразбиваемые операции), игнорируются.
Дополнительно к инструменту эта операция позволяет задавать ось развертки на одной из трех осей координат грани.

Операция задает:

- Типичные параметры сложной операции (см. информацию, приведенную об общем коде подпрограммы):
  - **Qx,Qy Zp**: начальные координаты размещения разработанных операций
  - **..**
  - **Свойства операции**: задает свойства, присвоенные операции

- Особые параметры функции операции со значением, аналогичным полям, определенным в окне инструмента:
  - **Ось развертки**: выбирает ось развертки на одной из трех осей координат грани (Z, X, Y)
  - **Присвоить число проходов**: при выборе эта опция запрашивает задание числа проходов. В противном случае запрашивает задание конечной глубины, на которой выполняется последняя подача.
  - **Конечная координата**: задает требуемую конечную глубину по оси развертки. Это задание игнорируется, если выбрана предыдущая опция
  - **Число проходов**: требуемое число проходов (действительное задание: 0-1000). Задание значимо, если выбрана опция **Присвоить число проходов**
  - **Завершает проходом при постоянном Z**: при выборе завершает изменение профиля добавлением прохода на постоянной глубине. При задании **Числа проходов** равным нулю или если **Конечная координата** не отличается от координаты конечной развертки исходного профиля: к профилю добавляется только проход с постоянной глубиной.

Выполнение заключительного прохода, в любом случае, зависит от проверки на исходном профиле в зависимости от выбранной **Оси развертки**:

- если ось Z: не должно создавать дуги на плоскости, отличной от xy
- если ось X: не должно создавать дуги на плоскости, отличной от yz
- если ось Y: не должно создавать дуги на плоскости, отличной от xz.

Применение запрограммированного инструмента с изменением исходного профиля определяется проверкой на исходном профиле в зависимости от выбранной **Оси развертки**:

- если ось Z: профиль должен замыкаться на плоскости XY и создавать изменение по Z между началом и концом
- если ось X: профиль должен замыкаться на плоскости YZ и создавать изменение по X между началом и концом
- если ось Y: профиль должен замыкаться на плоскости XZ и создавать изменение по Y между началом и концом.

Основное преимущество использования операции STOOL: ПОВТОРЕНИЕ ПРОФИЛЯ состоит в том, что исправленные профили адаптируются к изменениям исходных профилей, кроме того, можно работать в общей сложности на нескольких профилях, в т. ч. сложных.

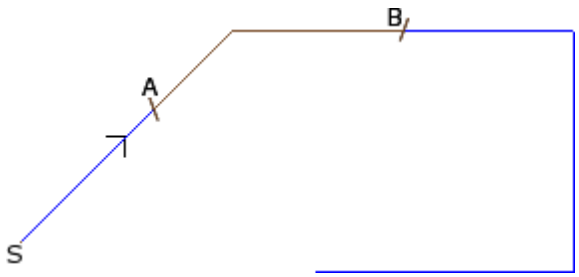
Повторить профиль

Повторяет часть текущего профиля. Команда **Повторить профиль**  находится в группе **Построения** вкладки **Инструменты.** Этот инструмент подключен, если текущая операция относится к профилю.

Повторяемая часть профиля определяется между двумя точками резки.

Профиль получается как геометрический повтор отрезков, находящихся между двумя точками резки, и открывается с копией исходной или эталонной подготовка (как задано в **Индивидуализировать->Технология->Технология по умолчанию** меню **Применение**).

Координаты точек резки могут вводится в поля редактирования или мышью в графической зоне (при щелчке на значке ).
Выбор опции *Применить к копии операций* применяет этот инструмент к копии операций и не меняет исходные линии.
На рисунке представлен пример профиля:



- (S) означает точку начала профиля;
- стрелка указывает направление перемещения против часовой стрелки
- профиль не замкнут.

На профиле указаны 2 точки резки (A) и (B). Эти две точки могут относиться к одному или разным отрезкам.
Часть профиля между двумя точками, всегда в изначальном направлении перемещения, извлекается из профиля.
Если инструмент вызывается для работы на текущем профиле, удаляются следующие части профиля:
- от (S) до точки (A);
- от (B) до конца профиля.

Если инструмент вызывается для работы на копии текущего профиля, добавляется новый профиль, заданный от точки (A) до точки (B).
Исправленный профиль генерируется с сокращением профиля до элементарных кодов и только с цифровыми заданиями, определяет в цифровом формате любой параметр, используемый при задании исходного профиля.

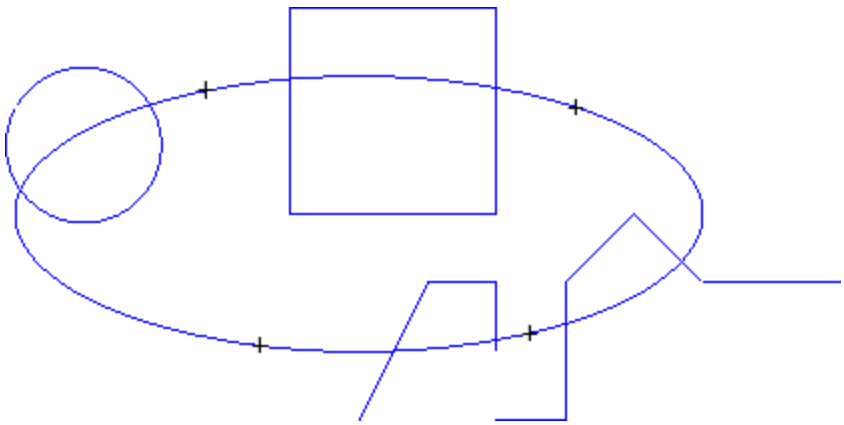
Разрезать профили

PROFESSIONAL

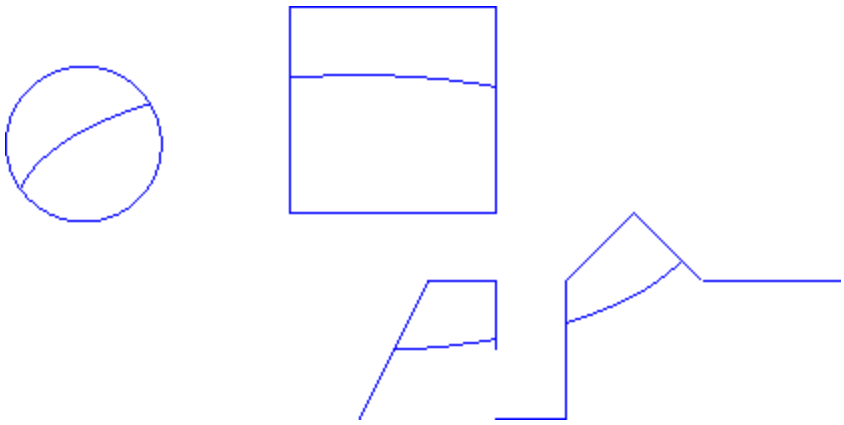
Этот инструмент позволяет резать части профиля в определенных точках *ребро резки,* найденных на пересечениях профилей.

Команда **Разрезать профили**  находится в группе **Построения** вкладки **Инструменты.**
В случае грани-детали этот инструмент активизирован только при активизированном двухмерном или каркасном представлении грани и действует только на профили, примененные к грани на текущем виде.

Определяющие ребра резки профили - это выбранные профили (при их наличии) или все отображенные профили грани.
Ребра резки указываются мышью по указаниям, приведенным в зоне Команд.
ВНИМАНИЕ: дуги не на плоскости XY и элементы разметки (L24) исключаются из оценки точек пересечения профилей.
На рисунке показана программа, состоящая из пересекающихся профилей. 4 крестика определяют ребра резки



После инструмента программа имеет вид, показанный на рисунке



Этот инструмент всегда применяется к исходным операциям программы

Построение профиля

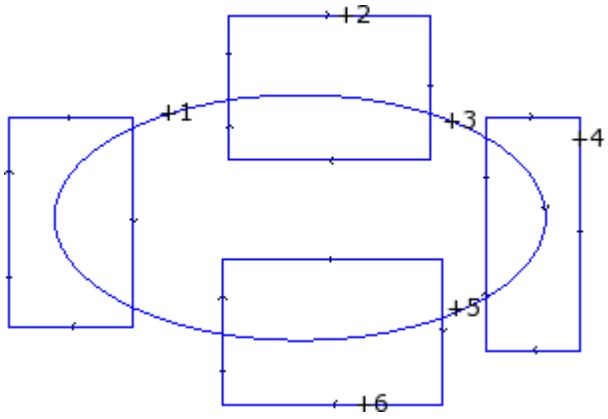
PROFESSIONAL

Этот инструмент позволяет построить новый профиль, выбирая один или более отрезков запрограммированных профилей. Выбранный отрезок должен иметь точку пересечения с отрезком предыдущего профиля. Команда **Построение профиля** находится в группе **Построения** вкладки **Инструменты.**

В случае грани-детали этот инструмент активизирован только при активизированном двухмерном или каркасном представлении грани и действует только на профили, примененные к грани на текущем виде. Выбор этой команды требует задания технологических данных нового профиля: операции setup и соответствующих технологических заданий. Образующие новый профиль отрезки указываются мышью по указаниям, приведенным в зоне команд.

ВНИМАНИЕ: дуги не на плоскости XY и элементы разметки (L24) исключаются из оценки точек пересечения профилей.

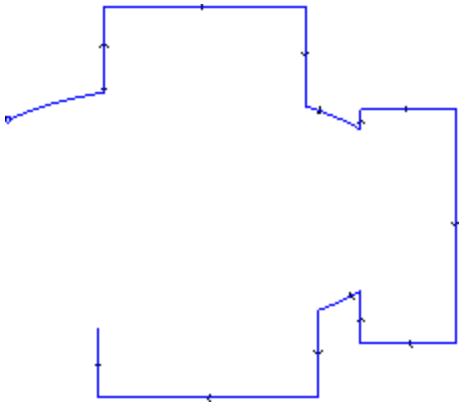
Пример:



На рисунке показана программа, состоящая из пересекающихся профилей. 6 крестиков определяют части, указанные для построения нового профиля. Крестики пронумерованы и указывают порядок, в котором выбранные части добавляются в новом профиле:

- крестик 1 определяет исходную точку. Отыскивается отрезок профиля, ближе всего расположенный к положению, в котором был сделан щелчок мышью
- крестик 2 выбирает, как продолжать профиль. Отыскивается отрезок профиля, ближе всего расположенный к положению, в котором был сделан щелчок мышью, и геометрически продолжающий отрезок, уже выбранный в качестве отрезка (1). Решение геометрической непрерывности может определить также инверсию отрезка (1) и/или отрезка (2) относительно направления исполнения исходных профилей
- крестик 3 выбирает, как продолжать профиль. Отыскивается отрезок профиля, ближе всего расположенный к положению, в котором был сделан щелчок мышью, и геометрически продолжающий отрезок, уже выбранный в качестве отрезка (2). Теперь решение геометрической непрерывности может определить только инверсию отрезка (3) относительно направления исполнения исходных профилей
- ..
- до крестика 6.

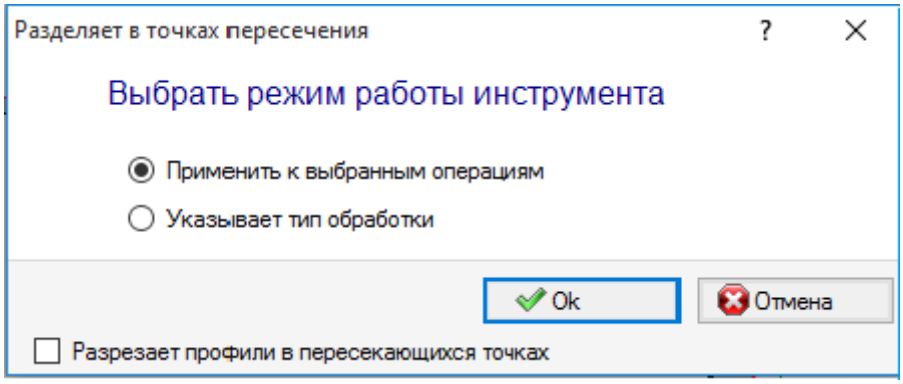
По завершении выбора эта команда подтверждается кнопкой **[Enter]** или выбором из локального меню (открывается правой кнопкой мыши). Теперь полученные позиции обрабатываются и к программе грани добавляется новый профиль без изменения исходных профилей.
На рисунке ниже представлен профиль, построенный по приведенным выше указаниям:



Разделить на точках пересечения

PROFESSIONAL

Этот инструмент позволяет выявлять точки пересечения между профилями и делить отдельные отрезки исходя из этих точек. Команда **Разделить на точках пересечения**  находится в группе **Построения** вкладки **Инструменты.**
В случае грани-детали инструмент действует только на профили, примененные на грани на текущем виде. При наличии выбранных профилей предлагается следующее окно:



Выбрать режим работы из двух предлагаемых:
- **Применить к выбранным операциям**: инструмент применяется к выборам
- **Указать операции**: при закрытии окна с подтверждением необходимо указать их интерактивно мышью по указаниям, приведенным в зоне Команды.

- **Разрезать профили по точкам пересечения**: выбрать для отделения каждого отдельного отрезка на определенные профили. В противном случае: отрезок профиля, на котором оказывается точка резки, делится на два, но профиль остается единым.

Если нет выбранных профилей, предлагается окно выбора опции резки и автоматически подключается выбор **Указать операции.**

Инструмент всегда применяется к исходным операциям программы и может работать также только на одном профиле.

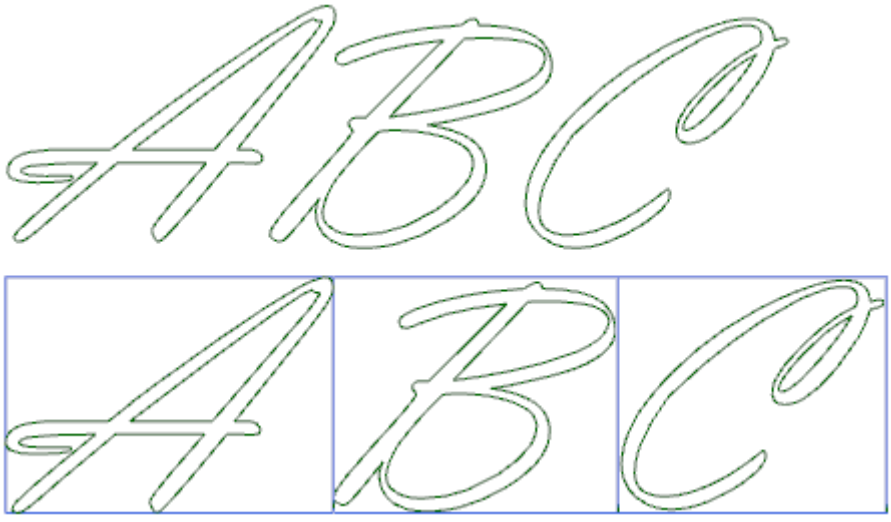
Создание текстов

PROFESSIONAL

Этот инструмент позволяет вставлять текст в программу грани напрямую, как профили. Команда **Создать текст** находится в группе **Построения** вкладки **Инструменты.**
Открытие этого окна может потребовать несколько секунд, необходимых для поиска и создания списка доступных шрифтов.

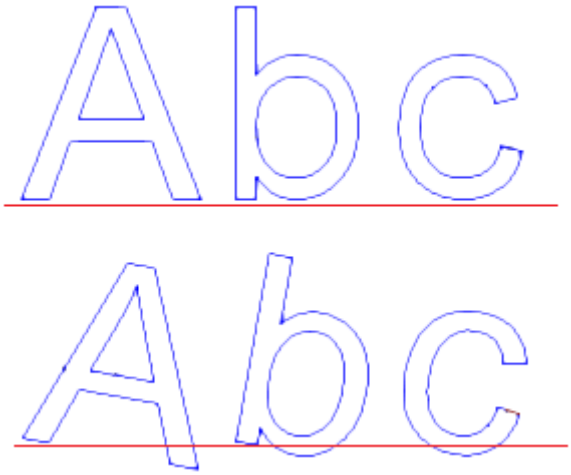
- **Текст**: вводимая надпись
- **Шрифт**: тип шрифта. Этот список обеспечивает доступность всех инсталлированных символов, для которых можно выбрать как минимум один из различных стилей (нормальный, курсив, полужирный, полужирный курсив).
- **Стиль**:стили, доступные для форматирования текста (нормальный, курсив, полужирный, полужирный курсив)
- **Высота прописных букв**: задает высоту буквы А в единицах измерения детали
- **Ширина интервала**: задает ширину пробелов при их наличии в тексте.
  - выбрать ячейку для того, чтобы поле стало редактируемым. Заданное здесь значение присваивается как ширина символа "пробел" с возможностью также обнуления;
  - если эта ячейка не выбрана, ширина, используемая для символа "пробел", является шириной, определенной для выбранного шрифта (соответствует ширине символа (–))
- **Расстояние между символами**: задает расстояние между отдельными символами надписи. Это задание используется, если не выбрана опция **Автоматическое распределение**, предусмотренная на второй странице. Перечень содержит два пункта для выбора режима, принимаемого для задания расстояния между двумя последующими символами в перечне:
  - **Метрическое**: расстояние определяется правилами, определенными для каждого отдельного символа шрифта
  - **Геометрическое**: расстояние определяется габаритным прямоугольником каждого отдельного символа.

На рисунке представлен пример надписи, созданной для двух случаев расстояния с заданным значением расстояния между символами =0.0:



Надпись, созданная в верхней части, использует *метрическое* расстояние. Надпись, созданная в нижней части, использует геометрическое расстояние: вокруг каждого символа представлен соответствующий габаритный прямоугольник.

- **Угол наклона**: задает наклон каждого символа относительно базовой линии развертки надписи. Это поле задается в градусах (°) и десятых долях градуса; значение по умолчанию - 0. Положительное значение обуславливает наклон символов вправо, соответствующий курсиву. На рисунке ниже одна и та же надпись выполнена с разным наклоном (0.0 и 10.0):

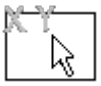


- **RightToLeft**: предусматривается для языков, в которых написание производится справа налево, как, например, в арабском языке или иврите. Эта опция инвертирует порядок символов надписи. При применении этой опции к примеру на рисунке была бы получена надпись "CBA".

При щелчке на значке  для подключения вкладки распределения текста и технологических заданий.

- **Выравнивание:** предусматриваются 4 альтернативы
  - **Выровнять влево**: выбор по умолчанию, может использоваться всегда.

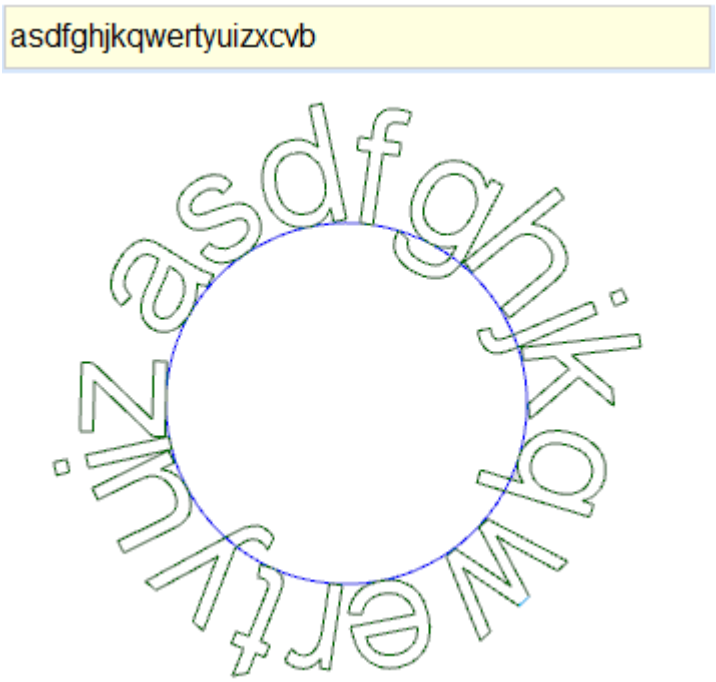
  Остальные три варианта применяются только если текст распределяется по геометрическому сегменту прямой, дуги или конуса:
  - **Выравнивание по центру**: текст отцентрован по сегменту
  - **Выровнять вправо**: текст выравнивается начиная с конечной части отрезка
  - **Автоматическое распределение текста**: выравнивает текст как в начальной, так и в конечной точке отрезка, добавляя необходимое пространство между символами для обеспечения равномерного распределения надписи. Выбор не является значимым, если надпись состоит только из одного символа.
- **Геометрия распределения текста**: распределение текста может задаваться со ссылкой на:
  - **точка:** с положением (X, Y) начала надписи и углом наклона надписи. Этот выбор всегда определяет применение **Выравнивание влево**.
  - **линейный отрезок:** с положением (X, Y) начала и конца отрезка. Опция **Инвертировать отрезок** позволяет применить инвертированную геометрию отрезка в начальной и конечной точках.
  - **часть окружности:** с положением (X, Y) начала и конца дуги, центра и вращения. Этот элемент может определять дугу или круг. Имеются следующие опции:
    - **Надпись внутри дуги***:* при выборе позволяет применить текст внутри дуги
    - **Инвертировать отрезок**: при выборе позволяет применить инвертированную геометрию отрезка
    - **дуга конуса:** с положением (X, Y) начала и конца дуги конуса, центра, крайней точки по одной из двух осей и вращением. Этот элемент может определять дугу конуса или целый конус с созданием эллипса или овала. Имеются следующие опции:
      - **Надпись внутри дуги***:* при выборе позволяет применить текст внутри дуги
      - **Инвертировать отрезок:** при выборе позволяет применить инвертированную геометрию отрезка



Bitmap позволяет считать (мышью) запрограммированный элемент распределения текста: точка, линейный отрезок или дуга: поля в зоне *Геометрия распределения текста* обновляются в зависимости от типа и геометрии выбранного отрезка. Затем настройки могут меняться исходя из потребностей, за исключением геометрического элемента конуса, для которого возможно только интерактивное определение.

При распределении на дуге круга или конуса реально создаваемая надпись не может превышать длину замкнутой фигуры. Сообщение сигнализирует, была ли обрезана надпись.

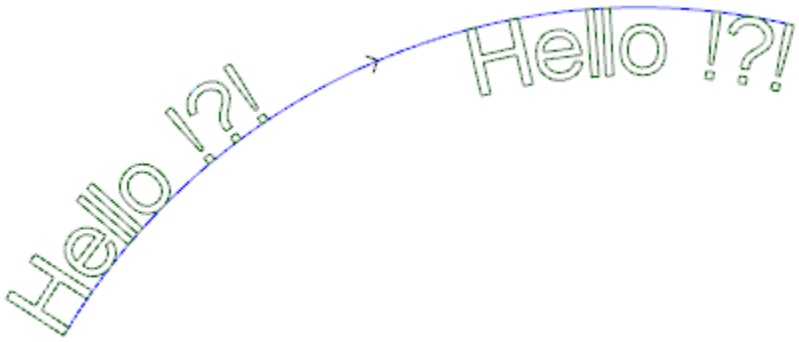
На рисунке ниже приводится пример, когда инструмент обрезал надпись:



Технологические задания вставленных профилей задаются при помощи следующих параметров:

- кнопка Технологии  с заданием подготовки, присвоенной профилям (код операции, технология, свойства операции).
- **Координата Z**: глубина выполнения профилей
- **Скорость интерполяции**: скорость, заданная для выполнения профилей.

Рассмотрим пример:



На рисунке представлен пример создания 2 одинаковых надписей с выбором неавтоматического распределения текста по круговому отрезку по часовой стрелке.

- Надпись слева создана без какого-либо дополнительного выбора.
- Надпись справа генерирована выбором выравнивания вправо и текста внутри дуги.

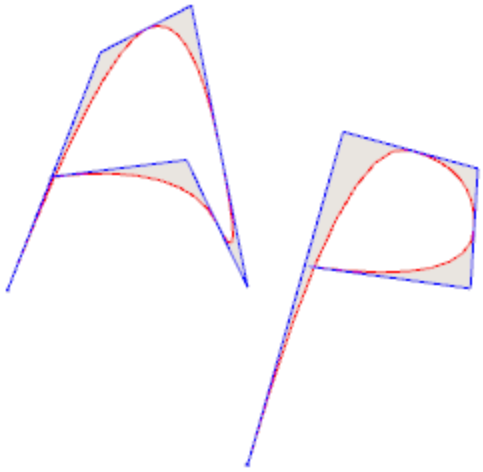
При наличии доступных шрифтов, заданных в клиентском режиме, на странице выбора используемого шрифта отображается также пункт выбора **Системного шрифта**:

- выбрать для использования одного из инсталлированных системных шрифтов (как описывается выше),
- оставить этот пункт невыбранным для использования одного из клиентских шрифтов. В этом случае: отсутствует выбор *Стиля* шрифта. Можно выбрать опцию обработки квадратичного B-сплайна для каждого из вставляемых профилей (см. инструмент: Создать сплайн из полилинии)

Все другие настройки интерпретируются с тем же значением, что и в общем случае использования системного штифта.

Файлы, задающие клиентский шрифт, архивируются в папке *TPACADCFG\CUSTOM\DBFONTS* с расширением *fcad*. Клиентский шрифт определяет более или менее полную совокупность символов (прописные и/или строчные буквы, цифры, знаки препинания и т. д.).

Символ описывается одним или несколькими профилями, каждый из которых символизируется как полилиния.
Подробное описание формата файла клиентского шрифта приводится в специальной документации.
Выбор клиентского шрифта осуществляется выбором в списке имен файлов.
На рисунке показан пример надписи с использованием гипотетического клиентского шрифта для написания "Ap":
• наружные контуры могут соответствовать исходным профилям
• внутренние контуры могут соответствовать профилям, обработанным кривой сплайн.
Фактическое изменение профилей в соответствии с логикой кривой сплайн зависит от исходного задания профилей каждого символа шрифта.
Зоны, показанные серым цветом, между кривыми каждой буквы, представлены на рисунке только для отображения расстояния между двумя кривыми.



Можно вводить тексты также путем вызова специальных макропрограмм в списке операций.
Пример:
в группе СПЕЦИАЛЬНЫЕ ФРЕЗЫ выберем операцию ТЕКСТ.

Это сложная операция, выполняемая при помощи макропрограммы и позволяющая задавать:
- Типичные параметры сложной операции (см. информацию, приведенную об общем коде подпрограммы):
  - **Qx,Qy Zp**: начальные координаты текста и глубина
  - **Позиционировать габаритный прямоугольник**: поле для размещения габаритного прямоугольника надписи
  - **Зеркальное отображение по горизонтали и Зеркальное отображение по вертикали**: активизирует требуемую симметрию
  - **Угол вращения (°)**: задает угол наклона текста
  - **Инвертировать**: активизирует инверсию при выполнении профилей
  - **Свойства операции**: задает свойства, присвоенные операции
- Задание технологии:
  - **Исходный элемент для Подготовки:** задать Имя операции setup, задающей технологию генерируемых операцией профилей (это поле предусматривается, если реализовано свойство Имя операций).
    Поиск операции осуществляется до текущей операции и должен соответствовать подготовке в режиме программирования в прямоугольной системе координат, а компиляция должна завершиться без ошибок. Поле **Комментарий** операции не может выбираться. На грани-детали эта операция должна применяться к одной грани. В случае множественных соответствий (более одной подготовки, запрограммированной с одним именем), выбирается последняя найденная, т. е. ближайшая к операции ТЕКСТ.
    В качестве альтернативы поле может задавать имя (параметр) Глобальной технологии (см. **Индивидуализировать->Технология->Коды по умолчанию**). В этом случае не требуется какого-либо дополнительного программирования.
    К заданиям свойств операции применяются те же критерии, которые используются в программировании всех сложных кодов, которые, как правило, соответствуют распространению ненулевых значений свойств, заданных для сложного кода (в нашем случае: операция ТЕКСТ).
    Рассмотрим практический пример: внешняя подготовка имеет слой L=2:
    - если операция ТЕКСТ имеет слой L=0: подготовка всех профилей сохраняет значение слой L=2
    - если операция ТЕКСТ имеет слой L=1: подготовка всех профилей будет иметь значение слой L=1.

    Исключение делается для поля B (конструкция) с учетом того, что принято задавать внешнюю подготовку конструкции для ее исключения из выполнения детали. В этом случае подготовками профилей, генерированных операцией ТЕКСТ, могут быть конструкции только если ТЕКСТ запрограммирована как конструкции.

В качестве альтернативы можно задать технологию профилей, задавая параметры, сгруппированные в узлы:

- **Технология, Продвинутые технологические данные:** два узла предлагают параметры для выбора инструмента, коррекции на инструмент, скорости и т.д.

ВНИМАНИЕ: если невозможно использовать внешнюю подготовку (не найдено какого-либо действительного соответствия для пункта **Исходный элемент для подготовки**), подается *Предупреждение* и технология профилей все равно задается с использованием настроек, относящихся к узлам**Технология, Продвинутые технологические данные.**

- Параметры, характерные для функций операции:
  - **Текст**: вводимый текст
  - **Шрифт**: тип символов, применяемых к тексту (список делает доступными все инсталлированные символы, для которых можно выбрать стили: нормальный, курсив, полужирный).
  - **Высота символа**: задает высоту буквы A (в единицах измерения детали)
  - **Расстояние между символами**: задает порядок определения расстояния между отдельными символами надписи (варианты: Метрическое, Геометрическое)
  - **Расстояние между символами**: расстояние между соседними символами
  - **Ширина интервала**: задает ширину пробелов при их наличии в текстовой надписи. Задать отрицательное значение для использования ширины шрифта.
  - **Угол наклона**: задает наклон каждого символа относительно базовой линии развертки надписи. Это поле задается в градусах (°) и десятых долях градуса.
  - **Полужирный:** подключение полужирного стиля
  - **Курсив**: подключение стиля курсива
  - **RightToLeft**: выбрать для написания справа налево (инвертирует положение символов текста)
  - **Элемент распределения текста**: задать Имя операции, задающей геометрию для распределения текста. Операция ищется до текущей операции и должна соответствовать линейному сегменту или дуге круга или дуге конуса. Кроме того, компиляция элемента должна завершиться без ошибок. Также не может иметь выбранным поле **Комментарий,** а при использовании на грани-детали применение должно осуществляться на той же грани, что и текущая операция (в нашем случае: ТЕКСТ). В случае множественных соответствий (несколько операций с одним именем) выбирается последняя найденная, т. е. ближайшая к операции ТЕКСТ
  - **Выравнивание**: выбрать режим выравнивания текста из четырех пунктов в списке. Если не задан элемент геометрического распределения (линейный сегмент или дуга), все равно будет применяться выравнивание влево.

Аналогичным образом доступна операция ГЕОМЕТРИЧЕСКОГО ТЕКСТА: с возможностью выбора шрифтов, заданных в клиентском режиме.

Создать сплайн из полилинии

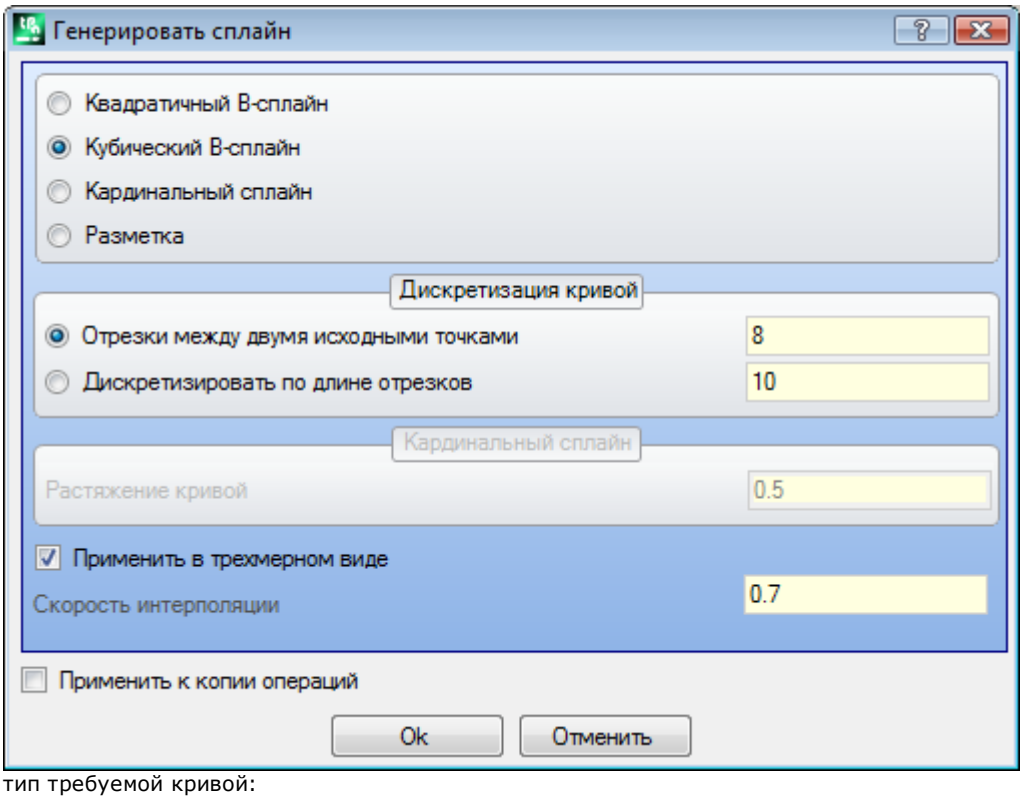
PROFESSIONAL

Команда **Создать сплайн из полилинии**  находится в группе **Построения** вкладки **Инструменты.**

Этот инструмент позволяет выполнять построение профиля по программированной полилинии.
Для каждого определенного профиля инструмент использует его вершины в качестве исходных точек (вершины управления или исходные точки) для создания кривой, интерполирующей вершины управления. На вычисленной теоретической кривой затем выполняется выборка по линейным отрезкам и в результате получается полилиния со следующими общими характеристиками:

- первая точка совпадает с начальной точкой исходного профиля
- последняя точка совпадает с конечной точкой исходного профиля
- прохождение кривой по другим промежуточным точкам исходного профиля зависит от типа выбранной кривой
- теоретическая кривая всегда непрерывная, без острых вершин.

Дуги и элементы разметки L24 исходного профиля рассматриваются как сегменты прямой. Все круги удаляются из профиля для оценки исходных точек. Кроме того, исходный профиль не может задавать дуги в плоскости, отличной от xy.



Выбрать тип требуемой кривой:

- **Квадратичный B-сплайн**: кривые вычисляются с решением квадратичной кривой Безье (требует не менее 3 исходных точек)
- **Кубический B-сплайн**: кривые вычисляются с решением кубической кривой Безье (требует не менее 4 исходных точек)
- **Кардинальный сплайн**: кривые вычисляются с особым решением кубической кривой Эрмита, которая называется кардинальным сплайном (требует не менее 3 исходных точек).
- **Разметка**: кривые вычисляются с решением одного единственного отрезка кривой рабочим кодом, который называется разметкой и иначе указывается как L24. Этот пункт может не присутствовать в окне. При выборе пункта *Разметка*: можно задать только поле **Скорость интерполяции.**

Выбрать тип выборки из двух опций:

- **Отрезки между двумя исходными точками**: число линейных сегментов, которые генерируются между двумя исходными точками (принимаются значения от 8 до 100). Это значение задает критерий выборки кривой. Предположим, что исходный профиль задан 5 линейными отрезками и что это поле равно 8: генерированная кривая будет иметь максимум 8*(5-2)=24 линейных отрезков (не генерируются отрезки длиной меньше *эпсилон* координаты * 5.0 или длиной, заданной в следующем поле);
- **Дискретизировать по длине отрезков**: выбрать для применения выборки на основе длины отрезков, на которые требуется разбить самый длинный отрезок исходного профиля. В этом случае: значение, заданное в предыдущем поле, игнорируется. Предположим, что самый длинный отрезок исходного профиля имеет длину 70 мм и что заданное здесь значение равно 0.5: автоматически вычисляется число выборок, равное 70/0.5=140 сегментов, выбранных для каждого сегмента, с минимальной длиной генерированных отрезков, в любом случае, не менее *эпсилон* координаты * 10.0.
  Даже если не требуется выполнять выборку на длине отрезков, заданное значение (в примере - 0.5), в любом случае, используется как минимальная длина фактически выбранных отрезков, с минимальным примененным значением *эпсилон* координаты * 5.0 и максимальным *эпсилон* координаты * 100.0.
- **Растяжение кривой**: значение, касающееся растяжения кривой в случае кривой *Кардинальный сплайн*. Действительны значения от 0.0 до 1.0 (недействительная настройка соответствует следующему интервалу):
  - 1.0 соответствует максимальному растяжению: вычисляемая кривая соответствует исходному профилю, разбитому на указанные сегменты;
  - 0.0 соответствует минимальному растяжению: вычисляемая кривая соответствует ситуации с максимальными отклонениями от исходного профиля;
- **Применить в трехмерном виде**: выбрать, чтобы активировать решение кривой также в соответствии с координатой глубины, создавая тем самым кривую в пространстве. Если это поле не выбрано, создаваемая кривая задает координату Z только на подготовке;
- **Скорость интерполяции**: задает скорость выполнения кривой сплайн

Генерированные профили открываются с копией исходной подготовки, при ее наличии, или с копией эталонной подготовки (как задано **Индивидуализировать->Технология->Коды по умолчанию** меню Применение).

Выбор опции *Применить к копии операций* применяет этот инструмент к копии операций и не меняет исходные линии.

Рассмотрим более подробно характеристики полученных кривых.
Первые два случая, кривые Безье, имеют общие характеристики:
- как уже говорилось: крайние точки кривой совпадают с крайними точками исходного профиля;
- начало кривой касательно первой стороне исходного профиля;
- конец кривой касателен последней стороне исходного профиля;
- кривая никогда не проходит по промежуточным точкам исходного профиля;

В компьютерной графике кривые Безье используются, например, для моделирования скошенных кривых.
Типовым примером являются системные шрифты TrueType, которые образованы квадратичными кривыми Безье.
В случае *Кардинального сплайна*:
- как уже говорилось: крайние точки кривой совпадают с крайними точками исходного профиля;
- кривая проходит по всем промежуточным точкам исходного профиля
- часть кривой между двумя исходными точками может находиться за пределами исходного отрезка.
- в случае замкнутой исходной полилинии развертка не зависит от точки начала/конца полилинии.

В случае *Разметки*:
- крайние точки кривой совпадают с крайними точками исходного профиля;
- кривая проходит по всем промежуточным точкам исходного профиля
- часть кривой между двумя исходными точками может находиться за пределами исходного отрезка
- полученная здесь кривая является непрерывной, без острых вершин, но только вследствие конкретного выбора.

Разметка имеет значение, отличное от других выборов кривых сплайн, т. к. связана с определенной операцией (L24), которая может использоваться и независимо от рассмотренного здесь инструмента.
В кривой *Разметка*, генерированной автоматически из исходной полилинии:
- первый элемент начинается с касательной первого отрезка полилинии
- завершает каждый элемент L24 касательной, заданной на последующем отрезке полилинии
- начинает каждый последующий элемент непрерывной касательной с предыдущим.

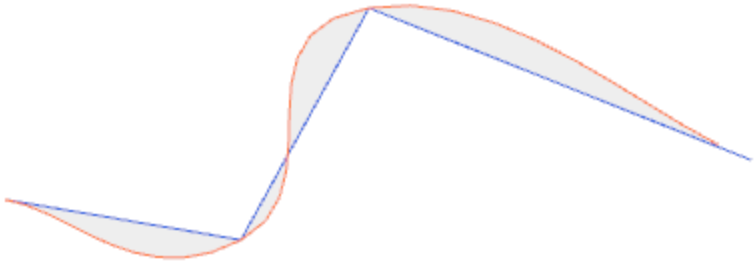
Генерированная таким образом разметка может затем меняться исходя из потребностей.
Ниже приводится пример создания кривых сплайн при выборе пункта **Квадратичный B-сплайн:**



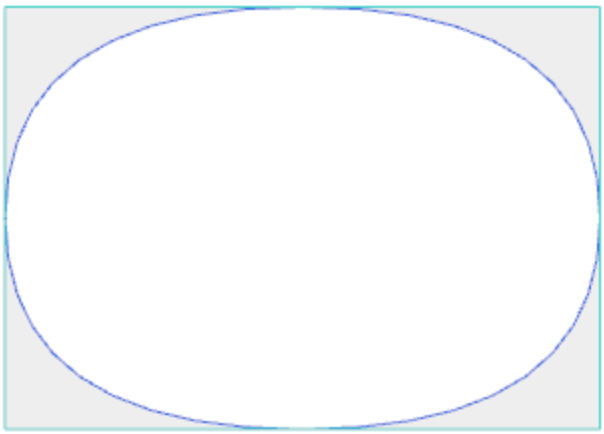
Из этого же профиля при выборе пункта **Кубический B-сплайн** получается кривая:



Пример создания кривых при выборе пункта **Кардинальный сплайн**:



В качестве последнего примера рассмотрим создание кривой **Квадратичный B-сплайн,** примененной к прямоугольнику/квадрату, введенному, например, командой меню **Чертеж**. Построенная кривая соответствует, в целом, эллиптическому маршруту:



Зоны, показанные на рисунке серым цветом между изначальной кривой и кривой сплайн, обозначают только расстояние между двумя кривыми.

Можно генерировать кривые сплайн также в форме сложной операции, вызывая операцию типа *Запрограммированные инструменты* в списке операций. В группе ИНСТРУМЕНТЫ выбирается операция STOOL: SPLINE:

- Поле **Операции** задает имена, заданные запрограммированным ранее операциям, соответствующим исходным профилям.

Профили могут быть получены также в результате применения сложных кодов и выполнение операции STOOL: SPLINE относится только к измененным профилям и не включает исходные профили. Операции, не используемые для целей требуемых функций (например, точечные или логические операции или сложные неразбиваемые операции), игнорируются.

Операция задает:

- Типичные параметры сложной операции (см. информацию, приведенную об общем коде подпрограммы):
  - **Qx,Qy Zp**: начальные координаты размещения разработанных операций.
  - **..**
  - **Свойства операции**: задает свойства, присвоенные операции.

- Это особые параметры функции операции со значением, аналогичным полям в окне инструмента:
  - **Тип кривой:** выбирает между кривой квадратичный B-сплайн, Кубический B-сплайн или Кардинальный сплайн (**ВНИМАНИЕ**: не кривые разметки)
  - **Отрезки между двумя исходными точками**: число линейных сегментов, которые генерируются между двумя исходными точками (принимаются значения от 8 до 50)
  - **Растяжение кривой**: задать растяжение кривой (значение от 0.0 до 1.0), используемой в случае кривой *Кардинальный сплайн*
  - **Применить в трехмерном виде:** выбрать, чтобы активировать решение кривой также в соответствии с координатой глубины
  - **Скорость интерполяции**: задает скорость выполнения кривой сплайн.

- Задание технологии:
  - **Исходный элемент для Подготовки:** задать Имя операции setup или Глобальной технологии, задающей технологию генерируемых операцией профилей (это поле предусматривается, если реализовано свойство Имя операций или есть задания Глобальных технологий). Интерпретация и

применение соответствуют указанному в параграфе **Создание текстов** применительно к операции ТЕКСТ, к которому и следует обращаться.

Основное преимущество использования операции STOOL:SPLINE состоит в том, что кривые адаптируются к изменениям исходных профилей.

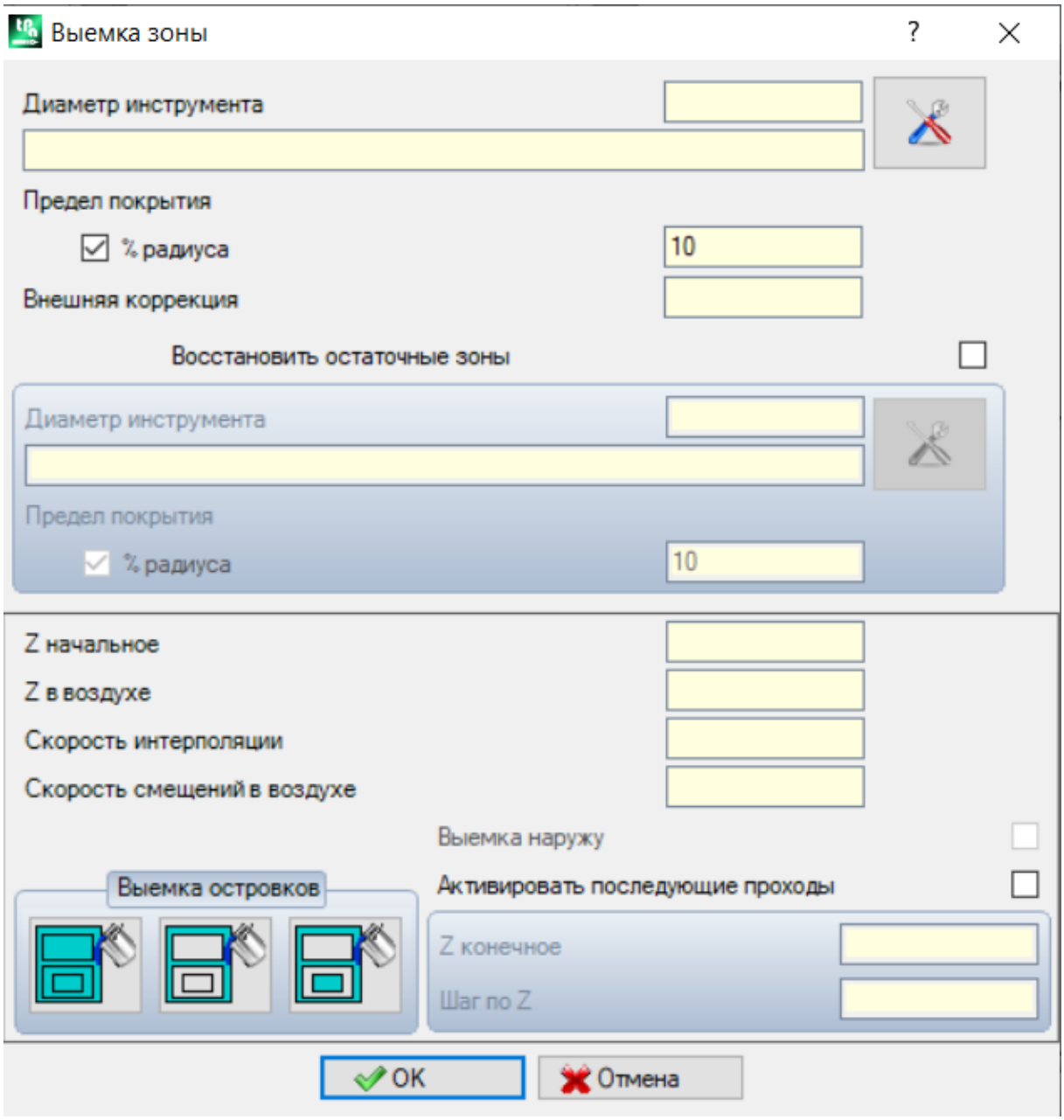
Выемка зоны



Этот инструмент позволяет выполнять выемку зоны, определенную замкнутым профилем, за счет вставки профилей выемки непосредственно в грань. Команда **Выемка зоны**  находится в группе **Построения** вкладки **Инструменты.**

В случае грани-детали этот инструмент активизирован только при активизированном двухмерном или каркасном представлении и действует только на профили, примененные к грани на текущем виде. Этот инструмент не рассматривает профили, имеющие активизированный параметр **Профиль выемки**. Этот параметр реализуется в подготовительных операциях с целью маркировки профилей, генерированных в ходе процедуры выемки.

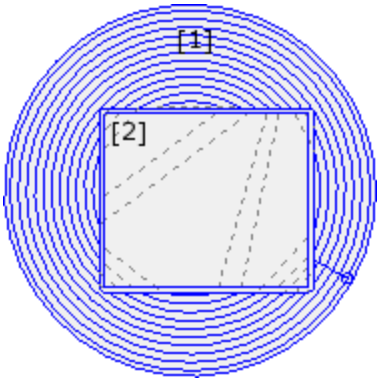
Кроме того, процедура выемки учитывает максимум 300 профилей.



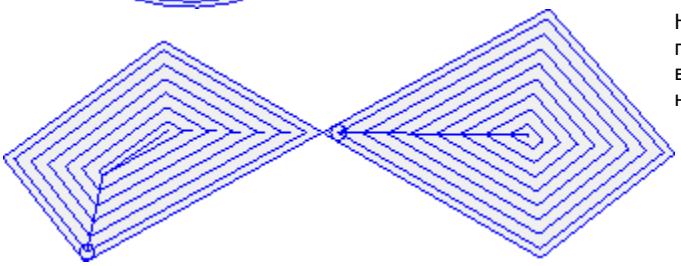
Выемка замкнутой зоны выполняется с созданием профиля внутри зоны, состоящего из последующих проходов, получаемых за счет постепенного смещения относительно исходного профиля до полного покрытия внутренней зоны. Если требуется, выемка обеспечивает сохранение внутренних замкнутых зон (островков) и делает попытку зачистить зоны с неполной выемкой ввиду необходимости соблюдения пределов исходно оставляемой зоны.
Привести текущую операцию на профиль, на котором необходимо выполнить выемку, и выбрать команду из меню.
Рассмотрим подробно поля в окне:

- **Диаметр инструмента**: задает диаметр инструмента. При помощи кнопки Технология можно выбрать код и технологию операции setup для использования в маршруте выемки. соответствующий диаметр указан в поле.
- **Предел перекрытия**: указывает, на сколько накладываются последующие проходы инструмента. Это поле может выражаться в абсолютных величинах (мм) или в **% заданного радиуса** инструмента. Инструмент интерпретирует положительное значение:
  - минимум равен 10% радиуса инструмента
  - максимум равен радиусу инструмента
- **Внешняя коррекция**: указывает, на сколько необходимо выйти из запрограммированного профиля при выполнении первого прохода. Это значение выражено в единицах измерения детали (мм или дюймах) и

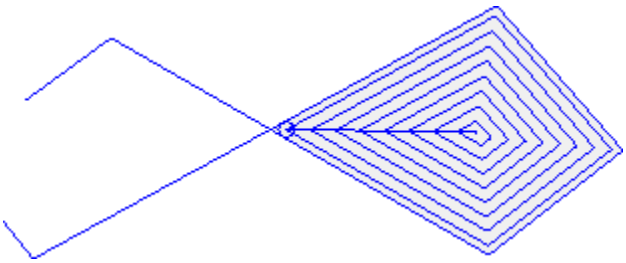
вычитается из значения радиуса инструмента. Интерпретирует как значимое (т. е. не равное нулю) положительное значение, которое больше или равно эпсилон * 10 и меньше или равно радиусу коррекции. Например, с эпсилон = 0.001 мм и единицей измерения программы в [мм]:
  - минимальное значение для этого поля равно 0.01
  - если задано меньшее значение: не применяет *Внешнюю коррекцию*
- **Зачистка оставшихся зон**: при выборе активизирует пункты следующего блока. Позволяет задавать вторую технология для выемки для использования в случае, если имеются внешние зоны, с выемкой, выполненной неполностью при применении основной технологии. Технология инструмента для зачистки задается таким же образом, как и технология по умолчанию:
  - операция остается такой же, что и в технологии по умолчанию
  - остается возможность задать поля технологии: заданный здесь инструмент должен иметь габарит (диаметр) меньше габарита (диаметра) инструмента по умолчанию, т. к. он должен работать в зонах с меньшим габаритом.
- **Z начальная**: задает координату глубины при выполнении профиля/ профилей выемки. Если требуется выполнение следующих проходов на разной глубине (см. ниже), соответствует глубине первого прохода
- **Z в воздухе**: задает координату безопасного подъема инструмента для добавленных перемещений в воздухе
- **Скорость интерполяции**: задает скорость перемещений на этапе выемки
- **Скорость смещения в воздухе**: задает скорость смещений, выполняемых на координате в воздухе. Отрезки опускания от координаты в воздухе до рабочей глубины выполняются на скорости врезки инструмента, заданной в подготовке (кнопка "Технология"). Если скорость опускания инструмента не задана, отрезки опускания выполняются на скорости смещения в воздухе.
- **Активировать последующие проходы**: активирует повторение цикла выемки на нескольких проходах, выполняемых на различной глубине
  - **Z конечная**: это конечная глубина, которая должна достигаться при выполнении последнего прохода
  - **Шаг по Z**: это изменение глубины для применения на последующих проходах
- **Выемка островков**: возможно три выбора (графические кнопки), слева направо:
  - **Игнорировать островки**: выполняет выемку зоны, определенной профилем, игнорируя замкнутые контуры внутри нее
  - **Выемка только наружу:** выполняет выемку внутри зоны, исключая, однако, замкнутые контуры внутри нее
  - **Выемка попеременно**: выполняет выемку внутри зоны, определенной профилем. При определении внутри нее замкнутого контура прерывает выемку до тех пор, пока не встретит другой замкнутый контур внутри предыдущего и от него возобновляет выемку, продолжая далее таким же образом.
- **Выемка наружу**: при активировании запрашивает выполнение выемки начиная изнутри зоны. Этот пункт может выбираться только при активированном пункте **Игнорировать островки.**



На рисунке показана выемка круга (1) с прямоугольным островком (2).
Выемка зоны между двумя профилями выполняется путем выполнения профиля, проходящего внутрь с последующими сокращениями.
Профиль выемки прерывается у прямоугольного островка: инструмент поднимается и перемещается в воздухе над островком (пунктирные отрезки) и опускается на рабочую высоту при возврате в зону выемки



На рисунке показана выемка профиля, генерирующая несколько замкнутых зон, выемка каждой из которых выполняется независимо.



На рисунке показана выемка незамкнутого профиля: проверка выемки выполняется по наличию замкнутых зон.

Можно выполнять выемки также в форме сложной операции путем вызова специальных макропрограмм в списке операций:

- в группе СПЕЦИАЛЬНЫЕ ФРЕЗЫ выбираются операции: ВЫЕМКА ПРЯМОУГОЛЬНИКА, ВЫЕМКА МНОГОУГОЛЬНИКА, ВЫЕМКА ЭЛЛИПСА…; с использованием сложной операции, присваивающей определенную замкнутую геометрию (прямоугольник, многоугольник, эллипс…) на основе заданных геометрических параметров и критериев ее выемки;

- в группе ПОДПРОГРАММЫ выбирается операция SEMPTY: она использует код применения подпрограммы, соответствующим образом предусмотренный для задания критериев выемки профилей вытекающих из применения подпрограммы;

- в группе ИНСТРУМЕНТЫ выбирается операция STOOL: EMPTY: поле **Операции** задает имена, заданные запрограммированным ранее операциям, соответствующим исходным профилям. Профили могут быть получены также в результате применения сложных кодов и выполнение операции STOOL:EMPTY относится только к измененным профилям и не включает исходные профили. Операции, не используемые для целей требуемых функций (например, точечные или логические операции или сложные неразбиваемые операции), игнорируются.

Во всех указанных случаях задающие выемку параметры находятся в специальном окне, производном из окна, которое рассматривалось для инструмента **Выемка зоны.**

В группе СПЕЦИАЛЬНЫЕ ФРЕЗЫ выберем операцию ВЫЕМКА ПРЯМОУГОЛЬНИКА.

Это сложная операция, выполняемая при помощи макропрограммы и позволяющая задавать:

- Типичные параметры сложной операции (см. информацию, приведенную об общем коде подпрограммы):
  - **Зацепление точки**: требуется зацепление к заданной ранее части профиля
  - **Зеркальное отображение по горизонтали и Зеркальное отображение по вертикали**: активизирует требуемую симметрию
  - **Угол вращения (°)**: задает угол наклона текста
  - **Свойства операции**: задает свойства, присвоенные операции

- Параметры, характерные для геометрии операции. В нашем случае:
  - **Центр X, Центр Y**: центр прямоугольника
  - **Длина, Высота**: размеры прямоугольника
  - **Радиус**: радиус на ребрах

- Параметры, характерные для функции выемки:
  - **Подготовка выемки:** можно задать число или строку
    - число присваивает код операции setup профилю выемки. Его можно задать путем прямого редактирования или путем выбора кода из списка setup, показанного в окне
    - строка задает Имя операции setup, присваивающей технологию, задаваемую профилю выемки (например, “aa”). Поиск операции осуществляется до текущей операции и должен соответствовать подготовке в режиме программирования в прямоугольной системе координат, а компиляция должна завершиться без ошибок. Кроме того, не может иметь выбранным поле **Комментарий**, а при использовании на грани-детали применение должно осуществляться на той же грани, что и текущая операция (в нашем случае: ВЫЕМКА ПРЯМОУГОЛЬНИКА).
      К заданиям свойств операции применяются те же критерии, которые используются в программировании всех сложных кодов, которые, как правило, соответствуют распространению ненулевых значений свойств, заданных для сложного кода (в нашем случае: операция ВЫЕМКА ПРЯМОУГОЛЬНИКА).
      Рассмотрим практический пример: внешняя подготовка имеет уровень L=2:
      - если операция ВЫЕМКА ПРЯМОУГОЛЬНИКА имеет слой L=0: подготовка всех профилей сохраняет значение уровня L=2
      - если операция ВЫЕМКА ПРЯМОУГОЛЬНИКА имеет слой L=1: подготовка всех профилей будет иметь значение уровня L=1.

      Исключение делается для поля B (конструкция) с учетом того, что принято задавать внешнюю подготовку конструкции для ее исключения из выполнения детали. В этом случае подготовками

профилей, генерированных операцией ВЫЕМКА ПРЯМОУГОЛЬНИКА, могут быть конструкции только если ВЫЕМКА ПРЯМОУГОЛЬНИКА запрограммирована как конструкция.
- строка может также задавать имя (параметр) Глобальной технологии (см.: **Индивидуализация TpaCAD->Технология->Коды по умолчанию**). В этом случае не требуется какого-либо дополнительного программирования.
- если поле не присвоено, используется код setup по умолчанию.

В качестве альтернативы можно задать технологию профилей, задавая параметры в:
- **Выемка**: поле открывает окно, аналогичное окну, реализованному инструментом, для задания параметров процедуры выемки: критериев выемки (предел перекрытия, направление выемки, реализация проходов и островков) и технологии.

ВНИМАНИЕ: если сложная операция предусматривает возможность создания профилей для **Зачистки оставшихся зон**, параметр **Подготовка выемки** может задавать:
- если значение цифровое, код операции setup также для генерированных профилей восстановления. В этом случае технология подготовки задается в **Выемке**;
- если строка, может задавать также второе Имя операции setup. Поиск операции осуществляется до текущей операции по тем же критериям подготовки, что и первичная выемка. Примеры:
"aa;bb": "aa" - это имя, используемое для поиска подготовки для задания первичным выемкам. "bb" - это имя, используемое для поиска подготовки для задания выемкам восстановления.

ВНИМАНИЕ: при невозможности использования одной или обеих внешних подготовок (для пункта **Подготовка выемки** ни одно соответствие не было признано действительным), подается *Предупреждение* и технология профилей, в любом случае, задается с использованием настроек, связанных с окном **Выемка**.

Поворот профилей в картезианской плоскости

Этот инструмент позволяет поворачивать один или несколько профилей вокруг одной из двух осей координат плоскости, задающей эту грань. Команда **Поворот профилей в картезианской плоскости**  находится в группе **Построения** вкладки **Инструменты.**
В случае грани-детали этот инструмент активизирован только при активизированном двухмерном или каркасном представлении и действует только на профили, примененные к грани на текущем виде.
Применяется к:
- всем профилям, имеющим по меньшей мере один выбранный элемент
- текущему профилю.

Этот инструмент не работает, если недоступны операции выполнения дуги на плоскостях: xz, yz, xyz.

Этот инструмент поворачивает профиль/ профили на 90° вокруг одной из двух осей координат плоскости грани. Рассмотрим возможные опции:

- **Сохранять одну точку вращения**: выбор действует при применении на нескольких профилях. Выбрать пункт для сохранения одного центра вращения для всех профилей: центр совпадает с начальной точкой первого поворачиваемого профиля. Если этот пункт не выбран: каждый профиль поворачивается вокруг своей исходной точки;

- **Плоскость вращения**: выбор выполняется из двух значений:
  - **Zx**: вращение выполняется вокруг оси X грани
  - **Yz**: вращение выполняется вокруг оси Y грани.

  Посмотрим, как изменились координаты каждого отрезка:

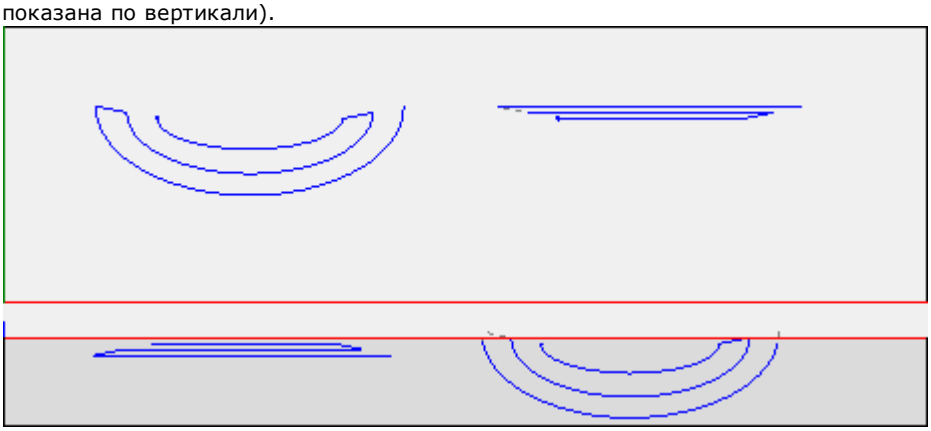
координаты по оси	если плоскость Zx	если плоскость Yz
X	они остаются заданными по оси X	они приводятся по оси Z
Y	они приводятся по оси Z	они остаются заданными по оси Y
Z	они приводятся по оси Y	они приводятся по оси X

- **Изменение на повернутой оси**: выбрать из двух предлагаемых пунктов, указывая режим обмена изменениями задействованных в обмене координат
  - **Совпадает**: изменения переставляются с сохранением знака;
  - **Не совпадает**: изменения переставляются с изменением знака на обратный.

  Применение выбора учитывает режим программирования оси глубины. Например, при плоскости вращения Zx: положительные изменения по оси Y приводятся в изменениях по оси Z

Например, этот инструмент может повернуть профиль выемки по двум плоскостям. На рисунке представлено:
- начало эллиптической выемки, запрограммированной в плоскости грани (профиль слева)
- в правой части профиль повернут в плоскости Zx.

Вверху приводится двухмерное изображение грани; внизу приводится передний вид грани (ось глубины показана по вертикали).



Выбор опции *Применить к копии операций* применяет этот инструмент к копии операций и не меняет исходные линии.
ВНИМАНИЕ: элементы разметки (L24) разбиваются на микроотрезки, задающие кривую.
ВНИМАНИЕ: этот инструмент обнуляет коррекцию на инструмент на маршруте.

PROFESSIONAL

Можно вызывать инструмент вращения на картезианской плоскости также в форме сложной операции, вызывая операцию типа *Запрограммированные инструменты* в списке операций. В группе ИНСТРУМЕНТЫ выбирается операция STOOL: STPLANE:
- поле **Операции** задает имена, заданные запрограммированным ранее операциям, соответствующим исходным профилям.

Профили могут быть получены также в результате применения сложных кодов и выполнение операции STOOL: STPLANE относится только к измененным профилям и не включает исходные профили. Операции, не используемые для целей требуемых функций (например, точечные или логические операции или сложные неразбиваемые операции), игнорируются.

Операция задает:
- Типичные параметры сложной операции (см. информацию, приведенную об общем коде подпрограммы):
  - **Qx,Qy Zp**: начальные координаты размещения разработанных операций.
  - **..**
  - **Свойства операции**: задает свойства, присваиваемые операции.

- Это особые параметры функции операции со значением, аналогичным полям в окне инструмента.

Основное преимущество использования операции STOOL: STPLANE состоит в том, что кривые адаптируются к изменениям исходных профилей.

10.5 Построения раскроя профилей

Раскрой

PROFESSIONAL

Этот инструмент позволяет позиционировать один или несколько маршрутов в прямоугольный или переменный контур для занимания минимально возможного пространства и, при необходимости, повторяя применение несколько раз. Размещение отдельных маршрутов выполняется, оценивая габаритный прямоугольник или форму маршрутов, которые могли быть приведены в положение минимального габарита путем трансформации поворота маршрутов.

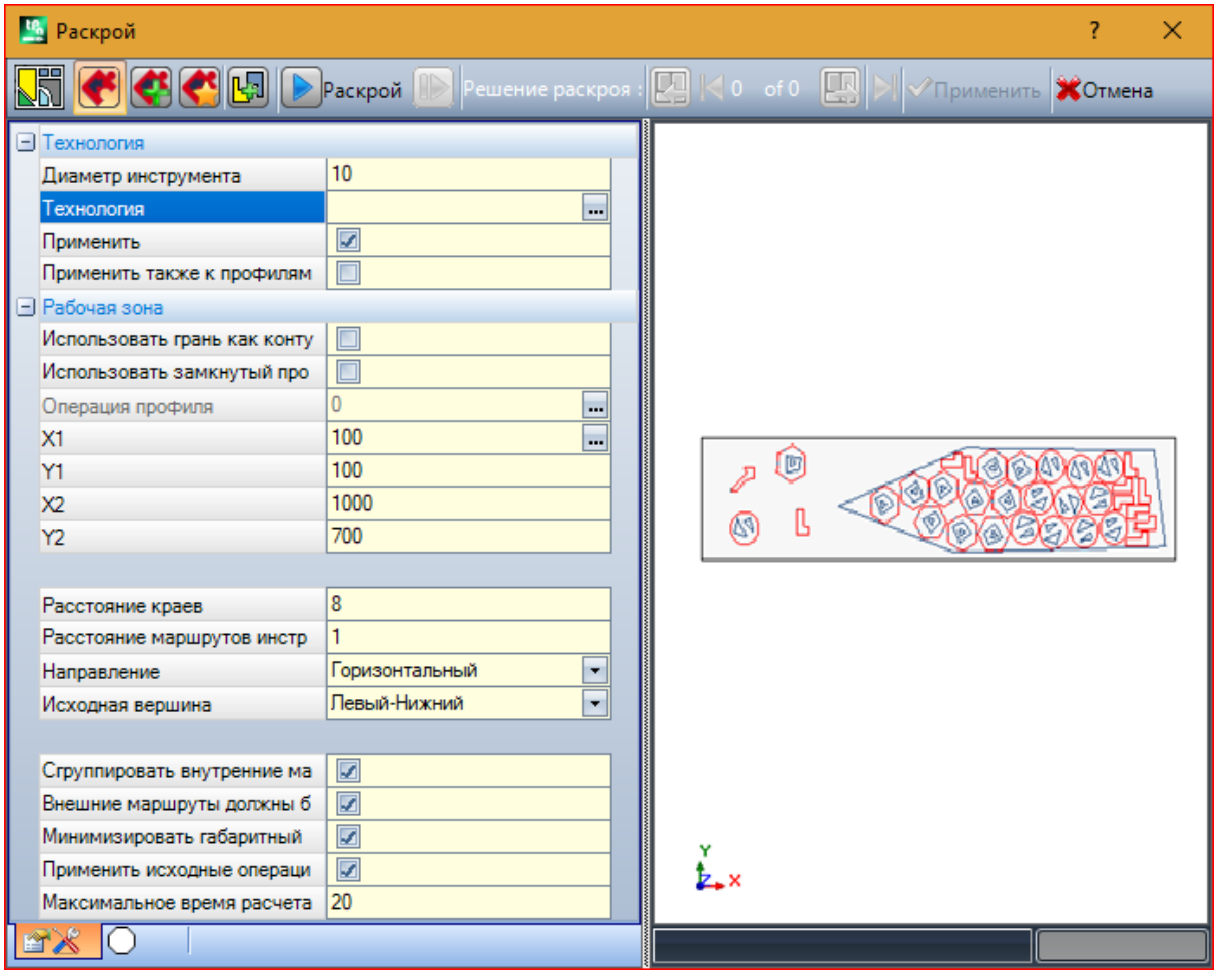
Функциональность инструмента не зависит от конкретного включения аппаратным ключом двумя возможными уровнями работы:
- Прямоугольный раскрой
- Раскрой True Shape (Фигурный Раскрой).

Если доступна работа True Shape (Фигурный Раскрой), вы можете выбрать уровень, который хотите применить.

Команда **Раскрой**  находится в группе **Раскрой профилей** вкладки **Инструменты.**

В случае грани-детали инструмент действует только на профили, примененные на грани на текущем виде.

Инструмент работает на выбранных или текущем профиле. Профили должны быть простыми и не обязательно замкнутыми. Оценка общих размеров профилей не учитывает запросы на исправление инструмента.
Можно создать автоматические и/или ручные группы тех же профилей для размещения более сложных форм.



Предлагается окно, разделенное на две области:
- слева настройки делятся на две страницы
- справа - графический просмотр применения инструмента.

Окно является размерным, а вертикальная разделительная панель позволяет изменять размеры двух зон.

Верхняя панель группирует общие доступные команды:

 Кнопка работает только в том случае, если *True Shape* доступен и изменяет изображение в зависимости от состояния выбора:

- выбор неактивен: соответствует работе *Раскрою True Shape*, и предлагаемое изображение является показанным выше
- выбор активен: соответствует работе *прямоугольному Раскрою*, и предлагаемое изображение  . Выбор соответствует единственному возможному состоянию, если *Раскрой True Shape* не доступен, и в этом случае кнопка становится невидимой.

Прямоугольный Раскрой управляет размещением деталей путем применения правил, которые соответствуют габаритным прямоугольникам (ограничительный каркас) каждой детали. Размещения выполняются в прямоугольном контуре.

Раскрой True Shape управляет размещением деталей, путем применения правил, которые соответствуют реальным размерам каждой детали. Размещения выполняются в любом закрытом контуре.

Следующие варианты доступны и применимы только в случае *Раскрой True Shape*:

-  **Допустить размещения на отверстиях:** выбирать, чтобы разрешить размещение внутри внутренних профилей, но только для групп, у которых есть соответствующий элемент в выбранном столбце. Если выбор неактивен, размещение не может быть применено во внутренних профилях независимо от групповых заданий
-  **Размещать в отверстиях рекурсивно:** выбрать, чтобы разрешить рекурсивные размещения во внутренних профилях
-  **Сосредоточиться на размещениях на отверстиях:** активный выбор способствует размещению внутри внутренних профилей
-  **Автоматические кластеры:** выбор позволяет применять автоматические кластеры профилей в отношении к отдельнм размещениям, а только для профилей, которые имеют соответствующий элемент в выбранном столбце. Для каждого профиля в списке, проверяется эффективность, которая может быть получена из автоматического кластера: кластер, который назначает эффективность, большую или равную значению, установленному в конфигурации функции Раскроя (см. руководство о функции Раскроя, главу **Конфигурация Раскроя->Опции Раскроя**) определяет привилегированное применение группы в отношении к размещению одного профиля. Исключаются из применения автоматических кластеров геометрические профили, соответствующие окружностям, коническим или правильным многоугольникам (вписанным на окружности).
-  **Размещения по матрице**: выбор позволяет применять размещения согласно матричной схеме, но только для деталей, имеющих соответствующий элемент в выбранном столбце. Опция может быть использована для генерации равномерного размещения в соответствии с рисунком по сетке. Детали, для которых требуется размещение по матрице, используются раньше остальных и размещаются с расположением столбцов * строк в зависимости от свободного места на панели. Чтобы определить метод размещения, каждый фрагмент также можно проанализировать, применяя стратегии автономного спаривания, чтобы оптимизировать сетку размещения. Матричное размещение выполняет размещения деталей с повторением единицы, которая может соответствовать одной детали, всегда повторяющейся с тем же самым вращением, или двумя частями, с взаимным соответствием, определенным с вращением на 180 °. Повторяющийся блок, состоящий из одной или двух частей, затем можно позиционировать, оценивая изменение поворота на 0 ° или 90 °.

 **Раскрой**: кнопка запускает проверку настроек и последующую оптимизацию раскроя. Команда может быть отменена, если настройки требуют исправлений.

Давайте подробно рассмотрим страницы настроек:

- **Технология:** выбрать технологию для присвоения профилям.
  - **Применить:** выбор для применения технологии
  - **Применить также к профилям с заданной технологией**: применяет технологию также к профилям, уже открытым операцией setup. Если эта опция не выбрана, инструмент применяется

только к открытым профилям или профилям, которые начинаются с операции ГЕОМЕТРИЧЕСКОЙ ПОДГОТОВКИ;

Выбор технологии не требуется: если он не выполняется, каждый профиль будет применяться без правки по сравнению с исходным технологическим программированием. В частности, можно применять профили с другим диаметром технологии: в этом случае геометрический габаритный прямоугольник каждого профиля увеличивается на заданный диаметр.

- **Рабочая область:** узел группирует поля, полезные для назначения контура, полезного для размещений

  - **Использовать грань как контур**: выбрать для использования всей текущей грани в качестве габаритного прямоугольника для размещений. В качестве альтернативы можно присвоить контур, указывая две крайние точки:
  - **X1, Y1**: координаты точки минимального габарита
  - **X2, Y2**: координаты точки максимального габарита.

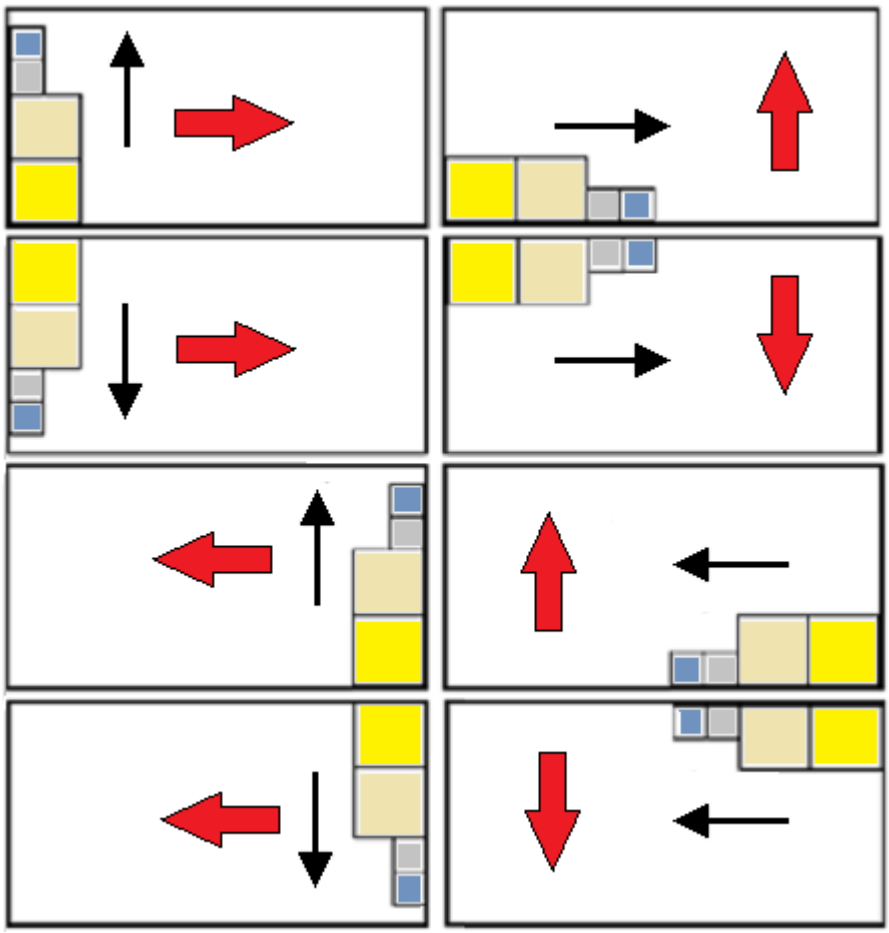
  Выбрать значок  поля **X1** для интерактивного считывания профиля: габаритный прямоугольник профиля автоматически задает поля 4 координат.
  Фиксируется минимальное значение для габарита контура размещения, применяемое для обоих размеров и равное максимальному габариту указанных профилей.
  - **Использовать замкнутый профиль как контур:** выбор доступен и применим только в случае *Раскроя True Shape.* Выбрать, чтобы разрешить использование контура, заданного закрытым профилем:
    - **Операция профиля:** выбрать значок  для выбора профиля в интерактивном режиме. Профили, используемые для размещений, исключаются, и профиль должен быть закрыт. Поля (**X1, Y1, X2, Y2**) обновляются автоматически со значениями, соответствующими размерам самого профиля.

- **Расстояние краев**: свободное пространство от краев прямоугольника заполнения
- **Расстояние маршрутов инструмента**: расстояние, прибавляемое к диаметру технологии, для определения фактического расстояния размещаемых деталей

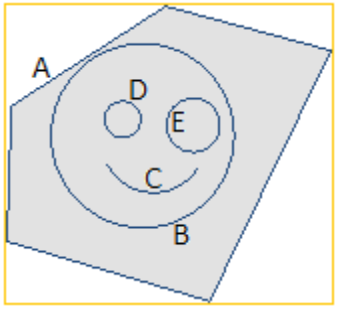
- **Направление**: выбрать направление подачи для размещений из двух предлагаемых пунктов:
  - Горизонтальное (на рисунке: случаи справа с горизонтальной красной стрелкой)
  - Вертикальное (на рисунке: случаи слева с вертикальной красной стрелкой)

- **Исходная вершина**: выбирает исходную вершину для размещений из четырех предлагаемых пунктов:
  - Слева-Внизу (на рисунке: случаи на первой строке)
  - Слева-Вверху (на рисунке: случаи на второй строке)
  - Справа-Внизу (на рисунке: случаи на третьей строке)
  - Справа-Вверху (на рисунке: случаи на четвертой, последней, строке)

Настройки **Направление** и **Исходная вершина** можно игнорировать при использовании переменного контура.



- **Сгруппировать внутренние маршруты**: выбрать для группирования в один элемент маршруты, имеющие габарит внутри другого маршрута. Пример группы, распознанной автоматически, соответствует рисунку.



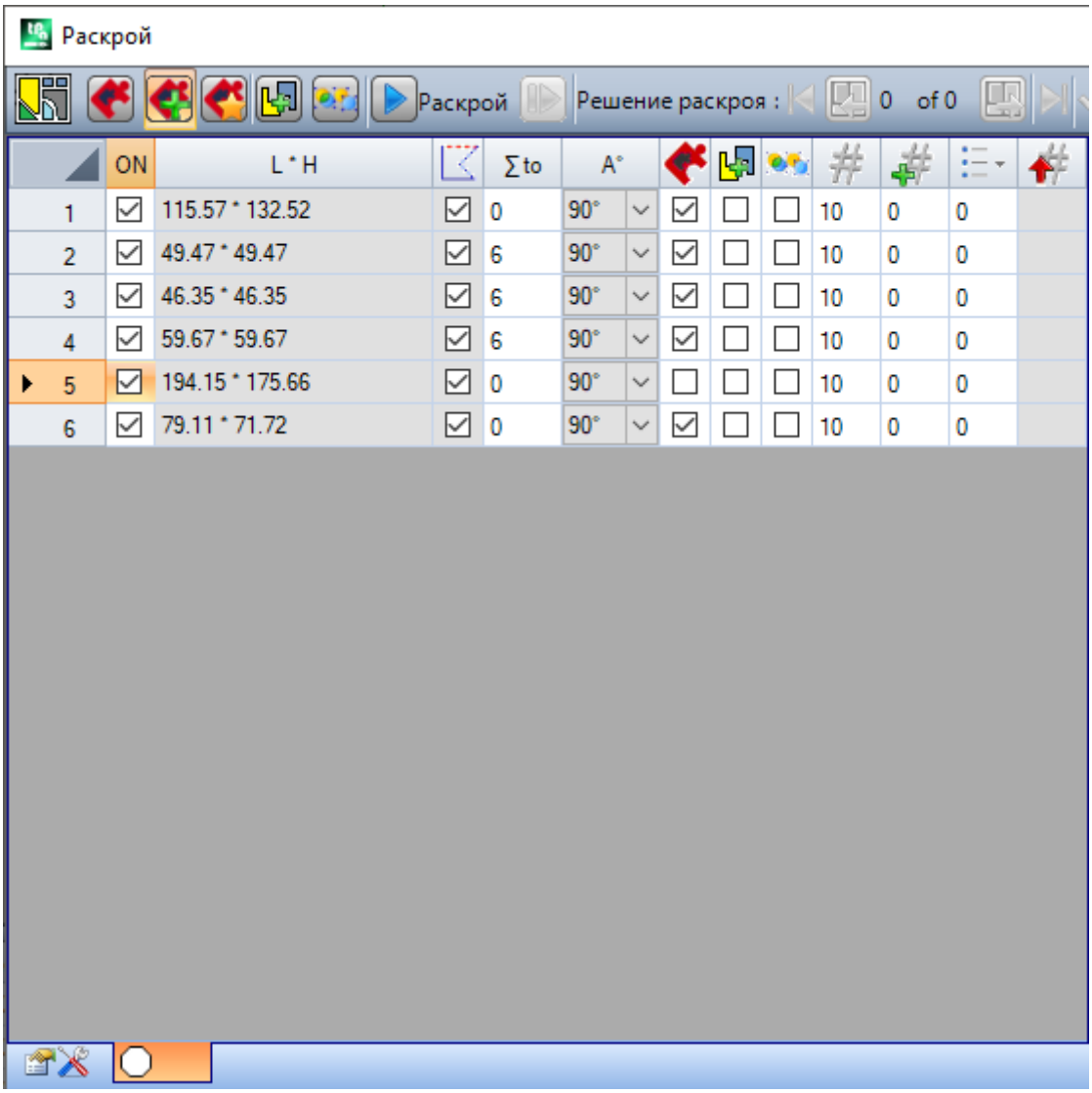
A - внешний профиль
B, C, D, E - это профили внутри **A**: габаритный прямоугольник каждого из них находится внутри габаритного прямоугольника **A**
Совокупность 5 профилей образует одну группу, а возможные позиционирования сохранят без изменений взаимные положения исходных маршрутов. Минимальная группа состоит из одного маршрута.

- **Внешние маршруты должны быть замкнуты**: выбрать, должны ли внешние маршруты быть геометрически замкнуты. В этом случае и в приведенном выше примере: **A** не распознавался бы как внешний профиль, если бы не был замкнутым.
ВНИМАНИЕ: из оценки крайних точек профиля исключены входные/выходные отрезки, запрограммированные на операции setup

- **Минимизировать габаритный прямоугольник**: выбрать, чтобы включить поиск вращения, которое соответствует минимальному размеру каждой группы маршрутов, для которых включено вращение (см. Вторую страницу настроек). Эта новая позиция заменяет оригинальную, для последующего применения возможных поворотов.
  Задание задается автоматически всем группам.

• **Применить исходные операции**: выбрать для размещения также исходных маршрутов. При неактивном выборе исходные операции не меняются и их позиционирование никоим образом не участвует в определении процедуры раскроя
- **Максимальное время расчета [с]:** задает максимальное время, за которое следует завершить расчет раскроя, выраженное в секундах. Минимальное значимое значение - 20 секунд, значение 0 отключает все временные пределы.

На второй странице окна перечислены группы профилей с возможностью изменения порядка их задания и размещения.



После применения заданий предыдущей страницы задания в таблице могут претерпевать более или менее комплексные изменения. Подтверждение настроек с отображенной первой страницей может не применяться напрямую, а вызывать подключение самой страницы так, чтобы позволить увидеть возможные автоматические изменения.

Каждая строка таблицы задает группу, а каждый столбец задает задание группы:

• **Заголовок строки**: номер по порядку, присвоенный автоматически и используемый в качестве однозначного идентификатора (ID) группы
• **ON**: выбранная клетка подключает использование группы. Выбрать ячейку заголовка столбца для правки клетки всех строк в таблице (при наличии выбранных строк правка ограничивается ими)

• **L*H**: размеры габаритного прямоугольника (эти поля не редактируются)
• : клетка выбрана, если внешний маршрут группы геометрически замкнут (поле не может редактироваться).
Столбец может быть не виден, если внешние пути должны быть замкрнуты (см.: настройка на предыдущей странице)

• **Σ to**: клетка может присвоить номер по порядку группы, с которой будет ассоциироваться текущая группа, в режиме ручного группирования. Согласно заданию на рисунке, группы (2, 3, 4) ассоциируются с группой 6:
  - таким образом, группа 6 будет состоять из 4 профилей,
  - габаритный прямоугольник образованной таким образом группы будет соответствовать объединению габаритов каждого исходного маршрута
  - возможные позиционирования будут сохранять без изменений взаимные положения исходных маршрутов

ВНИМАНИЕ: при подключенной опции **Внешние маршруты должны быть замкнуты** и если группа 6 состоит из одного профиля, ручное группирование будет применяться только если профиль геометрически замкнут. Если группа 6 задана с несколькими профилями, это условие уже выполняется.

Можно активировать быстрое задание вручную. В нашем примере:
  - выбрать строки групп (2, 3, 4)
  - щелкнуть правой кнопкой мыши ячейку заголовка группы 6

задание клеток столбца **Σ to** будет выполняться автоматически

• : выбрать клетку для подключения размещения группы также с поворотом на 90° относительно оригинального или минимального габарита, определенного автоматически. Если клетка не выбрана, деталь может размещаться только как изначально.

В случае *Раскроя True Shape*, выбор выполняется в списке из 3 записей, с возможностью выбора и вращения *any*, применимого только в функциональности *True Shape* (значение, соответствующее вращению *any*, назначается в *Конфигурации раскроя*).

ВНИМАНИЕ: при группировании нескольких профилей, достаточно только одного профиля, имеющего свои ограничения на вращение, чтобы деактивировать поле группы.
Выбрать ячейку заголовка столбца для правки клетки всех строк в таблице (при наличии выбранных строк правка ограничивается ими).

• : выбрать, чтобы разрешить размещение во внутренних профилях группы. Выбор игнорируется, если соответствующее полное включение на командной строке деактивировано. Столбец может быть невидимым, и выбор имеет смысл только в случае размещения *True Shape*

• выбрать поле, чтобы включить применение автоматического кластера профиля по сравнению с одиночным размещением. Вывбор игнорируется, если соответствующее полное включение на командной строке деактивировано.
Столбец может быть невидимым, и выбор имеет смысл только в случае размещения *True Shape.*

выбрать поле, чтобы запросить размещение в соответствии с разработкой по матрице. Выбор игнорируется, если соответствующая общая активация на панели команд деактивирована. Столбец может быть невидимым, и выбор имеет смысл только в случае размещения *True Shape* и размещения в прямоугольной рабочей области.

• : размещаемое количество. Задать положительное значение (>=0) не более 999. Поле инициализируется со значением **Повторения,** заданным на предыдущей странице. Можно править задание, дифференцируя требуемое количество для каждой группы. Задание группы, ассоциированной вручную с другой, игнорируется и применяется значение исходной группы.
В случае подключения только одной группы и значения 0 процедура раскроя будет стремиться разместить максимальное число возможных повторений для этой группы.
• : значение выше предыдущего задает максимальное используемое количество (не более 999):

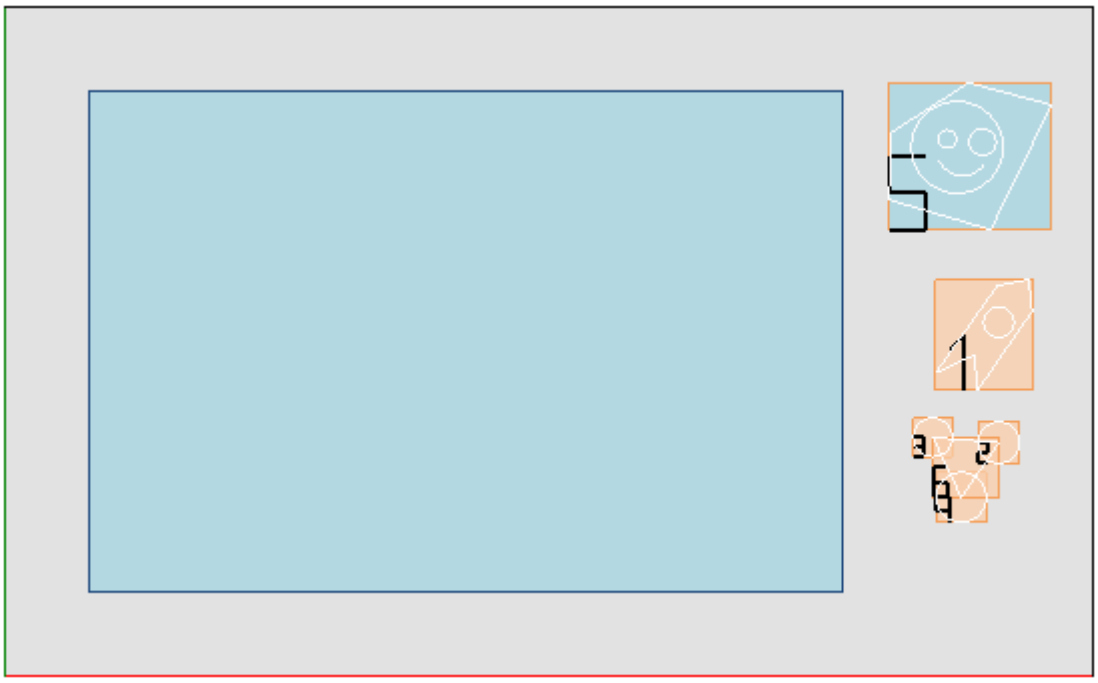
разница между двумя значениями - это количество, которое может использоваться для заполнения зоны заполнения только после размещения всех групп. Задание группы, ассоциированной вручную с другой, игнорируется и применяется значение исходной группы

 : столбец управляется автоматически и показывает количество, фактически используемое после решения Nesting (поле не может быть отредактировано).


• **Приоритет**: группы с назначенным приоритетом имеют преимущество в решении раскроя (значение по умолчанию: 0; максимальное значение: 100). Как интерпретируется значение приоритета, определяется опцией **Более низкий приоритет с возрастающим значением** в Конфигурации раскроя (см. руководство по функциям раскроя, в главе: **Конфигурация раскроя->Опции раскроя**).

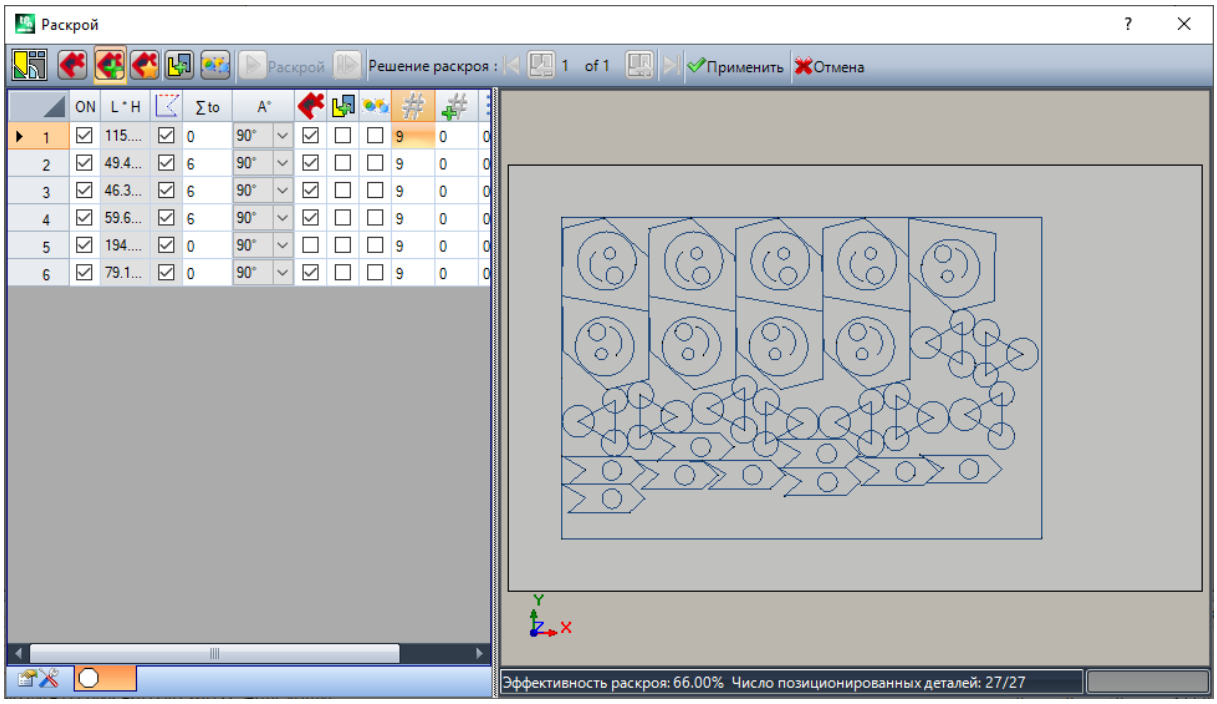
Максимальное количество профилей, которое можно запросить, составляет 10000.

Общие задания окна соответствуют представленной ниже ситуации



В левой части выделен прямоугольник размещения.
В правой части выделены группы профилей, каждая заключенная в габаритный прямоугольник и обозначенная номером по порядку:
• группы 1 и 5 показывают применение опции **Сгруппировать внутренние маршруты**
• остальные группы (2, 3, 4, 6) не группируются автоматически, т. к. не отвечают критерию внутренних маршрутов. Как показано выше, здесь они могут группироваться вручную в одну группу (в примере: 6).
При изменении текущей строки в таблице соответствующая группа выделяется графически (на рисунке: группа 5).

Предполагая *прямоугольный Раскрой* подтверждение настроек дает результат, показанный на рисунке:



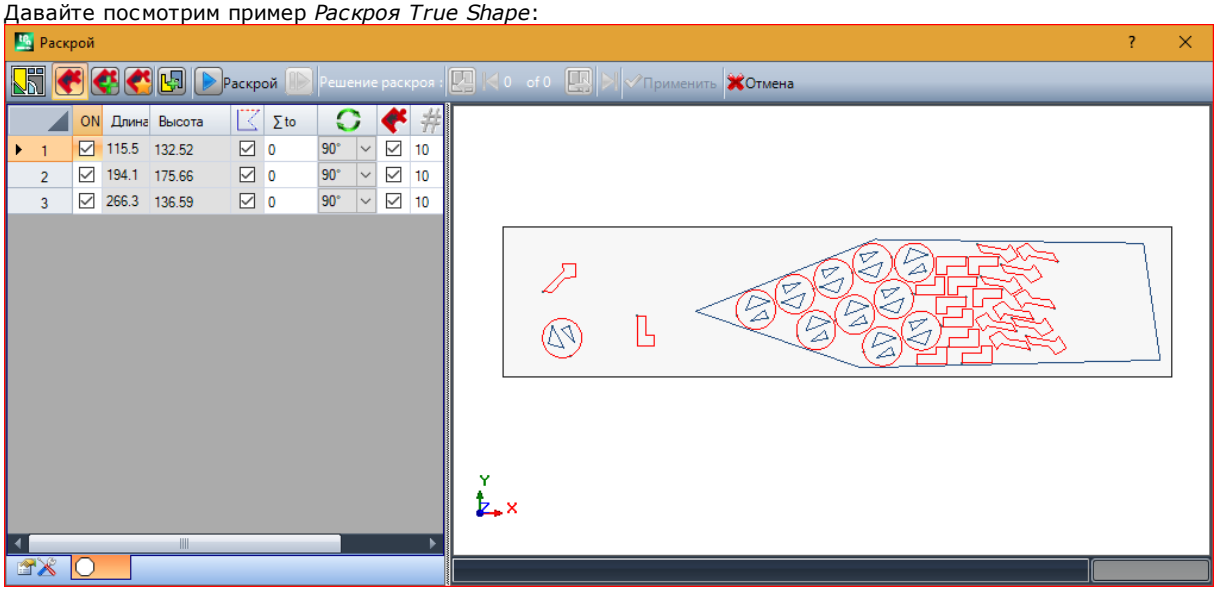
На рисунке выделен прямоугольник, заданный для размещения профилей.
Для каждой группы было выполнено размещение 9 пар, в общей сложности 27/27 запрошенных размещений.

Видно, как каждая группа может позиционироваться по-другому относительно исходных операций: подключение на поворот активировало поиск положения, соответствующего минимальному габариту. Размещения группы 1 (профиль, подобный стрелке с внутренним кругом) показывают, как возможны повороты на 90°.

Изменив настройки, можно будет запросить новое применение раскроя.

Применить: выбрать, чтобы применить результаты к текущей программе
Отмена: выбрать, чтобы закрыть окно, отменив команду.

Давайте посмотрим пример *Раскроя True Shape*:



Размещение профилей теперь возможно находится в непрямоугольной области.

 **Вновь запустить**: кнопка запрашивает снова запустить процедуру определения Раскроя *True Shape*, предполагая, что последнее решение рассчитано как отправная точка. Затем вы можете найти больше решений, максимум до 10, и прокрутить их, выбрав то, которое считается лучшим:

 **Перейти к следующему решению:**

 **Перейти к предыдущему решению:**

две кнопки позволяют прокручивать и активировать одно из вычисленных решений.

 **Применить:** выбрать, чтобы применить результаты текущего решения к текущей программе.

Запрос на применение процедуры Раскроя приводит к выполнению различных проверок действительности заданий с возможностью подачи сигналов и/или правки выполненных настроек. Рассмотрим подробно:

- габарит контура размещения не проверяет минимальные размеры
- по меньшей мере одна группа в списке должна быть активной.

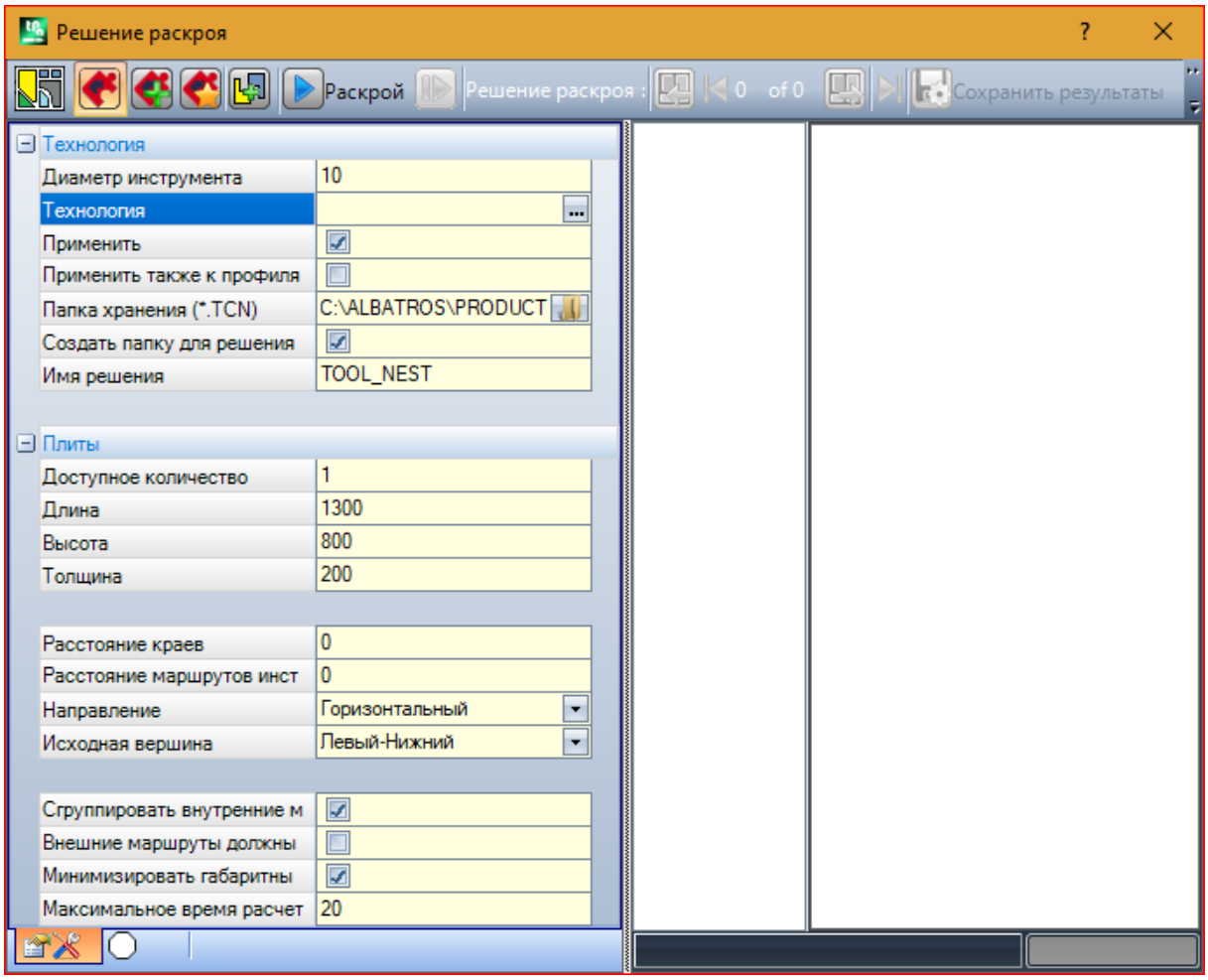
Решение раскроя



Команда **Решение раскроя**  находится в группе **Раскрой профилей** вкладки **Инструменты**. Команда недоступна в демонстрационном режиме или при активном виде грани, отличном от 1. Комплексное представление инструмента подобно предыдущей главе и ссылается на него для предварительного анализа.

Характеристикой инструмента является то, что теперь размещение групп профилей выполняется не в редактируемой программе, а с созданием новых программ (*.TCN), которые иначе указываются как листы.

Первая страница настроек имеет несколько вариантов по сравнению с первой страницей настроек команды **Раскрой**.

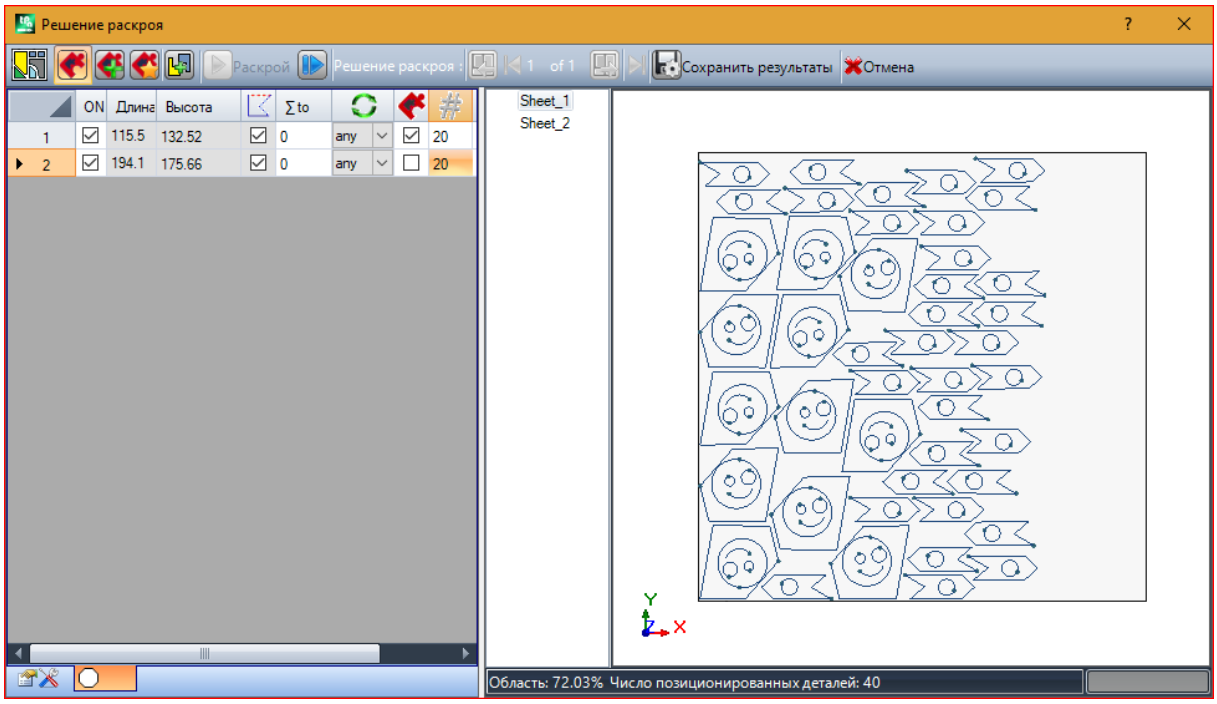


- **Папка хранения (.TCN)**: это маршрут, выбранный для *решения*
- **Создать папку для решения**: вместе с клеткой создается папка для архиваций, связанных с решением. Клетка выбрана и не может быть изменена
- **Имя решения**: имя, присвоенное решению.

Папка, которая будет создаваться для регистрации программ решения, должна указываться в пункте **Папка хранения**. Имена программ имеют общую матричную схему, состоящую из имени, определенного в поле **Имя решения**, за которым следует '_' (символ подчеркивания), буква для дифференцирования от предыдущих сохранений ('a', 'b',...) и номер по порядку.
Примеры составных имен: "tool_nest_a1", "tool_nest_a2", "tool_nest_b1".

Информация о плитах:
- **Доступное количество**: задать положительное значение (>=0) не более 100. Если значение - 0, процедура рассчитывает число панелей, необходимых для размещения общего числа групп профилей
- **Длина, Высота, Толщина**: размеры плит. Каждая программа (*.TCN) будет создаваться с заданными здесь размерами: единица измерения - такая же, что и в редактируемой программе.

Это изображение является примером применения *Раскроя True Shape*:



Решение может привести к созданию нескольких программ TCN: на рисунке 2.
Вы можете создать новые решения, как указано о предыдущем инструменте, и выбрать решение для применения.

По завершении процедуры отображаются сообщения, описывающие исход и указывающие число размещений и число зарегистрированных программ.

Программы TCN создаются с использованием файл прототипа программ или панелей раскроя в качестве отправной точки, в зависимости от того, что назначено в конфигурации функции Раскроя (см. руководство о функции Раскроя, главу **Конфигурация Раскроя->Опции Раскроя**).
В зависимости от параметров сохранения программ *.TCN и результатов работы функции Раскроя, может быть открыто окно для выбора также сохранения файлов с преобразованием формата, среди тех, которые доступны в конфигурации.

По завершении выполнения инструмента путь, заданный для сохранения программ *.TCN, задается как последний открытый для последующего открытия программы.

10.6 Продвинутые инструменты в программе грани

Создать фиктивную грань из геометрии

PROFESSIONAL

Эта команда подключена на виде грани и с программой с непустой гранью. Инструмент **Создать фиктивную грань из геометрии** вызывается в группе **Продвинутые** вкладки **Применить.**

В случае грани-детали:
- эта команда не работает, если текущая операция применяется к автоматической грани;
- эта команда не работает, если текущая операция применяется к автоматической или фиктивной грани типа кривой или поверхности;

- это команда отображает двухмерное представление, с гранью на виду, соответствующую грани применения текущей операции.

Это инструмент для упрощенного создания фиктивных граней на основе линейных или закругленных отрезков, уже запрограммированных на грани.
При обнаружении линейного отрезка, относящегося к ориентированному профилю, дается запрос, **ориентировать ли грань на основе профиля:**

- при положительном ответе создается грань, наклоненная относительно вертикального направления грани;
- при отрицательном ответе создается грань, вертикальная текущей грани.

Возможность создания грани из закругленного элемента определяется возможностью управлять Закругленными гранями.
После нахождения отрезка (мышью) открывается окно заданий фиктивных граней.

Поля задаются исходя из положений, определенных по указанному отрезку.
Вставка фиктивной грани в перечень граней программы осуществляется после выхода с подтверждением и проверки геометрической правильности грани.
После вставки можно продолжать команду, указывая другой линейный отрезок, или выйти из команды, нажимая клавишу **[ESCAPE]**.

ВНИМАНИЕ: если подключено управление закругленными гранями, можно выбрать также закругленный геометрический элемент (дугу в плоскости xy грани).

ВНИМАНИЕ: эта команда отсутствует на виде закругленной фиктивной грани или заданной как поверхность.

Грань, вставленная при помощи этой процедуры, никоим образом не остается связанной с линейным отрезком, использованным для ее задания. Этот отрезок может меняться или удаляться и это никоим образом не повлечет за собой автоматическое изменение или удаление грани.

Создать поверхность, исходя из геометрии

PROFESSIONAL

Эта команда подключена на виде грани и с программой с непустой гранью. Инструмент **Создать поверхность, исходя из геометрии**  находится в группе **Продвинутые** вкладки **Применить**, с активным подключением функции.

Описание ассоциированной функции см. в специальной документации, которую можно вызвать командой "Руководство по моделированию" в меню  .

Создать моделирование, исходя из геометрии

PROFESSIONAL

Эта команда подключена на виде грани и с программой с непустой гранью. Инструмент **Создать моделирование, исходя из геометрии**  находится в группе **Продвинутые** вкладки **Применить**, с активным подключением раздела Моделирование.

Описание ассоциированной функции см. в специальной документации, которую можно вызвать командой "Руководство по моделированию" в меню  .

Создать шрифт, исходя из геометрии

PROFESSIONAL

Эта команда подключается на виде грани с программой с непустой гранью и может быть не подключена в зависимости от конфигурации TpaCAD. Инструмент **Создать шрифт, исходя из геометрии** вызывается в группе **Продвинутые** вкладки **Применить.**
Эта команда позволяет задавать символ клиентским шрифтом начиная с запрограммированных профилей.

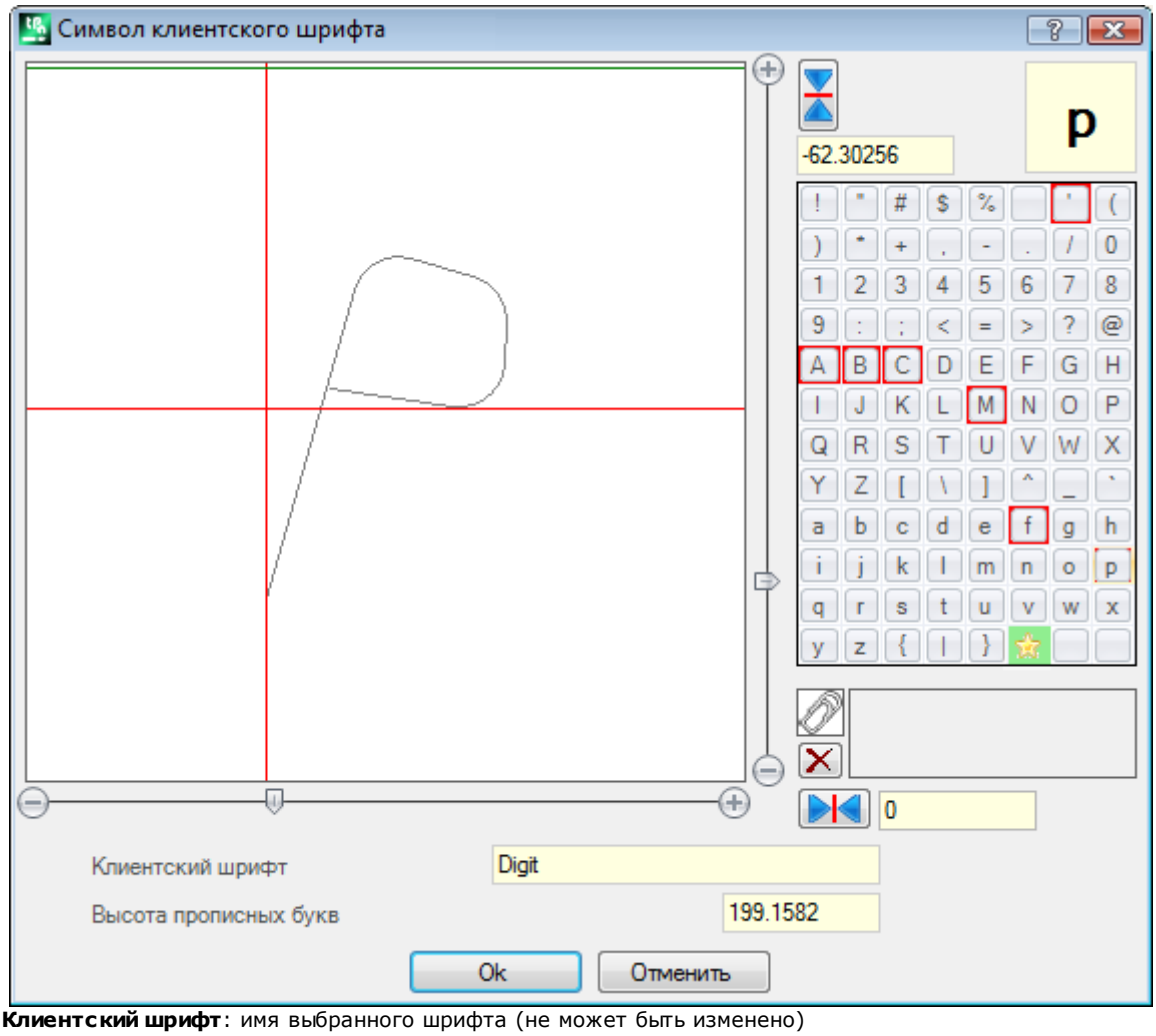
Скажем сразу, что целью TpaCAD является возможность использования клиентских шрифтов, а не полное тщательное их генерирование: описанный здесь инструмент покажется ограниченным, но он предусматривается только в качестве дополнительной функции, поэтому не должен считаться неполным.

Формат клиентских шрифтов прост и документирован и в базовой инсталляции TpaCAD приводится его пример.

Любой пользователь может создать по желанию другие, используя фоновое изображение текущей грани и вычерчивая полилинии по фоновому рисунку.
Инструмент *Создание шрифта, исходя из геометрии* может в этом случае использоваться для создания первой редакции символов.
Начиная с программ TpaCAD, например, импортированных из внешнего формата (DXF или др.), можно, например, предусмотреть разложение модуля *Экспорт в файл шрифта*.

Как уже говорилось, символ в файле клиентского шрифта описывается одним или несколькими профилями, каждый из которых характеризуется как полилиния.
Изначально предлагается список файлов, инсталлированных в качестве клиентских шрифтов, с возможностью задания нового шрифта.
Затем запускается процедура интерактивного выбора рассматриваемых профилей:
• можно выбрать только простые профили, т. е. полностью заданные линейными отрезками, дугами или разметкой;
• закругленные отрезки на плоскости не грани считаются линейными отрезками.

При закрытии интерактивной процедуры с подтверждением открывается окно, дополняющее команду:



- **Клиентский шрифт**: имя выбранного шрифта (не может быть изменено)
- **Высота прописных букв**: высота шрифта над базовой линией. Это значение определяет, насколько символ шрифта может подниматься над линией написания. Это поле неизменяемо и задается при создании шрифта (т. к. при задании первого символа шрифта) равным суммарному вертикальному габариту выбранного профиля/профилей. При создании шрифта обязательно следует начинать с задания прописной буквы **A**. В случае добавления к уже заданному шрифту это значение нельзя изменить.

Значение **Высота прописных букв** имеет соответствие также в системных шрифтах и может использоваться процедурами применения шрифта для масштабирования и позиционирования символов на линии написания.

На изображении слева в масштабе показаны выбранные профили.
Размеры этой зоны определяются таким образом, чтобы включить весь символ, позиционированный на верхний квадрант, как показано курсором в виде двух красных линий:

- точка пересечения двух линий является нулем размещения символа по горизонтали и по вертикали;
- размеры зоны над горизонтальной красной линией определяются как минимум равными большему значению из высоты шрифта и вертикального габарита представленных профилей;
- размеры зоны под горизонтальной красной линией определяются как минимум равными высоте шрифта.

Горизонтальная зеленая линия, приведенная в верхней части изображения, соответствует высоте шрифта.

При задании первого символа шрифта можно позиционировать символ только по одной горизонтальной линии курсора, в противном случае символа можно позиционировать вдоль обеих линий.
Позиционирование курсора на горизонтальной линии задает фиксированный сдвиг, добавляемый к горизонтальному габариту символа: этот сдвиг будет со знаком "плюс" при перемещении символа вправо и со знаком "минус" при перемещении влево.
Позиционирование на вертикальной линии курсора задает сдвиг относительно базовой линии применения символа. Этот сдвиг будет со знаком "плюс" при перемещении вверх и со знаком "минус" при перемещении вниз.
Значение смещений приводится в двух ячейках рядом со линейками прокрутки с возможностью прямого задания значений. Как показано на рисунке, для показанного символа задано:
- смещение вниз, равное -60.51295 мм
- нулевое смещение по горизонтали.

Позиционирование символа осуществляется при помощи двух линеек прокрутки, расположенных с правой стороны (вертикальная линейка) и внизу (горизонтальная линейка) изображения, у двух красных линий:
- переместить подвижную часть для неточного позиционирования;
- для точного позиционирования выбрать крайние кнопки (**+** и **-**).

Две кнопки, предусмотренные на линии с линейками прокрутки, содержат соответствующие отклонения от нулевого исходного значения:


обнуляет смещение по горизонтальной оси


обнуляет смещение по вертикальной оси.

Рассмотрим примеры написания **A g l q `.** В использованном системном шрифте они имеют значение расширения:
- под базовую линию - символы **g** и **q**;
- над базовой линией - символ **`**. Символ **`** , например, может иметь заданным также горизонтальный сдвиг (со смещением символа вправо от красного вертикального курсора) для увеличения реального габарита.

Горизонтальная линейка прокрутки в верхней части перемещает синюю вертикальную линию и позволяет задавать положение зацепления последующего символа. Позиционирование значимо, если находится справа от красной вертикальной линии (как показано на рисунке в примере).

ВНИМАНИЕ: горизонтальный сдвиг (красная вертикальная линия) и положение зацепления (синяя вертикальная линия) применяются в режиме **Метрическое расстояние**. В режиме Геометрического расстояния рассматриваемый габарит соответствует реальному габаритному прямоугольнику отдельного символа.

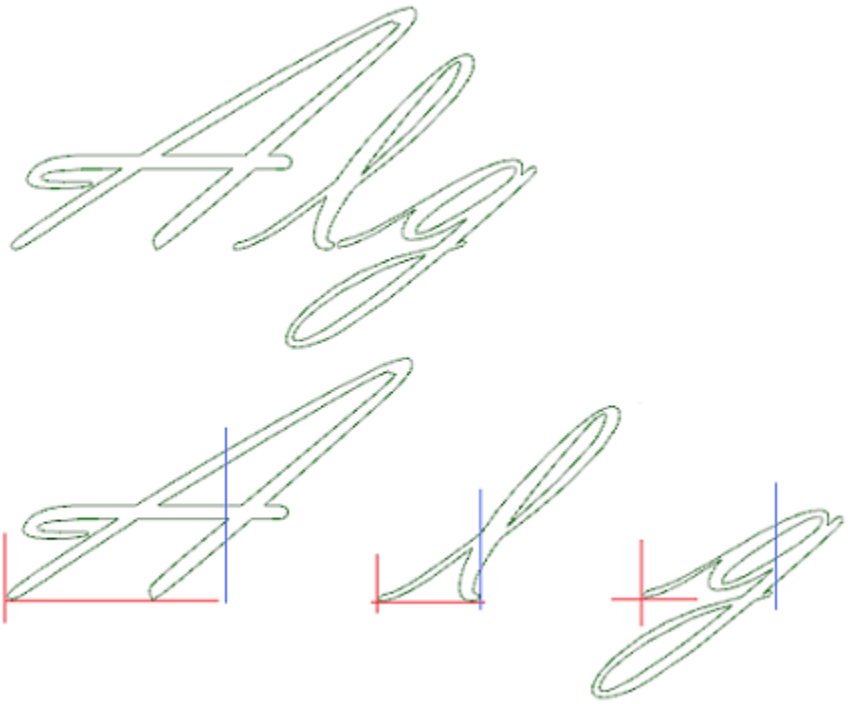
В правой части окна показаны:
- заданный символ (на рисунке: **p**);
- карта символов, которую можно задавать. Символы, уже заданные в шрифте, имеют красную каемку, которая используется только для выделения и не блокирует их модификацию.

Задаваемый символ может модифицироваться непосредственно в ячейке или при его выборе в карте.
Графический символ в виде звездочки - символ, используемый для записи незаданного символа.


Ассоциированное текстовое поле позволяет задавать символы, “использующие” один символ (в нашем примере на рисунке буква **P** будет представлена тем же профилем, что и буква **p**). Текстовое поле нельзя изменить напрямую:
- нажать кнопку **CTRL** и символ в карте для добавления или удаления символа из поля;
- нажать кнопку  для обнуления текстового поля.

Как уже говорилось, при создании шрифта обязательно следует начинать с задания прописной буквы **A**, при этом карта символов не интерактивна.

На рисунке задан клиентский шрифт исходя из инсталлированного системного шрифта:
- в верхней части рисунка надпись “Alg”создана с метрическим размещение
- в нижней части рисунка указаны положения курсоров, как они заданы в окне Клиентский шрифт.



10.7 Инструменты утилиты

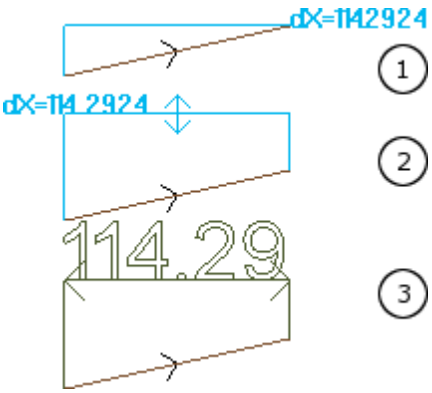
Проставление размеров

PROFESSIONAL

Команды находятся в группе **Проставление размеров** вкладки **Применить**.
Этот инструмент доступен только если в базе данных операций был задан специальный код операции для проставления размеров и если активизированы свойства операции "Конструкция" (поле "B") и "Имя" (поле "N").
Инструмент проставления размеров позволяет добавлять *линии* измерения к программе. Это особые операции, не определяющие никакого выполнения, а предусматривающие только отображения конструкции при помощи прямой документации программы. Выбор отрезка для проставления размеров выполняется непосредственно мышью в графической зоне. Индикация по последовательности операций приводятся в зоне "Команды".
Можно делать выбор из 4 различных типов проставления размеров:

- **Горизонтальный**: вставляет горизонтальную линию размеров и значение размера.
- **Вертикальный**: вставляет вертикальную линию размеров и значение размера.
- **Горизонтальный + Вертикальный**: вставляет вертикальную линию размеров и соответствующий размер. Вставляет горизонтальную линию размеров и размер.
- **Диагональный**: вставляет диагональную линию размеров и значение размера.

Пример горизонтального проставления размеров



На рисунках представлена последовательность этапов, необходимых для завершения вставки:

1) выделяются две крайние точки линейного отрезка для расчета горизонтального проставления размеров (dX=114.2924)

2) задается вертикальное положение надписи проставления размеров

3) после подтверждения команды вставляется горизонтальный отрезок со стрелками на концах и надпись со значением отрезка (в данном случае 114.29)

Введенная здесь информация по проставлению размеров никоим образом не зависит от элементов, на которых было выполнено измерение (точки привязки и пр.). Эти элементы могут меняться или удаляться, при этом не происходит какого-либо изменения или автоматического удаления.

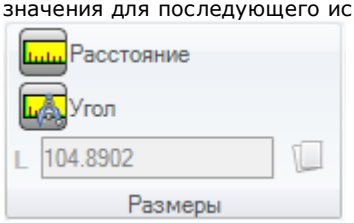
Размеры

Команды размеров находятся в группе **Размеры** вкладки **Применить.** Предусматриваются две специальные команды:

- **Расстояние:** измеряет линейное расстояние между двумя точками, определенными любым образом в графической зоне. Расстояние между двумя точками представлено линейным сегментом с указанием фактического расчетного расстояния.
- **Угол:** измеряет угол между вершиной и двумя линейными сегментами

Положение этих двух точек задается мышью непосредственно в графической зоне. Индикации приводятся в зоне команд. Из контекстуального меню можно подключить привязку к объектам или к сетке.
Может быть особенно интересно напомнить о возможности использования привязок:
- на Гранях;
- на Глубине;
- на объектах Закладки.

При подтверждении этой команды соответствующий размер приводится в меню, в группе Размеры вкладки Применить, как показано на рисунке. Показанная сбоку команда копирования выполняет копирование значения для последующего использования.



При линейном измерении (**Расстояние**) можно выбирать из измерений, определенных для двух положений:
- 3D: трехмерное расстояние
- dX, dY, dZ: расстояние вдоль координаты X/ Y/ Z. Если измерение было получено за счет использования указателя между различными гранями, расстояния относятся к абсолютной системе координат детали.

10.8 Общие инструменты программы

Это команды, имеющиеся в Общем виде, вызываемые в группе **Применить к детали** вкладки **Применить.** Все эти команды применяются ко всей программе и имеют эквивалент в инструментах, применимых в Виде грани.
При подтверждении комплексного инструмента программы отображается окно задания *Опций поиска*:
- **Соответствие на виде**: при подключении учитывает только отображенные операции (применяет активированные виды и фильтры изображения).

Рассмотрим подробно виды и примененные фильтры:
- поиск исключает следующие операции: логические, с активным полем C или с заблокированными свойствами (поле L, B,) или с недействительным рабочим кодом (т. е. операция, не имеющая соответствия в базе данных операций);
- с активным Видом выборов: принимает во внимание только выбранные операции;
- с активным Видом логических условий: принимает во внимание только операции, удовлетворяющие логическим условиям, включая исключения;

- с активным Видом фильтров уровней: принимает во внимание только операции, заданные с отображенным уровнем;
- с активным Видом специальных фильтров: принимает во внимание только операции, проверенные специальными фильтрами вида (поля B, O, K, K1; технология).

- **Применить к выбранным операциям**: при подключении рассматривает только выбранные операции (это поле подключено только при наличии выбранных операций). Подключение этой опции рассматривается только если не подключен пункт **Соответствие на виде**, который уже включает ее.

Применить технологию

Команда **Технология**  подобна инструменту Применить подготовку к профилю. Применяет технологию к точечным операциям или открытым профилям, т. е. без начальной подготовки или начинающихся с геометрической подготовки.
Тип задаваемой подготовки или точечного кода выбирается в окне, в котором предлагаются все операции выбранного типа, доступные в приложении.
Все технологические данные задаются в стандартном окне Технологическое задание. Нельзя задать какую-либо технологию открытым профилям или начинающимся с геометрической подготовки или с геометрическими точками, если они определены внутри сложной операции.

В точечной операции задание параметра "Диаметр" выполняется по следующим правилам:
- если операция точечного геометрического кода не имеет заданное значение диаметра, выполняется замена
- если операция точечного геометрического кода имеет заданное значение диаметра, замена не выполняется.

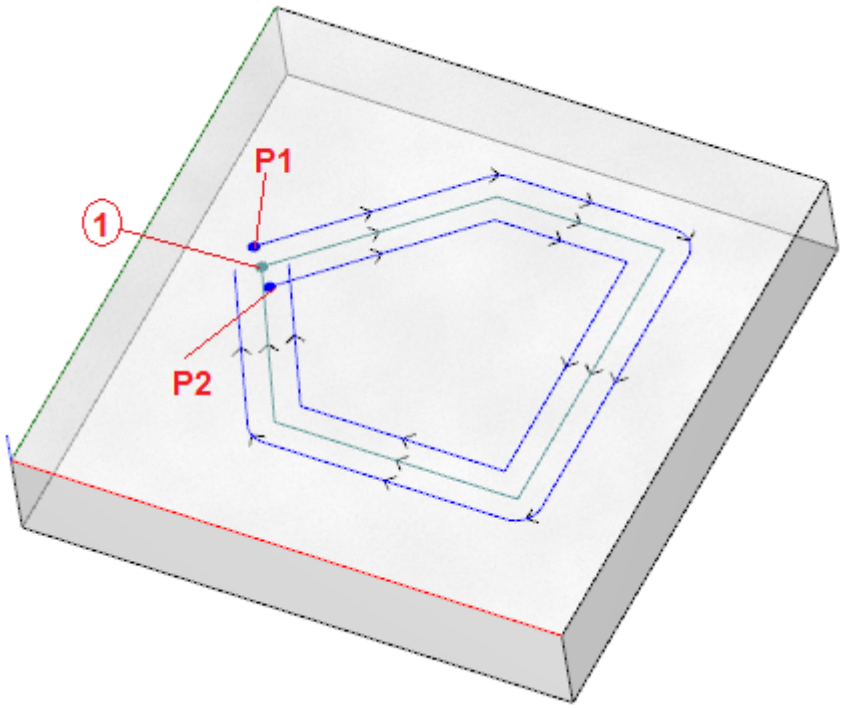
Преобразовать [мм]-[дюймы]

Эта команда **Преобразует [мм]-[дюйм]**  преобразует программу из [мм] в [дюймы] и наоборот.
Преобразование применяется только к информации, относящейся к положению или скорости.
Этот инструмент предназначен, в основном, для преобразования программ, импортируемых из других форматов (например, DXF) и написанных изначально с другими единицами измерения.
Эта команда может быть недоступна, а в случае ее необходимости требует правильного составления базы данных операций для правильного нахождения всех требующих изменения операций.
Эта команда не имеет эквивалента в Инструментах, применимых в Виде грани.

Этот инструмент предупреждает, может ли его применение затронуть параметрическое программирование: в этом случае можно выполнить преобразование только числовых форм, исключая параметрические.
Применение преобразования касается также разделов программы: размеры и единицы измерения, порядок выполнения, переменные “o” и/или “v” с заданным размером, геометрией моделирования, фиктивными гранями (добавленные геометрии и параметры).

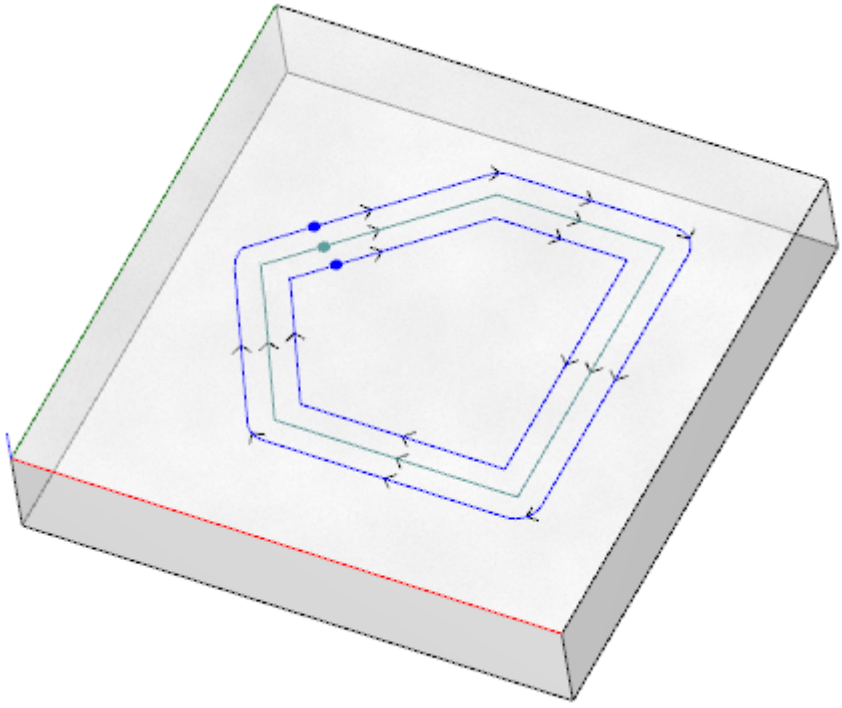
Подтвердить профили

Команда **Подтверждает профили**  анализирует точку подготовки замкнутых профилей и при необходимости перемещает ее для нейтрализации возможного применения коррекции на инструмент или даже просто входа инструмента в "неудобной" точке.
Рассмотренные ситуации соответствуют точкам начала ребристого профиля. Рассмотрим пример, показанный на рисунке:

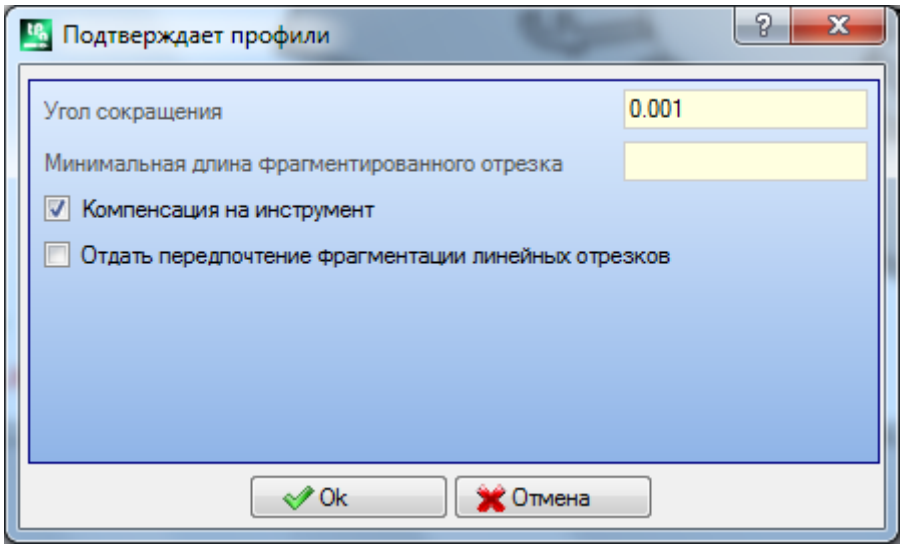


Центральный профиль соответствует запрограммированному профилю: подготовка показана как точка (1)
(P1) указывает подготовку профиля, полученную для левой коррекции
(P2) указывает подготовку профиля, полученную для правой коррекции.

Очевидно, что оба исправленных профиля имеют проблему:
(P1) больше не является замкнутым профилем
(P2) представляет пересечение крайних отрезков профиля.
В этом случае инструмент переместил бы подготовку на первый линейный отрезок, разбивая его на два отрезка, теперь при непрерывности касательной. На рисунке показана новая ситуация:



Критерии применения инструмента показаны в окне:



- **Угол сокращения**: задает значение углового допуска для распознавания условия непрерывности на исходной подготовке. Задает значение не выше 45° (при задании нулевого значения оценивает непрерывность касательной с допуском в 0.001°)
- **Минимальная длина разбитого отрезка**: задает минимальную длину для выбора отрезка профиля, на который перемещать подготовку
- **Компенсация на инструмент**: выбрать для учета габарита инструмента в оценке минимальной длины каждого отдельного отрезка профиля. В этом случае минимальная требуемая длина определяется равной диаметру * 3.0
- **Предпочесть фрагментацию линейных отрезков**: выбрать для запроса перемещения, если возможно, точки подготовки на линейный отрезок. В противном случае подготовка перемещается на первый отрезок, линию или дугу, имеющую минимальную длину.

Оценка отрезков на минимальной длине выполняется на плоскости грани (плоскость xy) и, в любом случае, учитывает как заданное значение (**Минимальная длина разбитого отрезка**), так и требуемое значение **Компенсации на инструмент**, с минимальным значением, зафиксированным на 50.0*эпсилон.
Инструмент исключает оценку отрезков профиля, заданных со сложными кодами или с дугами в плоскости #xy.
После определения отрезка, на который будет перемещаться подготовка, этот отрезок делится на два отрезка одинаковой длины.

Может произойти так, что профиль, заданный на отрезках, которые все являются маленькими, сделает применение инструмента невозможным.
Применение инструмента может определять утрату параметрических заданий

Применить сокращение к профилям

Команда **Сократить профили**  подобна инструменту Минимизировать профиль. Сокращает число отрезков, задавая критерии углового и/или линейного сокращения. Нельзя минимизировать профили, определенные внутри сложной операции.

Применить фрагментацию к профилям

Команда **Фрагментировать профили**  подобна инструменту Фрагментировать профиль. Фрагментирует отрезки профиля на отрезки максимальной заданной длины. Фрагментация касается только дуг с возможностью линеаризации фрагментированных отрезков. Эта команда не фрагментирует профили, определенные внутри сложной операции.
В диалоговом окне задаются параметры, которые уже рассматривались для инструмента Фрагментировать профиль, к которому и следует обращаться.

Применить соединение к профилям

Команда **Соединить профили**  подобна инструменту Соединить соседние профили

Для каждой соответствующей грани соединяются профили с проверкой геометрической непрерывности между точками начала и конца с возможностью оценки также **инверсии профилей**. Этот инструмент определяет исходную операцию соединения при первом возникновении операции setup (изолированной или нет) или отрезка профиля. Для того, чтобы расстояние соединения учитывалось также на компоненте глубины (ось Z), необходимо выбрать пункт **Применить в трехмерном виде.** Нельзя выполнить соединение профилей внутри сложной операции.
Процедура соединения может занять много времени в случае большого числа профилей, ввиду рекурсивного характера самой процедуры: для каждого запрограммированного профиля выполняется поиск возможных соединений со всеми другими профилями данной грани, и поиск заканчивается тогда, когда не находится ни одного возможного соединения. Рассматриваемый случаи могут соответствовать программам с тысячами профилей.
Выбор опции **Сократить поиск соответствий** позволяет ограничить область поиска соединений: для каждого профиля рассматриваются только профили, запрограммированные после него.

10.9 Общие трансформации программы



Это команды, имеющиеся в Общем виде, вызываемые в группе **Применить к детали** вкладки **Применить.** Все эти команды применяются ко всей программе и не имеют эквивалента в Инструментах, применимых в Виде грани. Для этих команд трансформации являются общими, потому что определяют изменение детали в комплексе. Трансформации при необходимости меняют также грань-деталь.

Общие условия для использования команд:
- должны быть подключены реальные грани
- при только одной подключенной реальной грани должно быть 1 либо 2
- грани 1 и 2, если они обе подключены, должны иметь плоскость XY грани, позиционированную таким же образом
- все подключенные боковые грани должны иметь ось Y в абсолютно вертикальном направлении с заданным направлением (абсолютная ось Z+ или Z-).

Особые условия для использования команд на текущей детали:
- деталь не должна задавать изменяемые геометрии (фиктивные или автоматические грани)
- при перенесении операции на другую грань эта грань должна быть подключена
- если требуется применить трансформацию поворота или зеркального отображения к операциям грани, трансформация должна быть применима ко всем операциям, при необходимости - с их разбивкой.

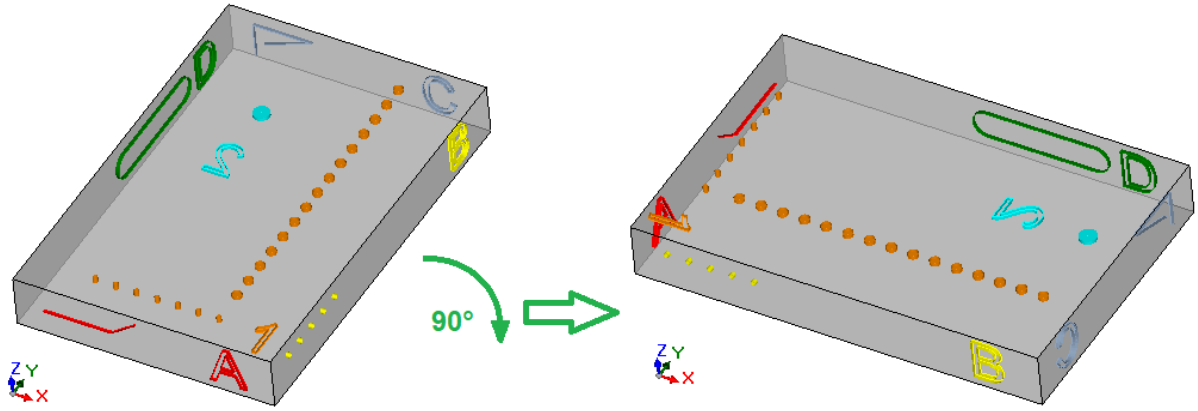
Выполнение команд связано с отменой списка команд, которые можно отменить или восстановить, также в случае отмены самих команд.
При отмене команд деталь восстанавливается в предыдущее состояние.

Повернуть деталь


Применяет поворот детали против часовой стрелки на 90°.

Особые условия для использования этой команды:
- должна предусматриваться операция, программирующая дугу в пространстве (код= 2110).

На рисунке представлен пример применения трансформации:

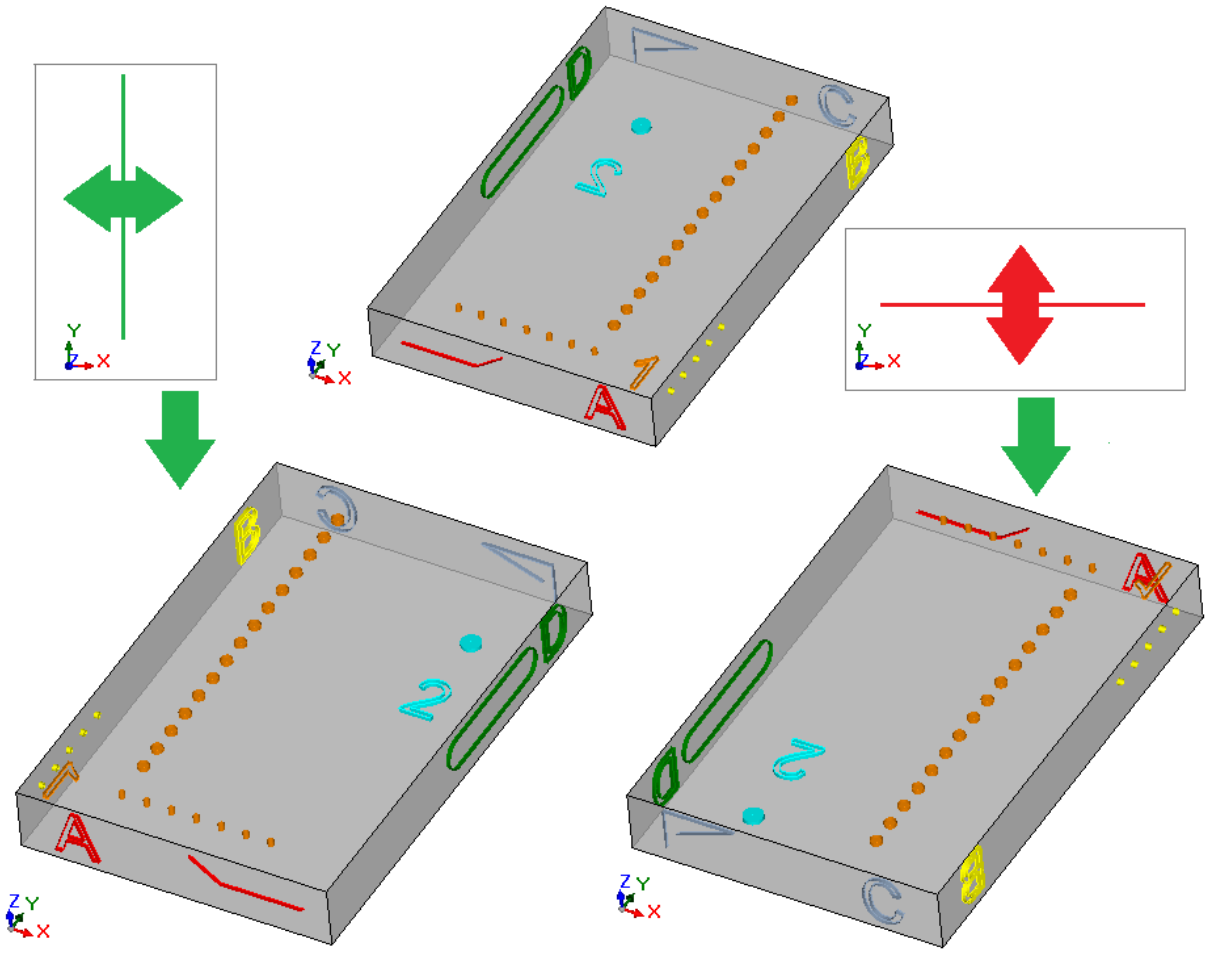


Последствия трансформации:
- операции граней 1 и 2 поворачиваются так, чтобы соблюдалось общее вращение 90° против часовой стрелки
- операции грани 4 переходят грани 3
- операции грани 5 переходят грани 4
- операции грани 6 переходят грани 5
- операции грани 3 переходят грани 6
- где необходимо, на боковых гранях применяется общая трансформация зеркального отображения по длине граней
- в размерах детали длина становится высотой, а высота - длиной.

Зеркальное отображение детали

 Применяет зеркальное отображение детали относительно вертикальной или горизонтальной оси симметрии.

На рисунке представлен пример применения трансформации:

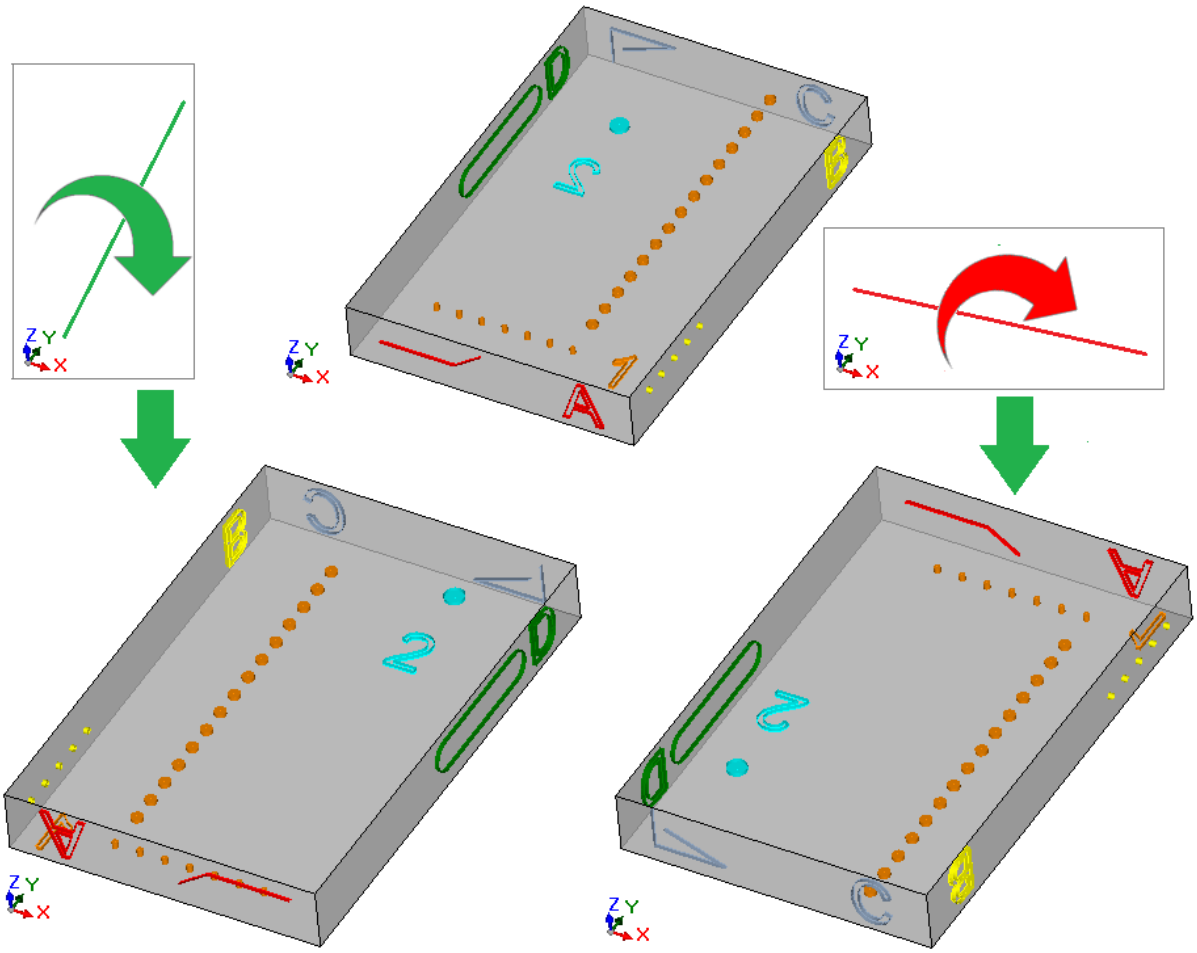


Последствия трансформации:
- к операциям граней 1 и 2 применяется общая трансформация зеркального отображения по вертикальной или горизонтальной оси симметрии, как требуется
- происходит обмен операций между парами боковых граней (4 и 6) либо (3 или 5), как необходимо
- где необходимо, на боковых гранях применяется общая трансформация зеркального отображения по длине граней.

Перевернуть деталь

 Применяет переворачивание детали относительно вертикальной или горизонтальной оси симметрии.

На рисунке представлен пример применения трансформации:



Последствия трансформации:

- происходит обмен операциями между парами 1 и 2
- к операциям граней 1 и 2 затем применяется общая трансформация зеркального отображения по вертикальной или горизонтальной оси симметрии, как требуется
- происходит обмен операций между парами боковых граней (4 и 6) либо (3 или 5), как необходимо
- где необходимо, на боковых гранях применяется общая трансформация зеркального отображения по длине и/или высоте граней.

11 Параметрическое программирование

11.1 Введение

Задания программы обычно допускают параметрическое задание.
Рассмотрим, например, переменные "o" и "v". Это числовые переменные, в поле задания которых обычно может задаваться число или числовое выражение
Большую специализацию имеют переменные "r". Тип переменной r не является фиксированным, а может выбираться из двух числовых (двойная и целая) и одного нечислового типа (строка).
Тип "Двойная" конфигурирует переменную "r" так же как, переменную "o" или "v" (для которых задание типа автоматическое). Поле задания обычно может задавать число или числовое выражение, а расчетное значение сохраняет дробную часть.
В случае целого типа поле задания может задавать число или числовое выражение и в расчетном значении дробная часть отбрасывается.
Зададим, например, для переменной r выражение: "1000/3":

- если переменная - двойного типа, расчетное значение = 333.333333
- если переменная - целого типа, расчетное значение = 333.

В случае типа строки поле задания задает, как правило, буквенно-цифровое выражение и значение, присваиваемое переменной, также является строкой. Типичное использование строки - задание имени подпрограммы или текста. Примеры задания: "porte\prg1.abc", "Ciao".

Для параметров операции действительны те же положения, что и для переменных "r": действительны как два числовых типа (двойная и целая), так и нецифровой (строка) тип переменных. Однако на этапе программирования выбор типа прозрачен, т. к. задан в задании базы данных операций.

11.2 Переменные и параметры числового типа

Числовое выражение - это выражение, которое может оцениваться как число. Элементы выражения могут включать любое сочетание ключевых слов (функции, используемые в параметрическом программировании), переменных (например, размеры детали), констант (например, пи) и операторов (например, +, -, *, /,|), результатом которых является число.
Числовое выражение должно задаваться:

- строчными буквами;
- могут использоваться символы, заключенные в ' ' (пробел) и '}' (дробные значения между 32 и 125)
- использование пробелов ограничивается функциями или переменными аргументами, относящимися к строкам;
- максимальное число допустимых символов - 100.

Примеры числового выражения:

- ”20”: это выражение может решаться напрямую. Оно напрямую задает числовое выражение
- ” (100+32)/2”: использует цифры, математические операторы, скобки
- ”r27+100”: использует цифры, переменные, математические операторы
- ”sqrt[r27+r15]-r5”: использует переменные, математические операторы, математическую функцию и аргумент.

Значение выражений интуитивно. Рассмотрим поэтапно, как оценивается каждое выражение:

- ”(100+32)/2”=(132)/2=132/2=66.0
- (значение r27=50) ”r27+100”=50+100=150.0
- (значение: r27=50, r15=30, r5=-5) ”sqrt[r27+r15]-r5”= sqrt[50+30]-(-5)=sqrt[80]-(-5)=9.944271-(-5) = 9.944271+5=14.944271

Приоритет операторов
Если выражение содержит различные операторы, каждая часть оценивается и решается в определенном порядке, который называется "приоритет операторов".
Арифметические операторы и логические операторы оцениваются на основе порядка приоритета, указанного в приведенном ниже списке:

- Умножение (*), деление (/, #), модуль (%), корректировка шага (?) и логические операторы (&, |);
- Прибавление и вычитание (+, –).

Если в выражении имеются операторы с одним приоритетом (например, умножение и деление), каждая операция оценивается в порядке ее отображения слева направо. Это касается и прибавления и вычитания в одном выражении.
За счет использования круглых скобок можно проигнорировать порядок приоритета и определить, что некоторые части выражения должны оцениваться раньше других. Максимальный предел встроенных скобок определяется только максимальной разрешенной длиной строки (100 символов). Операции в круглых скобках всегда имеют преимущество над остальными. В круглых скобках должен соблюдаться обычный приоритет между операторами.

Примеры выражений:
“2+3*4”: выполняет вначале умножение, а затем сложение. Поэтому: “2+12”=14

“(2+3)*4”: использование круглых скобок меняет результат. Поэтому: “(5)*4”=20.

11.3 Функции

Использование функций позволяет выполнять более сложную оценку по сравнению с той, которая разрешена операторами. Примером функции может быть "sqrt[r27+r15]-r5", в которой используется математическая функция sqrt, вычисляющая квадратный корень аргумента.
Функции делятся на две группы:

- с одним аргументом: примером является функция sqrt;
- с несколькими аргументами: примером является функция pown;

Функции с одним аргументом могут использоваться с двумя формализмами:

- числовым: аргумент - положительное число. Например, "sqrt25": аргумент (25) записан непосредственно после имени функции;
- нечисловым: аргумент - отрицательное число (например: -25) или выражен в параметрической форме (например, "r25", "100-32") . Например, "sqrt[r25]": теперь аргумент заключен в квадратные скобки.

Нечисловой формализм обязателен также для некоторых особых функций с одним аргументом, относящихся к группе Ссылки на переменные детали.

Функции с несколькими аргументами могут использовать только нечисловой формализм со следующим синтаксисом: имя[op1;op2;…;opn]:

- имя - имя функции. Пример: **pown**
- […] ограничивают операнды функции
- op1 первый аргумент
- ; разделитель между двумя аргументами
- op2 второй аргумент
- .
- opn последний аргумент.

Число аргументов функции с несколькими аргументами может быть фиксированным или переменным. В последующих параграфах подробно рассматриваются все функции с указанием, в частности, требуемого числа аргументов, и какие должны задаваться и какие нет.
Способ указания синтаксиса функции значим для интерпретации числа и использования аргументов и отражает общий формализм. Рассмотрим несколько примеров:

- pown[nb;ne] функция с 2 аргументами: оба должны быть заданы
- min[n1;…;n30] функция с переменным числом аргументов: допустимое число - от 1 до 30;
- case[nc;nc1:nv1;nc2:nv2;...;nvdef]функция с переменным числом аргументов: первые 3 (nc;nc1:nv1;nc2:nv2) должны задаваться, затем следует некоторое число необязательных аргументов (….;), а последний заданный (nvdef) имеет особую интерпретацию;
- prmac[(nm); nkind;(vdef)] 1-й и 3-й параметр заключены в круглые скобки (nm), (vdef): это означает, что аргумент может задаваться пустым (в этом случае функция применяет значение по умолчанию). Ввиду того, что vdef является последним аргументом функции, его можно совсем не задавать.

Не действует какой-либо предел соединения между функциями: он обуславливается только максимальной разрешенной длиной строки (100 символов).

11.4 Переменные и параметры типа строки

Примеры буквенно-цифрового выражения:

- ”porte\prg1.abc”: это выражение может решаться напрямую со значением, совпадающим со (строка)
”porte*r1.abc”: использует переменную (r1)
”qx=*r1.*r2”: использует переменные (r1, r2)
”*pr[r45+5]”: использует переменную (r45), функцию со ссылкой на переменную.

Значение этих выражений менее интуитивно по сравнению с числовыми выражениями. Рассмотрим поэтапно, как оценивается каждое выражение:

- (r1 - это переменная строка со значением ="prg1") -> “porte*r1.abc”= “porte\prg1.abc”

· (r1 и r2 - числовые переменные со значением =123 и 45) -> ”qx=*r1.*r2= ”qx=123.45”
· (r45 - числовая переменная со значением =2)-> “*pr[r45+5]” = “*pr[2+5]” = “*pr[7]” -> (r7 - переменная строка со значением ="prg1") ->="prg1".

Буквенно-цифровое выражение может задаваться:

- в т. ч. прописными буквами;
- разрешается использовать пробелы (в любом случае, должны удаляться пробелы в начале и в конце);
- могут всегда использоваться символы, заключенные в ' ' (пробел) и '}' (дробные значения между 32 и 125). Это символы общего отображения, не зависящие от международных настроек операционной

системы: цифры (0-9), строчные буквы (a-z), заглавные буквы (A-Z), символы пунктуации (напр., .,;:? !), арифметические операторы (напр.,+ - * /<>#%), скобки (напр., []{}())

- не существует ограничений по вводимым символам, включая возможность использовать все особые символы различных системных настроек помимо символов Unicode (т. е. японских, китайских, арабских и пр. символов).

В то время как формализм числового выражения полностью отвечает общим критериям решения выражения, буквенно-цифровое выражение интерпретируется на основе нескольких ранее определенных формализмов (частично указанных и решенных выше), которые должны обязательно соблюдаться::

- **"porte*r1.abc"**
  в этом формализме имеют параметрическую интерпретацию выражения "*rn", где n указывает переменную "r" к использованию (n=0-299).
  В примере:
  - если r1 имеет типологию строки, значение (строка) r1 заменяется выражением "*r1", как указано выше;
  - если же r1 имеет числовую типологию, строка, соответствующая целой части значения r1, заменяется выражением "*r1";
  - если r1 не задана, строка "0" заменяется выражением "*r1".

  Не устанавливается пределов числу замен. Таким образом, являются действительными следующие задания:
  "двери*r1.*r3"
  "abc*r5*r1.*r3".
  строка "abc*r500" не распознает никакой параметрической формы.

  Можно также извлечь часть строки, адресованную выражением "*rn".
  Синтаксис: "…*rn[ni;nc]…", где:
  - n = индекс (только числовой) переменной r (например, 5 для r5);
  - ni = начальное положение, с которого читать строку, заданную для r5 (значима с 1). Может задаваться:
    - числовым (например, ni=3);
    - с переменной r числового типа (например, ni=r2);
    - с переменной j (например, ni=j5);
    - с переменной $, если в тексте макропрограммы, (например, ni=$0);
  - nc = число считанных символов, начиная с ni (дополнительно). Может задаваться:
    - числовым (например, ni=3);
    - с переменной r числового типа (например, ni=r2);
    - с переменной j (например, ni=j5);
    - с переменной $, если в тексте макропрограммы, (например, ni=$0).

    Кроме того, реализовано также использование символических имен переменных r, в двух формах:
  - "…….*r\name\…." **ВНИМАНИЕ**: символическое имя должно завершаться символом '\'
  - "…….*r\name[ni;nc]…." **ВНИМАНИЕ**: здесь символическое имя должно завершатьтся символом '['

  Пример: "створки\ *r5[3;1].cnc"
  пусть r5 - переменная строка, заданная = "abcdef";
  ni=3: считывает r5 с третьего символа;
  nc=1: считывает 1 символ;
  -> получается решение "ante\c.cnc".

  Пример: "створки\ *r5[3].cnc"
  пусть r5 - переменная строка, заданная = "abcdef";
  ni=3: считывает r5 с третьего символа;
  nc не задан: не усекает строку;
  ->получается решение "ante\cdef.cnc".

  Пример: "створки\ *r5.cnc"
  пусть r5 - переменная строка, заданная = "abcdef";
  ->получается решение "ante\abcdef.cnc".

  Пример: "створки\ *r\str1\.cnc"
  пусть r5 - переменная строка, заданная = "abcdef", с именем ="str1";
  ->получается решение "ante\abcdef.cnc".

  Пример: "створки\ *r\pippo[3].cnc"
  пусть r5 - переменная строка, заданная = "abcdef", с именем ="pippo";
  ni=3: считывает r5 с третьего символа;
  nc не задан: не усекает строку;
  ->получается решение "ante\cdef.cnc".

В случае использования переменной числовой r, также можно запросить форматирование десятичной части.
Синтаксис: "…*rn[**d**;nc]…" где:
- n = индекс (только числовой) переменной r (например, 5 для r5);
- d = задать буква 'd'
- nc= число десятичных цифр (цифры исключаются, если не являются значимыми)

Пример: " *r5[d;4]"
пусть r5 числовая переменная со значением = 123.4006;
nc=4: задает первые 4 десятичного цифра ->получается решение "123.4006".
если (nc=3): задает первые 3 десятичного цифра ->получается решение "123.4" (последние 2 цифры исключаются, поскольку они не являются значимыми).

- **"*pr[r45]"**
этот второй формализм жестче предыдущего. Он интерпретирует только форму "*pr[…..]", в которой аргумент функции pr[..] может задавать любое числовое выражение.
Решение аргумента функции pr[..] имеет в результате числовое значение целого типа (n), которое, в свою очередь, определяет переменную rn (n - индекс переменной).
Обычно наблюдается случай, когда rn имеет тип строки, из этого следует, что значение (строка) rn задает значение строки буквенно-цифрового выражения.
Если же rn имеет числовой тип, строка, соответствующая целой части значения rn, задает значение строки буквенно-цифрового выражения. Рассмотрим пример:
r3 - переменная числового формата = 250.8
r5 - переменная типа строки = *pr[3]= "250";
Если rn не задана, строка "0" заменяется выражением *pr[…..]".

- **"*p[…]"**
формализм, подобный предыдущему, в котором аргумент функции p[..] может задавать любое числовое выражение.
Решение аргумента имеет в результате числовое значение (n): строка, соответствующая целой части значения (n), задает значение строки выражения.
Пример: "*p[1024/6]"
1024/6=170.6666 а результат - строка "170".

- **" *j1.*j2 "**
- **" *$1.*$2 "**
в этом формализме имеют параметрическую интерпретацию выражения "*jn" (и "*$n"), где n указывает переменную "j" (или "$") к использованию. Действительны положения, уже приведенные для аналогичного формализма, рассмотренного для переменных типа "r": в этом случае используются только цифровые переменные, поэтому строка соответствует целой части значения переменной.
В выражении типа строки можно использовать одновременно синтаксис всех реализованных переменных.
Пример: "створки\ *r5*j1.cnc"
пусть r5 - переменная строка, заданная = "abcdef";
пусть j1=4
->получается решение "створки\abcdef4.cnc".

"*geo[sub;..]", "*geo[param;..]", "*geo[lparam;..]"
эти формализмы интерпретируют только форму "*geo[…..]", где аргумент функции geo[..] может задать любое числовое выражение. Решение используемой функции geo[..], приводит к значению информации или параметру операции. В частности:
- в случае информации или параметра типа строки: соответствует результату функции
- в случае числовой типологии: результатом является строка, которая соответствует полной части значения.

Подробнее об использовании этих функций см. относительный абзац.

11.5 Числовые форматы специального применения

Рассмотрим форму специального параметрического префикса, даже если он не используется напрямую в программировании, он может создаваться при применении инструментов (вращение, зеркальное отображение,..).
Это форма программирования "a;…..", которую можно задать в числовых параметрах операции, имеющих значение координат.
Примеры правильного задания:
"a;500" значение параметра числовое
"a;l/2" значение параметра параметрическое.

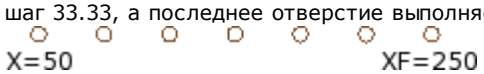
Форма "a;….." указывает, что соответствующая координата - абсолютного программирования.

Рассмотрим, например, операцию дуги: координаты центра интерпретируются относительно по сравнению с точкой начала дуги. Можно принудительно интерпретировать координаты абсолютно с использованием формы "a;….."

11.6 Термы выражения

Операторы

Арифметические

+	Сумма операндов. Пример: 100.6 + 7 = 107.6
-	Вычитание операндов. Пример: 100.6 - 7 = 93.6
*	Умножение операндов. Пример: 100 * 7 = 700
/	Деление между операндами. Знаменатель не может быть равен нулю. Ситуация ошибки: 125: знаменатель равен нулю; Пример: 100 / 7 = 14.285714
%	Модуль деления между операндами. Знаменатель не может быть равен нулю. Пример: 100 % 7 = 2 Процедура вычисления результата: • выполняет деление (100 / 7) = 14.285714 • отделяет дробную часть результата: (14.285714 – 14) = 0.285714 • умножает на делитель: 0.285714 * 7 = 2
#	Целое число деления между операндами. Знаменатель не может быть равен нулю. Пример: 100 # 7 = 14 Процедура вычисления результата: • выполняет деление: 100 / 7 = 14.285714 • отделяет целую часть результат: целое число (14.2857) = 14
?	Корректировка шага между операндами. Знаменатель не может быть равен нулю. Пример: 100 ? 7 = 7.14285 Процедура вычисления результата: • выполняет деление (100 / 7) = 14.285714 • отделяет дробную часть результата: (14.285714 – 14) = 0.285714 • умножает на делитель: 0.285714 * 7 = 2 (=100 % 7) • делит модуль на целое число от деления: 2 / 14 = 0.14285 • прибавляет к делителю: 7+ 0.14285 =7.14285. Первые три пункта вычисляют модуль. Таким образом, операнд ? возвращает делитель, измененный таким образом, чтобы получить целый результат деления. В случае, если результат деления будет меньше 1, эта операция возвращает делимое. Пример: 10 ? 15 = 10 {дает: 10/15= 0.6666} Пример: программирование подгонки X в случае, когда конечная координата X совпадает с положением сверления и необходимо скорректировать расстояние между отверстиями. Предположим, что Начальный x=50, Конечный x=250, Шаг=200?32. В результате получаем шаг 33.33, а последнее отверстие выполняется на координате X = 250. 

Логические

Должны считаться операторами продвинутого программирования.

&	Бинарная операция И побитового сравнения целой части операндов с отбрасыванием дробной части результата. Пример: 10.456 & 3.56 = 10 & 3 = 2 Рассматривается представление в двоичной системе 10 и 3: 10 = 1 0 **1** 0 and 3 = 0 0 1 1 0 0 1 0 = 2 дробная часть В битовом представлении результата сохраняется настройка на 1 битов, которые находятся на 1 в обоих операндах.

|	Бинарная операция ИЛИ побитового сравнения целой части операндов с отбрасыванием дробной части результата. Пример: 10.456 | 3.56 = 10 | 3 = 11 Рассматривается представление в двоичной системе 10 и 3: 10 = **1** 0 **1** 0 or 3 = 0 0 1 1 1 0 1 1 = 11 дробная часть В битовом представлении результата сохраняется настройка на 1 битов, которые находятся на 1 в одном или обоих операндах.

Скобки, разделители

(..)	Уровень скобок: могут вставляться без ограничений. Решение выражения вначале решает более внутренние скобки. Пример: "12*((r0+r3)*sqrt[12])" • решает: (r0+r3) {допустим =10} • решает: (10*sqrt[12])=34.64 • решает: 12*34.64=415.6921
[..]	Ограничители для аргументов функции, используемые для: • параметрического или отрицательного аргумента функции; • задания функции нескольким операндам. Примеры: sqrt[r12] функция с одним аргументом с параметрическим аргументом sin[-45] функция с одним аргументом с отрицательным аргументом min[r12;1;67] функция с несколькими аргументами
. ,	Разделитель между целой и дробной частью числового аргумента. Должен использоваться только один разделитель и он указан в конфигурации TpaCAD Примеры: 128.6 .965
;	Разделитель между аргументами функции с несколькими аргументами. Пример: pown[5;2]
"..."	Прямое задание строки (например, для функции : strcmp). Распознает также символ пробела. Пример: strcmp[5; "pippo"] оценивает r5 и сравнивает со строкой "pippo" strcmp[5; "ciao …"] оценивает r5 и сравнивает со строкой "ciao …"

Переменные аргументы

Общие аргументы

pi	при (π = 3.1415..). Использование: всегда.
eps	Эпсилон линейного положения, принимает значение, зависящее от единиц измерения программы: • 0.001 для мм • 0 001/25.4 для дюймов умноженное на коэффициент Множитель Эпсилон (задано при конфигурации TpaCAD) Использование: всегда. Пример: • при: программе в [мм] множитель = 10 -> eps равен: 0.001*10 = 0.01 • при: программе в [дюймах] множитель = 10 -> eps равен: 0.001*10/25.4 = 0,0003937
cnq	Коэффициент пересчета линейного положения, принимает значение, зависящее от единиц измерения программы: • 1 для мм • 1/25.4 для дюймов Использование: всегда. Переменный аргумент cnq используется при написании подпрограмм и/или макропрограмм для сравнения и/или прямого задания значений координат или скорости, в случаях, когда можно предвидеть, что подпрограмма (или макропрограмма) может использоваться в программе, написанной независимо от единиц измерения [мм] или [дюймы].

	Пример: • пишется подпрограмма (ОДИН) в единицах [мм] • эта подпрограмма выполняет операции сверления, расположенные относительно друг друга со сдвигом x • расстояние между отверстиями задается в переменной r (перезадаваемой): предположим, r3 • однако необходимо соблюдать минимальное расстояние в 20 мм: очевиден конфликт r3 с номером 20. Проблем не будет, даже если программа, использующая подпрограмму ОДИН, написана в [мм]. Однако если программа, использующая подпрограмму ОДИН, написана в [дюймах]: r3 теперь задан в [дюймах]. В этом случае больше нельзя сравнивать r3 непосредственно с 20. Оба случая действительны, если сравнение выполнялось с "20*cnq", которое равно: • 20 если программа использует единицу измерения [мм] • 20/25.4 = 0.7874 если программа использует единицу измерения [дюймы]
cnf	Коэффициент пересчета линейной скорости, принимает значение в зависимости от единицы измерения программы: • 1 для мм • 0.6561 для дюймов, со скоростью, выраженной в: м/мин или дюймы/сек • $6.561*10^{-4}$ для дюймов, со скоростью, выраженной в: мм/мин или дюймы/сек • 1000/25.4 для дюймов, со скоростью, выраженной в: м/мин или дюймы/мин • 1/25.4 для дюймов, со скоростью, выраженной в: мм/мин или дюймы/мин Использование: всегда. Использование аргумента cnf подобно cnq, только здесь пересчет относится к значению линейной скорости.
l h s	Размеры детали: "l" - длина, "h" - высота, "s" - ширина. Значения аргументов все также соответствуют размерам программируемой детали даже если, например, они используются в вызове подпрограммы. Использование: всегда.
face	Номер текущей грани. Различаются следующие ситуации использования: • в параметре операции: возвращает номер грани применения операции (значения от 1 до 6, если это реальная грань; от 7 до 99, если это фиктивная грань; от 101 до 500, если это автоматическая грань): • если грань реальная - это клиентский номер грани • если грань-деталь - соответствует полю F (при применении на автоматической грани возвращает номер, присвоенный автоматической грани, например, 120) • в списке переменных (o, v, r), задание фиктивных граней или клиентского раздела: возвращает значение -1. Значение аргумента все также соответствует текущей грани программируемой детали даже если, например, оно используется в вызове подпрограммы. Использование: всегда.
face0	Этот аргумент указывает, находимся ли мы в программировании грани-детали: • 1 условие выполняется • в противном случае получается 0. Использование: всегда.
faceauto	Этот аргумент возвращает номер автоматической грани. Это значение значимо только при программировании грани-детали: в случае автоматической грани, заданной ранее, возвращает значение от 101 до 500. Использование: всегда.
lf hf sf	Размеры текущей грани: "lf" - длина, "hf" - высота, "sf" - толщина. Можно использовать альтернативные моно-буквальные формы: «x» - это длина, «y» - это высота, «z» - это толщина. Различаются следующие ситуации использования: • в параметре операции: возвращают соответствующий размер грани применения операции: это размеры грани, соответствующие значению "face".

<table>
<tr><td></td><td>
• в списке переменных (o, v, r), задание фиктивных граней или клиентского раздела: возвращают соответствующий размер детали ("lf" означает "l", "hf" означает "h", "sf" означает "s").
Значения аргументов все также соответствуют размерам программируемой текущей грани, даже если, например, они используются в вызове подпрограммы.
Использование: всегда.
Пример:
• задаются размеры детали как l*h*s=1000*450*18
• посмотрим значения для трех переменных аргументов для шести граней детали
<table>
<tr><th></th><th>Грань 1</th><th>Грань 2</th><th>Грань 3</th><th>Грань 4</th><th>Грань 5</th><th>Грань 6</th></tr>
<tr><td>lf</td><td>1000</td><td>1000</td><td>1000</td><td>450</td><td>1000</td><td>450</td></tr>
<tr><td>hf</td><td>450</td><td>450</td><td>18</td><td>18</td><td>18</td><td>18</td></tr>
<tr><td>sf</td><td>18</td><td>18</td><td>450</td><td>1000</td><td>450</td><td>1000</td></tr>
</table>
</td></tr>
<tr><td>prgt</td><td>Тип детали: 0=программа, 1=подпрограмма, 2=макропрограмма.
Значение аргумента все также соответствует уровню доступа детали в программировании даже при использовании, например, в вызове подпрограммы.
Использование: всегда.</td></tr>
<tr><td>prgrd</td><td>Уровень доступа к детали: 0=уровень оператора, 1=уровень монтажника, 2=уровень производителя.
Значение аргумента все также соответствует уровню доступа детали в программировании даже при использовании, например, в вызове подпрограммы.
Использование: всегда.</td></tr>
<tr><td>prgwr</td><td>Уровень изменения детали: 0=уровень оператора, 1=уровень монтажника, 2=уровень производителя.
Значение аргумента все также соответствует уровню изменения детали в программировании даже при использовании, например, в вызове подпрограммы.
Использование: всегда.</td></tr>
<tr><td>prgnum</td><td>Номер по порядку последней запрограммированной операции в списке текущей грани, не комментария.
В списке переменных (o, v, r), задание фиктивных граней или клиентского раздела равно 0.

Использование: всегда.</td></tr>
<tr><td>'ch'</td><td>Заменяет дробное числовое значение, соответствующее символу ch. Правильные коды: от 32 (' ') до 125 ('}').). Прописные буквы преобразуются в строчные.
Использование: всегда.
Пример: 120+'a'=120+97=217
'a'-' ' = 97-32=65</td></tr>
</table>

Режим выполнения

<table>
<tr><td>prgrun</td><td>Активная рабочая среда: 0 = редактирование (среда ТраCAD); 1 = выполнение.
Аргумент prgrun может использоваться для дифференцирования клиентских сигналов ошибки:
• подключать сигналы ошибки применительно к технологическим соответствиям только на этапе выполнения;
• в крайнем случае - полное дифференцирование выполнения программы (как правило, макропрограммы) в двух средах.
Использование: всегда.</td></tr>
<tr><td>prgdraw</td><td>Активная рабочая среда: если prgrun=1, позволяет различать различные ситуации выполнения. В частности, положительное значение (1) указывает, что активен режим “просмотра графического представления”.
Типичное применение аргумента позволяет дифференцировать компиляцию программы с учетом графических и исполнительных потребностей.
Использование: всегда.</td></tr>
<tr><td>prgn</td><td>Флаг нормального исполнения: 1 = нормальное выполнение; 0 = другое выполнение.
Соответствует полю Исполнение как задано в Режиме выполнения.
Аргумент prgn позволяет дифференцировать задание программы исходя из того, как она направляется на выполнение. Примеры: выполнять или нет некоторые операции, задавать ли направления выполнения прорези.
Использование: всегда.</td></tr>
</table>

prgx	Флаг зеркального исполнения: 1 = зеркальное исполнение; 0 = другое выполнение. Соответствует полю *Исполнение* как задано в Режиме выполнения. Аргумент prgx позволяет дифференцировать задание программы исходя из того, как она направляется на выполнение. Использование: всегда.
prgy	Флаг зеркального исполнения Y: 1 = зеркальное исполнение Y; 0 = другое исполнение. Соответствует полю *Исполнение* как задано в Режиме выполнения. Аргумент prgy позволяет дифференцировать задание программы исходя из того, как она направляется на выполнение. Использование: всегда.
prgxy	Флаг зеркального исполнения XY: 1 = зеркальное исполнение XY; 0 = другое выполнение. Соответствует полю *Исполнение* как задано в Режиме выполнения. Аргумент prgxy позволяет дифференцировать задание программы исходя из того, как она направляется на выполнение. Использование: всегда.
prarea	Зона выполнения. Соответствует полю *Рабочая зона*, как оно задано в Режиме выполнения. Аргумент prarea позволяет дифференцировать задание программы исходя из того, как она направляется на выполнение. Использование: всегда.
prqx prqy prqz	Отсчет X, Y, Z в зоне выполнения. Соответствует полям *Сдвиги упора в рабочей зоне* как задано в Режиме выполнения. Аргументы prqx/y/z позволяют дифференцировать задание программы исходя из того, как она направляется на выполнение. Использование: всегда.
prun1 prun2 prun3 prun4 prun5 prun6 prun7 prun8	Параметры, добавляемые при выполнении. В режиме редактирования (prgrun = 0): параметры всегда имеют значение 0. В режиме выполнения (prgrun = 1) им могут задаваться значимые значения для обработки и выполнения программы. Использование: всегда.

Задания среды

Эти аргументы могут использоваться при написании подпрограмм и/или макропрограмм, для сравнений и/или прямых заданий значений (координат, осей вращения), в случаях, когда можно предусмотреть, что подпрограмма (или макропрограмма) может использоваться в непредварительно определенной конфигурации.

Должны считаться аргументами продвинутого программирования.

sysface	Геометрия граней 0= грани в прозрачном представлении 1= клиентские системы. Использование: всегда.
sysquad	Рабочий квадрант: значение от 1 до 4. Использование: всегда.
sysz	Указывает направление оси Z при программировании на грани: 0= если значения отрицательны на входе в деталь; 1= если значения положительны на входе в деталь. Использование: всегда.
sysxz	Указывает тип дуги, запрограммированной на плоскости XZ грани: 0 = если ось X является превалентной для этой плоскости (плоскости XZ); 1 = если ось Z является превалентной для этой плоскости (плоскости XZ). Использование: всегда.
sysbeta	Указывает вращение, предусмотренное для оси горизонтального вращения: 0 = если положительно в сторону X+ абсолютной системы координат 1 = если положительно в сторону X- абсолютной системы координат 2 = если вращение осуществляется вокруг оси X (положительно в сторону Y- абсолютной системы координат). Использование: всегда.

sysfeed	Указывает единицу измерения программирования линейной скорости: • в программе с единицами [мм] можно выбрать: 0=[м/мин.] 1=[мм/мин.] • в программе с единицами [дюймы] можно выбрать: 0=[дюймы/с] 1=[дюймы/мин.]
sysline syschord	Аргумент sysline указывает критерий, заданный для фрагментации дуг; 0 = дуги фрагментируются в зависимости от заданной длины 1 = применен критерий ошибки хорды. Аргумент syschord указывает: • длину фрагментации дуги (в первом случае) • ошибку хорды (во втором случае) значение syschord всегда преобразуется в единицу измерения программы. Использование: всегда.
syskey	Этот аргумент указывает, выполняется ли условие наличия ключа в режиме Professional: • 1 условие выполняется • 0 в противном случае (ключ в базовом или демонстрационном режиме) Использование: всегда.
sysname	Этот аргумент указывает, подключено ли управление поля Имя операции: • 1 условие выполняется • 0 в противном случае (поле недоступно) Использование: всегда

Переменные для детали

o0-o15 o\name	Переменные "o" детали. Различаются различные ситуации использования: • в тексте программы или подпрограммы: используются только заданные реализованные переменные (может быть число меньше 16, в крайнем случае - никакое); • в тексте макрокоманды: используются всегда в количестве 16; Вторая указанная форма соответствует символическому формализму, где: • неизменяемая часть "o\"; • переменная часть ("имя"): символическое имя, заданное для переменной. Возвращаемые значения все также соответствуют переменным программируемой детали, даже если, например, они используются в вызове подпрограммы. Они не используются: • при задании переменной 'o', 'v'; • при задании клиентских функций. Ситуации ошибки: • 114: использование в неправильном контексте; • 121: неправильный индекс переменной "o"
v0-v15 v\name	Переменные "v" детали. Различаются различные ситуации использования: • в тексте программы или подпрограммы: используются только заданные реализованные переменные (может быть число меньше 16, в крайнем случае - никакое); • в тексте макрокоманды: используются всегда в количестве 16; Вторая указанная форма соответствует символическому формализму, где: • неизменяемая часть "v\"; • переменная часть ("имя"): символическое имя, заданное для переменной. Возвращаемые значения все также соответствуют переменным программируемой детали, даже если, например, они используются в вызове подпрограммы. Не может использоваться: • при задании переменной 'o', 'v'; • при задании клиентских функций. Ситуации ошибки:

	• 113: использование в неправильном контексте; • 120: неправильный индекс переменной "v".

r0-r299 r\name	Переменные "r" детали. Все также используются в количестве 300: незаданная переменная все равно имеет числовое значение 0.0. Вторая указанная форма соответствует символическому формализму, признанному при задании переменной или числового параметра: • неизменяемая часть "r\"; • переменная часть ("имя"): символическое имя, заданное для переменной. Они не используются: • при задании переменной 'o', 'v' • при задании клиентских функций. Ситуации ошибки: • 112: использование переменной r в неправильном контексте; • 117: неправильный индекс переменной r • 102: если формат **"...*rn[ni;nc]**..", с ni или nc, заданным с неправильным синтаксисом (см. в этой связи: Переменные и параметры типа строки). --- Интерпретация возвращенных значений требует разграничения различных ситуаций использования (рассмотренные здесь положения действительны для всех функций, использующих переменные r): • в списке переменных r: считываются переменные списка; • при применении операции непосредственно в программа грани детали при программировании: считываются переменные r детали; • при внутренней развертке сложной операции (т. е. операций, интерпретируемых при применении подпрограммы или макропрограммы): • для задания повторно задаваемой переменной подпрограммы или в параметре операции: выполняет поиск переменных на уровнях расширения ранее (в крайнем случае - до списка переменных r главной программы); • для задания не задаваемой повторно переменной: выполняет поиск переменных на уровнях расширения ранее (в крайнем случае - до списка переменных r главной программы), но начиная со своего уровня. Пример задания переменной r типа строки: "двери*r28.*r29" пусть: r28: переменная типа строки со значением (строкой) ="p007" r29: переменная числового типа, со значением (числовым) =12.5 развертывание выражения дает значение (строку) ="porte\p007.12".
j0-j99	Глобальные программные переменные. Различаются следующие ситуации использования: • в параметре операции: применяет реальные значения переменных j; • в списке программных переменных r: применяет всегда нулевые значения; Они не используются: • при задании переменной 'o', 'v'; • при задании клиентских функций; • при задании переменных геометрий (ребра фиктивных граней). Ситуации ошибки: • 115: использование в неправильном контексте; • 118: неправильный индекс переменной "j".
$0-$299	Вспомогательные переменные: могут использоваться только при написании макропрограммы. Они не используются: • при задании переменных 'o', 'v', 'r'; • при задании клиентских функций; • при задании переменных геометрий (ребра фиктивных граней); • в тексте программы. Ситуации ошибки: • 111: использование в неправильном контексте; • 119: неправильный индекс переменной $.

Ссылки на переменные детали

Это параметрические формы, позволяющие сжать считывание программных переменных.

Обычно они используются в написании макропрограмм. Должны считаться формами продвинутого программирования.

pr[.]	Ссылка на переменную r. Обязательна параметрическая форма (использование скобок [.]) и решает с принятием для выражения значения переменной rn, где n=числовое значение в квадратных скобках. При задании переменной или числового параметра: возвращает числовое значение переменной rn (0.0 если переменная не числовая или не задана); При задании нечисловой переменной или параметра (строка), если узнается значимая форма "*pr[..]": • если rn определяет переменную типа строки: возвращает решенную соответствующую $; • если rn определяет переменную числового типа: возвращает $, соответствующую целой части значения переменной; • если rn определяет незаданную переменную: возвращает строку "0". Не может использоваться: • при задании переменной 'o', 'v'; • при задании клиентских функций. Ситуации ошибки: • 112: использование переменной r в неправильном контексте; • 117: неправильный индекс переменной r. Примеры: pr[12]: возвращает значение r12 pr[10+5]: возвращает значение r15 pr[r1], если r1=7: возвращает значение r7.
pj[.]	Ссылка на программную переменную j. Обязательна параметрическая форма (использование скобок [.]) и решает с принятием для выражения значения переменной jn, где n=расчетное значение в [.]. Различаются следующие ситуации использования: • в параметре операции: применяет реальные значения переменных j; • в списке программных переменных r: применяет всегда нулевые значения; Не может использоваться: • при задании переменной 'o', 'v'; • при задании клиентских функций; • при задании переменных геометрий (ребра фиктивных граней). Ситуации ошибки: • 115: использование в неправильном контексте; • 118: неправильный индекс переменной "j". Примеры: pj[12]: возвращает значение j12 pj[10+5]: возвращает значение j15 pj[r1], если r1=7: возвращает значение j7.
p$[.]	Ссылка на вспомогательную переменную в тексте макропрограммы. Обязательна параметрическая форма (использование скобок [.]) и решает с принятием для выражения значения переменной $n, где n=расчетное значение в [.]. Эта функция может использоваться только при написании макропрограммы. Не может использоваться: • при задании переменной 'r', 'o', 'v'; • при задании клиентских функций; • при задании переменных геометрий (ребра фиктивных граней); • в тексте программы. Ситуации ошибки: • 111: использование в неправильном контексте; • 119: неправильный индекс переменной $. Примеры: p$[12]: возвращает значение $12 p$[10+5]: возвращает значение $15 p$[r1], если r1=7: возвращает значение $7.

Задания, связанные с применением подпрограммы или макропрограммы

Это аргументы, возвращающие информацию по применению подпрограммы или макропрограммы, использование которых подразумевается в тексте подпрограммы или макропрограммы.

subx suby subz	Они дают координату размещения x, y, z: • при применении цикла (подпрограммы или макропрограммы) они принимают значение точки применения с решенными зацеплением точки и соответствующего программирования; • при задании переменной 'r': принимают значение 0.0. Аргументы subx/y/z позволяют узнать точку применения, заданную в вызове подпрограммы, в самой подпрограмме. Они не используются: • при задании переменной 'o', 'v'; • при задании клиентских функций; • при задании переменных геометрий (ребра фиктивных граней). Ситуации ошибки: • 109: использование в неправильном контексте;
subang subang0	Возвращает угол вращения: • при применении цикла они принимают значение угла вращения; • при задании переменной 'r': принимают значение 0.0. Аргумент subang (допускается краткая форма: suba) возвращает значение поворота, заданное в вызове подпрограммы, в самой подпрограмме. Аргумент subang0 позволяет узнать значение поворота, заданное в вызове подпрограммы, в самой подпрограмме, но включая все вложенные вызовы. Рассмотрим пример вызова подпрограммы с поворотом на 20.0°, которая, в свою очередь, имеет второй вызов с поворотом на -5.0°. На более внутреннем уровне результирующий поворот (полученный из: subang0) будет равен 20.0°+(-5.0°)=15.0°. Они не используются: • при задании переменной 'o', 'v'; • при задании клиентских функций; • при задании переменных геометрий (ребра фиктивных граней). Ситуации ошибки: • 109: использование в неправильном контексте;
subinv subinv0	Возвращает задание инверсии: • при применении цикла они принимают значение параметра инверсии (1, если требуется); • при задании переменной 'r': принимают значение 0. Аргумент subinv (допускается краткая форма: subi) позволяет узнать, потребовал ли вызов подпрограммы инверсию исполнения, в самой подпрограмме. Аргумент subinv0 возвращает значение инверсии, заданное в вызове подпрограммы, в самой подпрограмме, но включая все вложенные вызовы. Они не используются: • при задании переменной 'o', 'v'; • при задании клиентских функций; • при задании переменных геометрий (ребра фиктивных граней). Ситуации ошибки: • 109: использование в неправильном контексте.
submir submir0	Возвращает задание зеркального отображения: • при применении цикла они принимают значение, соответствующее запросу зеркального отображения (0=неактивно; 1= зеркальное отражение по x; 2= зеркальное отражение по y; 3= зеркальное отражение по x+y); • при задании переменной 'r': принимают значение 0. Аргумент submir (допускается краткая форма: subm) позволяет узнать, потребовал ли вызов подпрограммы симметрию исполнения, в самой подпрограмме.

	Аргумент submir0 позволяет узнать, потребовал ли вызов подпрограммы симметрии выполнения, в самой подпрограмме, но включая все вложенные вызовы. Рассмотрим пример вызова подпрограммы с активным зеркальным отображением по x, которое, в свою очередь, имеет второй вызов с активным зеркальным отображением по x+y. На более внутреннем уровне запрос зеркального отображения будет только зеркальное отображение по y. Они не используются: • при задании переменной 'o', 'v'; • при задании клиентских функций; • при задании переменных геометрий (ребра фиктивных граней). Ситуации ошибки: • 109: использование в неправильном контексте.
sublink	Возвращает задание зацепления точки: • при применении цикла они принимают значение 1, если требуется зацепление точки; • при задании переменной 'r': принимают значение 0. Аргумент sublink позволяет узнать, потребовал ли вызов подпрограммы зацепления точки исполнения, в самой подпрограмме. Допускается краткая форма: subl. Они не используются: • при задании переменной 'o', 'v'; • при задании клиентских функций; • при задании переменных геометрий (ребра фиктивных граней). Ситуации ошибки: • 109: использование в неправильном контексте.
substr substr0	Возвращает задание растягивания: • при применении цикла принимает требуемое значение растягивания; • при задании переменной 'r' принимают значение 1.0. Аргумент substr (допускается краткая форма: subs) позволяет узнать, потребовал ли вызов подпрограммы пересмотра исполнения, в самой подпрограмме. Аргумент substr0 позволяет узнать, потребовал ли вызов подпрограммы пересмотра исполнения, в самой подпрограмме, но включая все вложенные вызовы. Рассмотрим пример вызова подпрограммы с коэффициентом растяжения, равным 2.0, который, в свою очередь, разворачивает второй вызов с коэффициентом растяжения, равным 0.5: на более внутреннем уровне результирующий коэффициент растяжения будет равен 2.0*0.5=1.0. Они не используются: • при задании переменной 'o', 'v'; • при задании клиентских функций; • при задании переменных геометрий (ребра фиктивных граней). Ситуации ошибки: • 109: использование в неправильном контексте.
subemp	Возвращает задание выемки: • при применении цикла они принимают значение 1, если требуется выемка; • при задании переменной 'r': принимают значение 0. Этот аргумент позволяет узнать, потребовал ли вызов подпрограммы применение выемок, в самой подпрограмме. Допускается краткая форма: sube. Они не используются: • при задании переменной 'o', 'v'; • при задании клиентских функций; • при задании переменных геометрий (ребра фиктивных граней). Ситуации ошибки: • 109: использование в неправильном контексте.
subface	Возвращает грань применения.

	• при применении цикла принимают значение номера примененной грани (если грань реальная - это клиентский номер грани) • при задании переменной 'r': принимают значение -1. Допускается краткая форма: subf. Они не используются: • при задании переменной 'o', 'v'; • при задании клиентских функций; • при задании переменных геометрий (ребра фиктивных граней). Ситуации ошибки: • 109: использование в неправильном контексте.
submaster	Возвращает информацию о многократном вызове: • при применении цикла принимает значение: • -2: если соответствует главному вызову; • >0: если соответствует вложенному вызову и возвращает номер главной грани (если грань реальная - это клиентский номер грани); • в противном случае получается -1; • при задании переменной 'r': принимает значение -1. Они не используются: • при задании переменной 'o', 'v'; • при задании клиентских функций; • при задании переменных геометрий (ребра фиктивных граней). Ситуации ошибки: • 109: использование в неправильном контексте.
subxm subym subzm	Возвращают координату размещения (x, y, z) точки применения **главного вызова**, следовательно, относящиеся к случаю многократного вызова. Как уже для аргументов (subx, suby, subz): • при применении цикла (подпрограмма или макропрограмма), принимают значение точки применения, с решенными зацеплением точки и соответствующего программирования; • при задании переменной 'r': принимают значение 0.0. Если аргумент **submaster** возвращает: • -2: если возвращенное значение соответствует (subx, suby, subz); • >0 если соответствует вложенному вызову и возвращенное значение возвращает поле (subx, suby, subz) как применитое на главную грань. Они не используются: • при задании переменной 'o', 'v'; • при задании клиентских функций; • при задании переменных геометрий (ребра фиктивных граней). Ситуации ошибки: 109: использование в неправильном контексте.
subvl subvb subvo subvm subvk subvk1 subvk2	Они возвращают значение, заданное в поле свойств: L (subvl) B (subvb) O (subvo) M (subvm) K (subvk) K1 (subvk1) K2 (subvk2) • при применении цикла они принимают значение свойства • при задании переменной 'r' принимают значение 0. Они не используются: • при задании переменной 'o', 'v'; • при задании клиентских функций; • при задании переменных геометрий (ребра фиктивных граней).

	Ситуации ошибки: • 109: использование в неправильном контексте.
rempty[nn]	Проверяет, что задана переменная r, индекс которой указан nn. Возвращает: • 0, если переменная задана; • в противном случае получается 1. Обязательна параметрическая форма (использование скобок []) Если аргумент функции параметрический и имеет значимый формат "rn" или "r\name", функция действует напрямую на переменную rn. Могут возникнуть следующие ситуации: • в случае подпрограммы или макропрограммы возвращает значение 0, если переменная r цикла задана при вызове или соответствует переменной подпрограммы или макропрограммы, которую нельзя перезадать. • при задании переменной r или не в применении цикла принимает значение 0, если переменная r программы задана Не может использоваться: • при задании переменной 'o', 'v' • при задании клиентских функций Ситуации ошибки: • 112: использование переменной r в неправильном контексте; • 117: неправильный индекс переменной r Примеры (с использованием при разработке подпрограммы): rempty[12], rempty[r12]: возвращает 0 если переменная r12 задана, 1 иначе
rempty[nn;valdef;(nmin);(nmax)]	Проверяет, что переменная r, индекс которой указан nn, задана, возвращая значение на переменную: • если переменная задана: заданное значение в конечном итоге возвращается к максимальному диапазону значений; • иначе: значение по умолчанию. Если аргумент функции является параметрическим и имеет значительный формат "rn" или "r \ name", функция действует непосредственно на переменную rn. Аргументы: nn = индекс или имя переменной r. Значение nn должно находиться в пределах от 0 до 299; valdef = значение по умолчанию, которое должно возвращаться если переменная не задана nmin = Минимальное допустимое значение (применяется при задании) nmax = Максимальное допустимое значение (применяется при задании) • не выполняется проверка правильности крайних значений: может быть nmin > nmax; • (nmin, nmax) используются, только если переменная задана Функция работает с переменными числовой типологии, иначе: • если задана, связанное значение равно нулю (=0.0) Используются те же соображения использования, что и для предыдущего прототипа. Не используется: • при задании переменной 'o','v' • при задании клиентских функций Ситуации ошибки: • 112: использование переменной r в неправильном контексте; • 117: индекс переменной r неправильный

	• 124: число операндов >4 • 130: аргумент valdef опущен (пустое задание). Пример (с использованием при разработке подпрограммы): rempty[r12; 32;1;96]: отличаются случая • если переменная r12 не задана ->32. Иначе • со значением r12 в диапазоне (1; 96) ->r12 • со значением r12 менее 1 ->1 • со значением r12 более 96 ->96.
rvalue[cnd;nn;value]	Требует задания значения переменной r, индекс которой обозначается nn: • функция должна быть **одна** в поле задания • фактическое применение результата функции происходит только на этапе использования подпрограммы (или макропрограммы) • функцию можно использовать при назначении переменной r подпрограммы: частной и числовой. Возврат функции не является значительным • переменная, для которой запрашивается назначение значения (указывается аргументом: nn), должна присутствовать в той же подпрограмме, должна соответствовать индексу ниже, чем у соответствующей переменной, и быть: общедоступной, числовой, не назначенной (т. е. возвращение функции rempty[nn] имеет значение 1). Аргументы: cnd: флаг назначения (0 = выключено = не применяется; 1 = включено = применяется) nn = индекс переменной r. Значение nn должно находиться в пределах от 0 до 299. Допустимые форматы см. функцию (rempty); value = значение, назначаемое переменной (nn). Не используется: • при задании переменной 'o','v' • при задании клиентских функций Ситуации ошибки: • 112: использование переменной r в неправильном контексте; • 117: индекс переменной r неправильный • 124: число операндов #3. Пример (с использованием при разработке подпрограммы): rvalue[r5; 10; 96]: отличаются случая • если переменная r5 имеет значение 0 -> функция не действует. Аналогичная ситуация, если: • функция используется при задании переменной с индексом ниже 10 (например: r8), или общедоступной или строкового типа • подпрограмма не задает переменную r10, или r10 является частной или строковой, или имеет допустимое назначение. В противном случае: • числовое значение 96 будет присвоено переменной r10 • любое использование функции «rempty [10]» всегда возвращает значение 1. Эту функцию следует использовать только в том случае, если действительно необходимо.

Задание клиентских разделов

Аргументы позволяют получать прямой доступ к данным, заданным в клиентских разделах только для пунктов с числовым значением (числа с запятой или целые, выбор из списка).
Если указанный раздел или пункт не задан, аргумент принимает нулевое (0) значение.
Обычно используются при написании подпрограмм и/или макропрограмм и должны считаться формами продвинутого программирования.

Эти аргументы не могут использоваться:
- при задании переменных ('o', 'v', 'r')
- при задании переменных геометрий
- при задании клиентских разделов
- при задании клиентских функций.

szs\name	Возвращает пункт с числовым значением раздела "Специальные уставки": • неизменяемая часть "szs\"; • переменная часть ("name"): символическое имя, заданное для этого пункта в разделе. Пример: "szs\aaa" : возвращает значение поля с именем "aaa".
szi\name	Возвращает пункт с числовым значением раздела "Дополнительная информация".
szo\name	Возвращает пункт с числовым значением раздела "Настройки оптимизации".
szl\name	Возвращает пункт с числовым значением раздела "Настройки связей".

Глобальные переменные

Использование Глобальных переменных возможно при их специальном подключении в конфигурации ТраCAD.
Это чисто числовые переменные, вызываемые только по имени, с максимальным числом 300.
Список глобальных переменных также задается в конфигурации ТраCAD. Это информация, которую невозможно изменить в программе.

glb\name	Возвращает значение, заданное в соответствующей Глобальной переменной: • неизменяемая часть "glb\"; • переменная часть ("имя"): символическое имя переменной. Использование: всегда. Если переменная не задана, подается общий сигнал ошибки синтаксиса.

Вспомогательные функции

Эти функции обычно используются в написании подпрограмм и/или макропрограмм.
Должны считаться формами продвинутого программирования.

nfa	Возвращает реальный номер грани (клиентской), указанной как аргумент. Эта функция работает только в случае аргумента со значением от 1 до 6: аргумент интерпретируется как клиентский номер грани, а функция возвращает реальный номер грани. В противном случае, функция всегда возвращает целую часть аргумента. Использование: всегда. Ситуации ошибки: нет. Примеры: пусть задана клиентская нумерация граней в соответствии с представленным ниже: грань 1 -> клиентский номер: 5 грань 2 -> клиентский номер: 6 грань 3 -> клиентский номер: 1 грань 4 -> клиентский номер: 4 грань 5 -> клиентский номер: 2 грань 6 -> клиентский номер: 3 nfa5=nfa[5]=1 nfa2=nfa[2]=5

nfc	Возвращает клиентский номер грани (реальной), указанной как аргумент. Эта функция работает только в случае аргумента со значением от 1 до 6: аргумент интерпретируется как реальный номер грани, а функция возвращает клиентский номер грани. В противном случае, функция всегда возвращает целую часть аргумента. Использование: всегда. Ситуации ошибки: нет. Примеры: (пусть задана клиентская нумерация граней, как указано для функции: nfa) nfc1=nfc[1]=5 nfa2= nfc[2]=6

Математические функции

abs	Вычисляет абсолютное значение аргумента. Допускается краткая форма: a. Использование: всегда. Ситуации ошибки: нет. Примеры: abs5 = 5 abs[-5] = 5
sqrt	Извлекает квадратный корень из аргумента. Значение аргумента должно быть положительным (>=0.0). Допускается краткая форма: q. Использование: всегда. Ситуации ошибки: • 127: отрицательный аргумент. Примеры: sqrt[25] = 5 sqrt[-25] <- определяет ошибку (127)
int	Возвращает целую часть аргумента, полученную отсечением. (удаляет дробную часть). Допускается краткая форма: i. Использование: всегда. Ситуации ошибки: нет. Примеры: int[-12.8] = -12.0 int[12.9] = 12
inv	Возвращает обратный аргумент (1/x). Аргумент не может быть равен нулю. Допускается кратка форма: v. Использование: всегда. Ситуации ошибки: • 125: аргумент равен нулю. Примеры: inv2 = 0,5 inv0 <- определяет ошибку (125)
pow	Возводит аргумент в степень 2. Допускается кратка форма: p. Использование: всегда. Ситуации ошибки: нет. Примеры: pow3= 9 pow[-3]= 9 pow0 = 0
pown[nb;ne] pown[nb;ne1;ne2]	Возводит первый аргумент (основание) в степень, полученную из второго аргумента (показатель степени): • nb (1-й аргумент) = основание • ne (2-й аргумент) = показатель степени Применяется к целой части • ne1, ne2 (2-й и 3-й аргумент) = показатель степени вычисляется как ne=(ne1/ne2) и используется без усечения до целой части Может быть 2 или 3 аргумента функции. Особые случаи:

	• nb#0.0, ne=0.0 возвращает 1 • nb=0.0, ne=0.0 возвращает 1 • ne2=0.0 использует форму с 2 аргументами Использование: всегда. Ситуации ошибки: 123: число операндов =0; 124: число операндов #2,3; 128: nb=0.0, ne<0.0 (отрицательный). Примеры: pown[5;2] = 5 * 5 =25 pown[5;3.5] = pown[5;3.5;0] =pown[5;3] = 5 * 5 * 5 = 125 pown[5; 0] = 1 pown[0; 5] = 1 pown[5;1;2] = 5 ^1/2 = 2,236 (математический эквивалент: квадратный корень из (5 ^1)) pown[5;2;3] = 5 ^2/3 = 2.924 (математический эквивалент: кубический корень из (5 ^2))
round	округляет аргумент до ближайшего целого числа. Использование: всегда. Ситуации ошибки: нет. Примеры: round[12.8] = 13 round[12.3] = 12 **round[12.5] = 12 <- до 0,5, округляет в меньшую сторону** **round[12.501] = 13 <- свыше 0,5, округляет в большую сторону** round[-10.3] = -10 round[-10.7] = -11
range[nval;(nmin);(nmax)]	Утверждает значение в пределах максимального интервала значений. • nval (1-й аргумент) = значение, которое должно утверждаться • nmin (2-й аргумент) = минимальное допустимое значение (применяет при задании) • nmax (3-й аргумент) = максимальное допустимое значение (применяется при задании) Не проводится контроль правильности крайних значений: может быть nmin > nmax. Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов >3; • 130: опускается первый аргумент. Примеры: range[5;0;10] возвращает 5 range[5;0;4] возвращает 4 (ограничивает максимальным значением) range[-5;0;4] возвращает 0 (ограничивает минимальным значением) range[-5;;4] возвращает -5 (не применяет минимальное значение) range[5;0] возвращает 5 (не применяет максимальное значение)

odd	Возвращает 1, если целая часть аргумента - нечетная; в противном случае - 0. Использование: всегда. Ситуации ошибки: нет. Примеры: odd12.8 = odd12 = 0 odd13.8 = odd13 = 1
sign[nval]	Возвращает: • 1, если аргумент положительный или равен нулю (>=0) • -1, если аргумент отрицательный.

	Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов #1; Примеры: sign[12.8] = 1 sign[-13.8] = -1
hypot[c1;c2] hypot[c1;c2; c3]	Возвращает гипотенузу прямоугольного треугольника с заданными катетами. Может быть 2 или 3 аргумента функции. С 3 аргументами треугольник задается в пространстве. Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов #2 и #3; Пример: hypot[5;2;1] = sqrt[5*5+2*2+1*1] = 5.477 hypot[5;2;1] = sqrt[5*5+2*2+1*1] = 5.477
min[n1;..;n30] max[n1;..;n30] ave[n1;..;n30] sum[n1;..;n30]	Возвращают: меньшее, большее, среднее значение или сумму значений, принятых за аргумент. Максимальное число аргументов - 30. Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов >30. Примеры min[5;12;3;25] возвращает 3 max[5;12;3;25] возвращает 25 sum[5;12;3;25] возвращает 5 + 12 + 3 + 25 = 45 ave[5;12;3;25] возвращает (5 + 12 + 3 + 25) /4 = 11,25 <- делит сумму на число аргументов
minr[n1;n2] maxr[n1;n2] aver[n1;n2] sumr[n1;n2]	Возвращают: меньшее, большее, среднее значение или сумму значений, присвоенных переменным r в интервале (n1, n2). Аргументов функции должно быть 2. Соответствующие переменные могут быть любого типа: • числовая (двойная, целая): функция считывает значение; • строка: функция принимает значение 0.0. Аргументы n1 и n2 должны определить интервал переменных между r0 и r299. Они не используются: • при задании переменной 'o', 'v' • при задании клиентских функций. Ситуации ошибки: • 112: использование переменной r в неправильном контексте; • 123: число операндов =0; • 124: число операндов #2; • 117: неправильный индекс переменной r. Пример r10=minr[2;5], со значениями r2=5; r3=12; r4=3; r5=25: функция возвращает 3; эквивалентно использованию min[5;12;3;25]
minj[n1;n2] maxj[n1;n2] avej[n1;n2] sumj[n1;n2]	Возвращают: меньшее, большее, среднее значение или сумму значений, присвоенных переменным j в интервале (n1, n2). Аргументов функции должно быть 2. Аргументы n1 и n2 должны определить интервал переменных между j0 и j99. Различаются следующие ситуации использования: • в параметре операции: применяет реальные значения переменных j; • в списке переменных r программы: применяет всегда нулевые значения; Они не используются: • при задании переменной 'o', 'v' • при задании клиентских функций • при задании переменных геометрий (ребра фиктивных граней).

	Ситуации ошибки: • 123: число операндов =0; • 124: число операндов #2; • 115: использование в неправильном контексте; • 118: неправильный индекс переменной j. Пример maxj[2;5] (со значениями j2=5; j3=12; j4=3; j5=25) -> функция возвращает 25.
min$[n1;n2] max$[n1;n2] ave$[n1;n2] sum$[n1;n2]	Возвращают: меньшее, большее, среднее значение или сумму значений, присвоенных переменным $ в интервале (n1, n2). Аргументов функции должно быть 2. Аргументы n1 и n2 должны определить интервал переменных между $0 и $299. Они не используются: • при задании переменной 'r', 'o', 'v'; • при задании клиентских функций; • при задании переменных геометрий (ребра фиктивных граней); • в тексте программы. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов #2; • 111: использование в неправильном контексте; • 119: неправильный индекс переменной $. Пример ave$[2;5] (со значениями $2=5; $3=12; $4=3; $5=25) -> функция возвращает 11,25

Тригонометрические функции

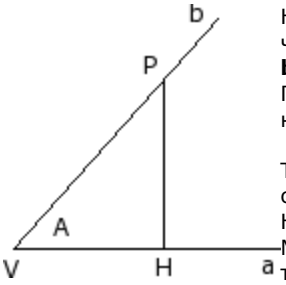
Начала тригонометрии

Приведенные здесь основы тригонометрии могут помочь в решении возникающих при программировании геометрических проблем.

Основные единицы измерения плоских углов: градус и радиан.

В математике обычно используется линейное измерение углов с единицей измерения радиан, однако более распространенной единицей измерения углов является, несомненно, градус. Поэтому рассматриваемые ниже тригонометрический функции запрашивают и возвращают угловые значения, выраженные в градусах.

Напоминаем, что 1 радиан = (180/p) °, где (p = 3.1415..).



Напоминаем также, что для угла положительным считается направление против часовой стрелки. Рассмотрим угол **A** (в радианах) с вершиной **V** и сторонами **a** и **b**.
Произвольно отметим на луче **b** точку **P**, отличную от вершины **V**, спроецируем ее на луч **a**: точка **H** - это точка перпендикуляра, опущенного из **P** на **a**.

Теперь рассмотрим прямоугольный треугольник **VHP** и отношения между ориентированными сегментами:
HP/VP; VH/VP; HP/ VH
Можно доказать, что эти отношения зависят только от угла A, а не от выбора точки P на луче b.

Три записанные отношения определяют три функции угла **A**, которые называются следующим образом:

синус **A**	$\frac{HP}{VP} = sin\ A$ в частности, при **VP**=1, **HP**=sin**A**
косинус **A**	$\frac{VH}{VP} = cos\ A$ в частности, при **VP**=1, **HP**=cos**A**
тангенс **A**	$\frac{HP}{VH} = tg\ A$

Можно доказать также значимое отношение: $(\sin\mathbf{A})^2 + (\cos\mathbf{A})^2 = 1.0$.

Со ссылкой на рисунок действуют следующие указанные соответствия:



Aє	sinA	cosA	tgA=sinA/cosA
0	0.0	1.0	0.0
0ч90	0.0ч1.0	1.0ч0.0	0.0÷ +(бесконечность)
90	1	0.0	+(бесконечность)
90ч180	1.0ч0.0	0.0ч(-1.0)	-(бесконечность) ÷ 0.0
180	0.0	-1.0	0.0
180ч270	0.0ч(-1.0)	(-1.0)ч0.0	0.0 ÷ +(бесконечность)
270	-1.0	0.0	-(бесконечность)
270ч360	(-1.0)ч0.0	0.0ч1.0	-(бесконечность) ÷ 0.0
360	0.0	1.0	0.0

Функции

sin	Вычисляет синус аргумента (выражен в °). Значение функции находится в интервале (-1.0 ч 1.0). Допускается краткая форма: s. Использование: всегда. Ситуации ошибки: нет. Пример sin[90]= 1 sin[-90] = -1
cos	Вычисляет косинус аргумента (выражен в °). Значение функции находится в интервале (-1.0 ÷ 1.0). Допускается краткая форма: c. Использование: всегда. Ситуации ошибки: нет. Пример cos[90]= 0 cos[gr[pi]] = -1
tan	Вычисляет тангенс аргумента (выражен в °). Допускается краткая форма: t. Использование: всегда. Ситуации ошибки: • 132: неправильный угол для расчета тангенса Пример tan[45]= 1 tan[90]= определяет ошибку 132 tan[-90]= определяет ошибку 132
asin,as	Вычисляет арксинус аргумента. Возвращаемое функцией значение выражено в ° (градусах) и находится в интервале от 0 до 180°. Значение аргумента должно находиться в пределах от - 1 до 1. Допускается краткая форма: d. Использование: всегда. Ситуации ошибки: • 126: аргумент за пределами интервала значений (- 1; 1). Пример asin1= 90 asin[-1] = -90
acos,ac	Вычисляет арккосинус аргумента. Возвращаемое функцией значение выражено в ° (градусах) и находится в интервале от 0 до 180°. Значение аргумента должно находиться в пределах от - 1 до 1. Допускается краткая форма: e. Использование: всегда. Ситуации ошибки:

	• 126: аргумент за пределами интервала значений (- 1; 1). Пример acos0= 90 acos[-1] = 180
atan,at	Вычисляет арктангенс аргумента. Возвращаемое функцией значение выражено в ° (градусах) и находится в интервале от -90° до 90°. Допускается краткая форма: f. Использование: всегда. Ситуации ошибки: нет. Пример atan1= 45 atan[-1] = -45
gr	преобразует аргумент из радианов в градусы (°): 1 радиан =(180/пи) °. Допускается краткая форма: g. Использование: всегда. Ситуации ошибки: нет. Пример gr[pi] = 180
atan2[y,x]	Вычисляет арктангенс (y/x). Возвращаемое функцией значение выражено в ° (градусах) и находится в интервале от -180° до 180°. Если оба аргумента равны нулю, возвращает значение 0. Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов #2; Пример atan2[1;0]=90 atan2[0;0]=0 atan2[1;1]=45

Функции, действующие на строки

Должны считаться функциями продвинутого программирования.

strlen[nn]	Возвращает количество символов, заданных для переменной r заданного с аргументом индекса. Рассмотрим строку: • задания: если переменная числовая, • значение: если переменная - типа строки. В случае незаданной переменной возвращает значение 0. Значимый формат Если аргумент функции - параметрический и имеет значимый формат "rn" или "r\name" (в этом случае обязательна параметрическая форма: использование скобок [.]): функция действует непосредственно на переменную rn. Не может использоваться: • при задании переменной 'o', 'v'; • при задании клиентских функций Ситуации ошибки: • 112: использование переменной r в неправильном контексте; • 117: неправильный индекс переменной r. Пример 1 При задании переменной типа строки r5="submio*r4", с r4="pippo" Она решает параметризацию r5: "submio\pippo" strlen5 возвращает значение 12, соответствующее числу символов в "submio\pippo"; strlen[r5] возвращает значение 12, соответствующее числу символов в r5="submio\pippo"; Пример 2 Пусть будет задана числовая переменная r5="r4/12" strlen5 возвращает значение 5, соответствующее числу символов в "r4/12".

getat[nn;np]	Возвращает дробное значение, соответствующее символу, извлеченному из строки: • задания: если переменная числовая, • значение: если переменная - типа строки. Возвращает значение 0, в случае: • незаданной переменной, • неправильного положения символа (меньше 1 или больше длины строки). Аргументы: nn = индекс переменной r. Значение nn должно находиться в пределах от 0 до 299; np = положение символа в значении строки (значимо с 1). Значимый формат Если первый аргумент функции параметрический и имеет значимый формат "rn" или "r\name", функция действует напрямую на переменную rn. Значимый формат Первый аргумент может напрямую задавать строку между двумя кавычками (напр., “pippo”). В этом случае: функция может использоваться всегда. Не может использоваться: • при задании переменной 'o', 'v' • при задании клиентских функций Ситуации ошибки: • 112: использование переменной r в неправильном контексте; • 123: число операндов =0; • 124: число операндов #2; • 117: неправильный индекс переменной r. Пример 1 при задании переменной типа строки r5="s2*r4", с r4="pippo" решает параметризацию r5: "s2\pippo" getat[5;2] возвращает 50, что соответствует дробному значению символа '2' getat[5;6] возвращает 112, что соответствует дробному значению символа 'p' Пример 2 пусть будет задана числовая переменная r5="r4/12" strlen[5,2] возвращает 52, что соответствует дробному значению символа '4'.
strcmp[n1;n2]	Возвращает значение сравнения между двумя строками: • 0, если две строки одинаковы, • <0, если первая строка меньше второй, • >0, если первая строка больше второй. Сравнение не учитывает различия между прописными и строчными буквами. Сравнение завершается при выявлении неравенства или после сравнения строк. Если две строки оказываются одинаковыми в конце одной из них, а в другой еще остаются символы, эта строка будет считаться большей. Возвращаемое значение - результат последнего выполненного сравнения. Аргументы: n1= индекс первой переменной r n2= индекс второй переменной r Аргументы могут иметь параметрическое задание. Действует на переменную r любого типа: • если типа строки: применяет к решенной $, • если числового типа: применяет к $ ввода данных, • если переменная не задана: применяет к пустой строке. Один (или оба) аргумента могут задавать непосредственно строку, заключенную в двойные кавычки. Значимый формат Если аргумент функции параметрический и имеет значимый формат "rn" или "r\name", функция действует напрямую на переменную rn. Не может использоваться: • при задании переменной 'o', 'v' • при задании клиентских функций. Ситуации ошибки: • 112: использование переменной r в неправильном контексте; • 123: число операндов =0; • 124: число операндов #2; • 117: неправильный индекс переменной r.

	Примеры: strcmp[5; "pippo"] оценивает r5 и сравнивает со строкой "pippo" strcmp[r5; "pippo"] оценивает r5 и сравнивает со строкой "pippo" strcmp["pippo";r6] оценивает r6 и сравнивает со строкой "pippo" strcmp[r5;r6] оценивает и сравнивает r5 и r6.
strfind[n1;n2(;np)]	Возвращает положение возникновения одной строки в другой. Аргументы (n1, n2) имеют значение и синтаксис, аналогичные функции *strcmp.* Аргумент np = начальное положение для поиска (значимо от 1). Не может использоваться: • при задании переменной 'o', 'v' • при задании клиентских функций. Ситуации ошибки: • 112: использование переменной r в неправильном контексте; • 123: число операндов =0; • 124: число операндов #2; • 117: неправильный индекс переменной r. Примеры: (при: r5=”pippo”, r6=”i”) strfind[5;"p"] оценивает r5 и выполняет поиск первого возникновения “p” (возврат: 1) strfind[r5;"p";4] оценивает r5 и выполняет поиск первого возникновения “p” начиная с 4-о символа (возврат: 4) strfind["pippo";r6] выполняет поиск r6 в строке "pippo" (возврат: 2) strfind[r5;r6] выполняет поиск r6 в r5 (возврат: 2) strfind[r5;r6;3] выполняет поиск r6 в r5 начиная с 3-о символа (возврат: 0) strfind["pippo";”i”] выполняет поиск ”i” в строке "pippo" (возврат: 2)

toolex[nn;nfield] tooltip[nn;nfield]	Применяются к переменной r типа строки или интерпретируют строку значения. Аргументы: nn = индекс переменной r nfield = индекс поля (см. далее). Аргументы могут иметь параметрическое задание. Эти функции возвращают: toolex: значение поля nfield, получаемое в строке значения tooltip: 1 если поле nfield оказывается числовым, 0 если переменная не задана или числовая или nfield неправильно. Поле признается: числовым (без знака и без дробных долей) или нечисловым. Для обеих функций реализован особый случай nfield = 0: возвращают число полей, узнанных в строке значения. Значимый формат Если первый аргумент функции параметрический и имеет значимый формат "rn" или "r\name", функция действует напрямую на переменную rn. Значимый формат Первый аргумент может напрямую задавать строку между двумя кавычками (напр., “pippo”). В этом случае: функция может использоваться всегда. Не может использоваться: • при задании переменной 'o', 'v' • при задании клиентских функций. Ситуации ошибки: • 112: использование переменной r в неправильном контексте; • 123: число операндов =0; • 124: число операндов #2; • 117: неправильный индекс переменной r. Пример: r5="12|25;64" **toolex[5]=5** Число узнанных полей = 5 **tooltip[5;1]=1** **toolex[5;1]=12** 1-е поле значения = 12 числовое поле **tooltip[5;2]=0** **toolex[5;2]=124** 2-е поле значения = дробное значение '|' = 124 нечисловое поле ....

	3-е поле значение = 25 числовое поле 4-е поле значение = дробное ';' = 59 нечисловое поле 5-е поле значение = 64 числовое поле n-ое поле (n > 5) значение = 0 нечисловое поле. Приведенный пример показывает, как эта функция может использоваться для толкования соответствующего программирования, например, в маске инструментов. В этом случае в представленном примере может быть запрошен выбор инструментов в положении от 12 до 25, плюс 64.

Логические функции

ifelse[nc;n1;n2;(eps)]	Минимальный тернарный оператор: • возвращает n1, если nc отличен от нуля, • в противном случае возвращает n2. Аргументов функции должно быть 3 или 4. Сравнение равенства между nc и нулевым значением (0) оценивается по значению эпсилон: • в варианте с 3 аргументами применяемый эпсилон - значение от 0.0 до 0.001, как задано в конфигурации TpaCAD (Эпсилон, используемый в логических сравнениях) • в варианте с 4 аргументами эпсилон задается на последний аргумент (значим в абсолютном значении). Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов #3; Функция всегда оценивает как n1, так и n2, даже если возвращает только одно из двух значений. Однако фильтруются некоторые особые ситуации, соответствующие математическим ошибкам, которые могут возникнуть при оценке невозвращенного терма. В частности, не сигнализируются следующие ошибки: • 125: знаменатель равен нулю, при выполнении делений или функции inv; • 127: отрицательный аргумент в функции sqrt; • 126: аргумент за пределами интервала значений (- 1; 1) в функциях asin, acos. • 128: аргумент ne <0 или >10 в функции pown. Пример: ifelse[50;100;l/2] = 100 ifelse[0;100;l/2] = l/2
ifcase[nc1;nesp;nc2;n1;n2;(eps)]	Полный тернарный оператор: оценивает условие (nc1 ? nc2) с '?', заданным nesp: • если условие выполняется, возвращает n1, • в противном случае n2. Функция возвращает n2 даже если *nesp* задан на неправильное значение. Аргументов функции должно быть 5 или 6. Аргумент *nesp* интерпретируется для задания условия между nc1 и nc2: • значение 0 соответствует < (меньше) • значение 1 соответствует a <= (меньше или равно) • значение 2 соответствует > (больше) • значение 3 соответствует >= (больше или равно) • значение 4 соответствует = (равно) • значение 5 соответствует <> (не равно) Для аргумента nesp вместо числового значения можно также задать соответствующие символические формы. Например, условие "больше или равно" может задаваться значением 3 или как ">=". Отношение различия может выражаться как "<>" или как "#". Условие сравнения между nc1 и nc2 оценивается по значению эпсилон:

	• в варианте с 5 аргументами применяемый эпсилон - значение от 0.0 до 0.001, как задано в конфигурации TpaCAD (Эпсилон, используемый в логических сравнениях) • в варианте с 6 аргументами эпсилон задается на последний аргумент (значим в абсолютном значении). Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов #5; Функция всегда оценивает как n1, так и n2, даже если возвращает только одно из двух значений. Однако фильтруются те же особые ситуации, рассмотренные для функции ifcase, при математических ошибках, которые могут возникнуть при оценке невозвращенного терма. Примеры ifcase[5; >=;12;3;25] = 25 ifcase[5;<;12;3;25] = 3 ifcase[5;<> ;12;3;25] = 3
case[nc;nc1:nv1;nc2:nv2;..;nvdef]	Оператор тестирования условия: оценивает условия (nc = nc1), (nc = nc2), возвращая значение "nv", присвоенное первому проверенному условию. Тестирует возникновения условия, среди заданных, возвращая значение, присвоенное правильному проверенному условию. Аргументы таковы: • nc : оцениваемое значение • nc1 : первое значение сравнения с nc • nv : значение, возвращенное функцией, если nc = nc1 • nvdef: значение по умолчанию, если не было определено какого-либо соответствия. Аргумент *nvdef* не является обязательным (если не задан, имеет значение 0), если задан, должен следовать как последний аргумент. Разделительный знак между nc и nc1 - ";" (точка с запятой), а между nc1 и nv1 - обязательно ":" (двоеточие). Максимальное число управляемых случаев равно 10, включая nvdef. Все аргументы могут быть числовыми или параметрическими. Условие сравнения между nc и заданными значениями nc* оценивается по значению эпсилон от 0.0 до 0.001, как задано в конфигурации TpaCAD (Эпсилон, используемый в логических сравнениях). Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов <2 или >11. Функция всегда оценивает nv1, nv2…. nvdef, даже если возвращает только одно из двух значений. Однако фильтруются те же особые ситуации, рассмотренные для функции ifcase, при математических ошибках, которые могут возникнуть при оценке невозвращенных термов. Пример case[h;100:r0;200:h-100;l:l/2;h] • если h=100, возвращается значение переменной (r0) • если h=200, возвращается значение, даваемое (h-100) • если h=l, возвращается значение, даваемое (l/2) • если не было выявлено какого-либо соответствия, возвращается значение (h).
not[nc]	Оператор отрицания аргумента: • если nc=0, возвращает 1; • если nc#0 (отлично от 0), возвращает 0. Сравнение равенства между nc и значением ноль (0) оценивается по значению эпсилон от 0.0 до 0.001, как задано в конфигурации TpaCAD (Эпсилон, используемый в логических сравнениях). Примеры

	not[5] = 0 not[0] = 1

Технологические функции

Как правило, можно получить доступ к любой информации (параметру) по технологической установке. Как мы уже видели ранее, реальное задание технологии установки зависит от одного приложения, однако имеется группа общих функций, дающих доступ к этой информации.

Каждый параметр:

- как правило, доступен с цифровым идентификатором (тип) или, дополнительно, - типом строки. См., например, функцию *prtool*, в которой *nkind* определяет тип параметра в цифровой форме или как имя. Для параметров, представляющих особый интерес, nkind может указывать символическое имя со следующим формализмом:
    - неизменяемая часть "p\";
    - переменная часть, не персонализируемая (см. таблицу ниже) - символическое имя, заданное для параметра. Связь между символическим именем и числовым идентификатором (типом) параметра выполняется автоматически
- в некоторых случаях доступен с указанием функциональности. См. функцию prfi, которая напрямую считывает параметр, задающий диаметр инструмента
- может адресоваться абсолютным образом как ячейка матрицы, с указанием (строка, столбец). Этот режим адресации предусматривает, что параметры установки могут интерпретироваться в матричной организации: каждый элемент информации имеет свои идентификационные данные (цифровые и буквенные), а также *положение* в матрице. Функции (prmxmac, prmxgru,…) имеют доступ к каждому отдельному параметру таким образом. Использование этих функций предусматривает досконального знание структуры параметров станка и поэтому разрешается только разработчикам.

Технологические параметры, заданные с символическим формализмом

p\gron	Тип параметра активизации группы головки (числовой: 6)
p\face	Тип параметра задания одной/нескольких рабочих граней шпинделя (числовой: 6)
p\ofx p\ofy p\ofz	Типы параметров: • сдвиг (x/y/z) группы • сдвиг (x/y/z) шпинделя (числовые: 100, 101, 102)
p\xmax p\xmin	Типы параметров минимального и максимального пределов позиционирования группы головки на оси X (числовые: 150, 151)
p\ymax p\ymin	Типы параметров минимального и максимального пределов позиционирования группы головки на оси Y (числовые: 152, 153)
p\zmax p\zmin	Типы параметров минимального и максимального пределов позиционирования группы головки на оси Z (числовые: 154, 155)
p\cmax p\cmin	Типы параметров минимального и максимального пределов позиционирования группы головки на оси C (числовые: 158, 157)
p\betamax p\betamin	Типы параметров минимального и максимального пределов позиционирования группы головки на оси B (числовые: 160, 159)
p\attr	Типы параметров оснащения шпинделя или положения держателя инструмента (числовой: 220)
p\fitool	Тип параметра диаметра инструмента (числовой: 1002)
p\tiptool	Тип параметра типа инструмента (числовой: 1001)
p\lltool	Тип полезной длины инструмента (числовой: 109)
p\ltottool	Тип общей длины инструмента (числовой: 111)

p\lauxtool	Тип вспомогательной длины инструмента (числовой: 112)
p\ariatool	Тип параметра координаты инструмента в воздухе (числовой: 121)
p\feedmin	Тип параметра минимальной рабочей скорости инструмента (числовой: 2004)
p\feedmax	Тип параметра максимальной рабочей скорости инструмента (числовой: 2006)
p\feed	Тип параметра рабочей скорости инструмента по умолчанию (числовой: 2005)
p\rpmmin	Тип параметра минимальной скорости вращения инструмента (числовой: 2001)
p\rpmmax	Тип параметра максимальной скорости вращения инструмента (числовой: 2002)
p\rpm	Тип параметра скорости вращения инструмента по умолчанию (числовой: 2003)
p\invtool	Тип параметра зеркального инструмента (числовой: 124)

Функции доступа к общей группе установки

Должны считаться функциями продвинутого программирования.

primp[nkind;(vdef)]	Возвращает параметр установки: • nkind = тип параметра (обязательно) • vdef = значение по умолчанию, если требуемый параметр не найден. Если не задан или пуст, использует 0. Если nkind =0 (ноль), функция возвращает число станков, сконфигурированных в установке. Распознается также форма *primp["nameKind";(vdef)]*, где параметр указан по имени. Значение "nameKind" должно согласовываться на этапе определения технологических характеристик установки. Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов >2; • 130: аргумент nkind опущен (задание пусто). Пример: primp[1100]: возвращает значение параметра 1100 (0, если параметр не найден) Пример: primp[1100]: возвращает значение параметра 1100 (0, если параметр не найден)

Функции доступа к ветви станка для конфигурации групп головки

Должны считаться функциями продвинутого программирования

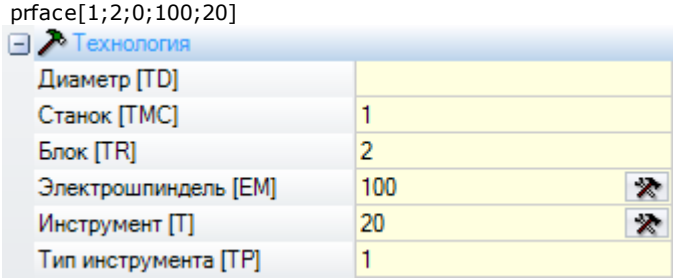
prmac[(nm);nkind;(vdef)]	Возвращает параметр станка: • nm = станок (обязательно) (если задание пусто: использует =1) • kind = тип параметра (обязательно) • vdef = значение по умолчанию (если требуемый параметр не найден). Максимальное задаваемое значение для nm определяется конфигурацией технологии установки. Если nkind =0 (ноль), функция возвращает 1 (отлично от 0), если станок сконфигурирован и присутствует в установке. Распознается также форма *prmac[(nm);"nameKind";(vdef)]*, где параметр указан по имени. Значение "nameKind" должно согласовываться на этапе определения технологических характеристик установки. Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов <2 или >3; • 130: аргумент nkind опущен (задание пусто).

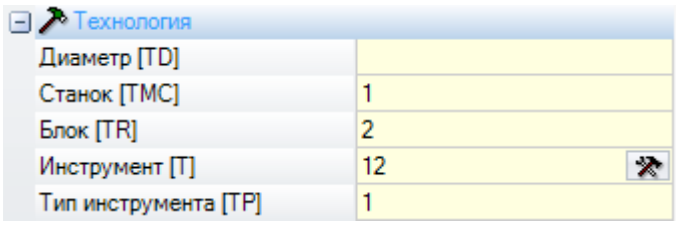
	Примеры • prmac[2;1100]: возвращает значение параметра (1100) станка 2 (0, если параметр не задан) • prmac[;1100;100]: возвращает значение параметра (1100) станка 1 (станок по умолчанию) (100, если параметр не задан) • prmac[2;0]: возвращает значение параметра имени зон станка 2 (0, если параметр не задан)
prgr[(nm);(ng);nkind;(vdef)]	Возвращает параметр группы головки: • nm = станок (обязательно) (если задание пусто: использует =1) • ng = группа головки (обязательно) (если задание пусто: использует =1) • kind = тип параметра (обязательно) • vdef = значение по умолчанию (если требуемый параметр не найден). Если не задан или пуст, использует 0. Максимальные используемые для nm и ng значения определяются конфигурацией технологий установки и станка. Если nKind =0 (ноль), функция возвращает 1 (отлично от 0), если группа сконфигурирована и присутствует. Распознается также форма *prgr[(nm);(ng);"nameKind";(vdef)]*, где параметр указан по имени. Значение "nameKind" должно согласовываться на этапе определения технологических характеристик установки. Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов <3 или >4; • 130: аргумент nkind опущен (задание пусто). Примеры • prgr[2;3;1100]: возвращает значение параметра (1100) группы 3 станка 2 (0, если параметр не задан) • prgr[2;;1100;100]: возвращает значение параметра (1100) группы 1 (группы по умолчанию) станка 2 (100, если параметр не задан) • prgr[2;3;p\ofx]: возвращает значение параметра (p\ofx = сдвиг x группы) группы 3 станка 2

Функции доступа к инструментам

Если в Конфигурации TpaCAD включено управление универсальным инструментом, можно присвоить значение 0 станку, группе и шпинделю. Однако всегда надо использовать полный прототип функции.

prface[(nm);(ng);(np);(ns);nt;(side)] prface[(nm);(ng);(nt);side] prface[ng;nt;nside] prface[nt;nside]	Проверяет, может ли инструмент работать на грани nside: • nm = станок (обязательно) (если задание пусто: использует =1) • ng = группа головки (обязательно) (если задание пусто: использует =1) • np = если оснащение, определенное (ns;nt), уточняет держатель, параметр np указывает положение оснащенного инструмента на держателе • ns = шпиндель (обязательно) • nt = оснащаемый инструмент/держатель на шпинделе ns или шпиндель. (если nt не задан, задается значение 0) • nside = грань (если опущен или задание пусто, задает nside = текущая грань; если задан, интерпретирует клиентский номер грани) Функция возвращает значение 1, если тест пройден, в противном случае возвращает 0. Можем рассмотреть несколько конкретных случаев: (ns<=0; nt=0) Не указана правильная технология. Функция возвращает значение 0; (ns>0; nt=0)

	Шпиндель ns оснащен, как определяется конфигурацией группы головки. В частности: • если ns не оснащен (это положение электрошпинделя со сменой инструмента), эта проверка выполняется на информации рабочей грани, заданной электрошпинделю • если ns оснащен держателем, np указывает положение инструмента на держателе (умолчание, если np=0: первое сверло). Проверка осуществляется только на конфигурации ns. (ns<=0; nt#0) Теперь шпиндель указан в nt и оснащен, как определяется конфигурацией группы головки (см. случай: (ns>0; nt=0), теперь со шпинделем в nt). (ns>0; nt#0) Если nt имеет значимое значение (в правильном диапазоне инструмента или держателя), шпиндель ns считается оснащенным nt. В частности, если nt указывает держатель, np указывает положение инструмента на держателе. В случае оснащения на держателе: • если np не задан: проверка выполняется на первом оснащенном инструменте. • если np задан неправильно (<=0 или за пределами максимально допустимого значения), функция возвращает значение 0.
	Максимальные значения, используемые для (nm, ng, np, ns, nt), определяются конфигурацией технологий установки, станка, группы и каталога инструментов и держателей. Зависит от конфигурации технологий также порядок оснащения скорее инструмента, чем держателя инструмента. Номер грани nside задает плоскость и ориентацию оси z. Ниже приводятся эквивалентные формы уменьшенных форматов: prface[nm;ng;nt;side] означает ->prface[nm;ng;0;-1;nt;side] prface[nm;ng;nt;side] означает ->prface[1;ng;0;-1;nt;side] prface[nt; nside] означает -> prface[1;1;0;-1;nt;side]. Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов <3 или >6 • 130: опущены оба аргумента ns и nt (задание пусто). Примеры prface[;;;;90]: головка, если шпиндель обозначен как (nm=1; ng=1; np=0; ns=1; nt=90), может работать на активированной грани. Со ссылкой на программирование технологии операции, рассмотрим, какое соответствие задать параметрам функции *prface* для испытания рабочей грани в случае: • программирования технологии для шпинделя и инструмента: prface[1;2;0;100;20]  • программирования технологии для шпинделя и инструмента: prface[1;2;0;100;20] • программирования технологии для шпинделя, где возможны две эквивалентные формы: prface[1;2;0;0;12] prface[1;2;0;12;0]

	
prfi[(nm);(ng);(np);(ns);nt] prfi[(nm);(ng);(ns);nt] prfi[(nm);(ng);nt] prfi[ng;nt] prfi[nt]	Возвращает диаметр инструмента: • nm = станок (обязательно) (если задание пусто: использует =1) • ng = группа головки (обязательно) (если задание пусто: использует =1) • np = если оснащение, определенное (ns;nt), уточняет держатель, параметр np указывает положение оснащенного инструмента на держателе • ns = шпиндель (обязательно) • nt = оснащаемый инструмент/держатель на шпинделе ns или шпиндель. (если nt не задан, задается значение 0) Можем рассмотреть те же особые случаи, которые мы рассматривали для функции *prface.* Ниже приводятся эквивалентные формы уменьшенных форматов: prfi[nm;ng;nt] означает -> prfi[nm;ng;0;-1;nt] prfi [ng; nt] означает -> prfi[1;ng;0;-1;nt] prfi [nt] означает-> prfi[1;1;0;-1;nt] Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов <3 или >5; • 130: опущены оба аргумента nt и ns (задание пусто). Примеры prfi[1;1;90]: возвращает диаметр шпинделя, определенного как (nm=1; ng=1; np=0; ns=1; nt=90) prfi[0;0;;0;1015]: возвращает диаметр универсального инструмента, определенного как (nm=0; ng=0; np=0; ns=0; nt=1015)
prrot[(nm);(ng);(np);(ns);nt] prrot[(nm);(ng);(ns);nt] prrot[(nm);(ng);nt] prrot[ng;nt] prrot[nt]	Возвращает направление вращения инструмента: • nm = станок (обязательно) (если задание пусто: использует =1) • ng = группа головки (обязательно) (если задание пусто: использует =1) • np = если инструмент, обозначенный (ns; nt), указывает держатель инструмента, параметр np указывает положение инструмента, установленного на держателе инструмента. • ns = шпиндель (обязательно) • nt = снащаемый инструмент/держатель на шпинделе ns или шпиндель. (Если nt не задан, задается значение 0) Функция возвращает значение 1, если направление против часовой стрелки, в противном случае возвращает 0. Можем рассмотреть те же особые случаи, которые мы рассматривали для функции *prface*. Ниже приводятся эквивалентные формы уменьшенных форматов: prrot[nm;ng;nt] означает -> prrot[nm;ng;0;-1;nt] prrot[ng; nt] означает -> prrot[1;ng;0;-1;nt] prrot[nt] означает -> prrot[1;1;0;-1;nt] Использование: всегда. Ситуации ошибки:

	• 123: число операндов= 0; • 124: число операндов <3 или >5; • 130: опущены оба аргумента nt и ns (задание пусто).
prtool[(nm);(ng);(np);ns;(nt);nkind;(vdef)] prtool[(nm);(ng);nt;nkind;(vdef)] prtool[nm;ng;nt;nkind] prtool[ng;nt;nkind] prtool[nt;nkind]	Возвращает общий параметр инструмента: • nm = станок (обязательно) (если задание пусто: использует =1) • ng = группа головки (обязательно) (если задание пусто: использует =1) • если оснащение, определенное (ns;nt), уточняет держатель, параметр np указывает положение оснащенного инструмента на держателе • ns = шпиндель (обязательно) • nt = оснащаемый инструмент/держатель на шпинделе ns или шпиндель. (если nt не задан, задается значение 0). • nkind = тип параметра (обязательно) • vdef = значение по умолчанию (возвращается, если параметр не найден). (если vdef не задан, задается значение 0). Можем рассмотреть те же особые случаи, которые мы рассматривали для функции *prface*. Максимальные значения, используемые для (nm, ng, np, ns, nt), определяются конфигурацией технологий установки, станка, группы и каталога инструментов и держателей. Зависит от конфигурации технологий также порядок оснащения скорее инструмента, чем держателя инструмента. Распознаются также формы с *nKind*, замененным на *"nameKind"*, где параметр указан по имени. Значение "nameKind" должно согласовываться на этапе определения технологических характеристик установки.
	Ниже приводятся эквивалентные формы уменьшенных форматов: prtool[nm;ng;nt;nkind;vdef] означает -> prtool[nm;ng;0;-1;nt;nkind;vdef] prtool[nm;ng;nt;nkind] означает-> prtool[nm;ng;0;-1;nt;nkind;0.0] prtool[ng;nt; nkind] означает -> prtool[1;ng;0;-1;nt;nkind;0.0] prtool[nt;nkind] означает -> prtool[1;1;0;-1;nt;nkind;0.0]. Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов <2 или >7; • 130: опущены оба аргумента nt и ns или nkind (задание пусто). Примеры prtool[1;2;;100;3;100]: возвращает значение параметра (100) инструмента, обозначенного как (nm=1; ng=2; np=0;ns=100, nr=3). Если не находится требуемый параметр, возвращает 0. prtool[1;1;90;p\fitool]: считывает диаметр шпинделя в положении (nm=1; ng=1; nt=90). Параметр p\fitool может заменяться значением 1002. prtool[1;1;90;p\tiptool]: считывает значение типа шпинделя в положении (nm=1; ng=1; nt=90).
prtip[(nm);(ng);(np);ns;(nt)] prtip[(nm);(ng);(ns);nt] prtip[nm;ng;nt] prtip[ng;nt] prtip[nt]	Возвращает тип инструмента: • nm=станок (обязательно) (если задание пусто: использует =1) • ng = группа головки (обязательно) (если задание пусто: использует =1) • np= если оснащение получено из (ns;nt), определяет держатель, указывает положение оснащенного инструмента на держателе

	• ns=шпиндель (обязательно) • nt=оснащаемый инструмент/держатель на шпинделе ns или шпиндель (если задание пусто: использует =0) Можем рассмотреть те же особые случаи, которые мы рассматривали для функции *prface.* Ниже приводятся эквивалентные формы уменьшенных форматов: prtip[nm; ng; nt] означает->prtip[nm;ng;0;-1;nt] prtip[ng; nt] означает->prtip[1;ng;0;-1;nt] prtip[nt] означает->prtip[1;1;0;-1;nt]. Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов <1 или >5; • 130: опущены оба аргумента nt и ns
prfulcrox [(nm);(ng);(np);ns;(nt)] prfulcroy [(nm);(ng);(np);ns;(nt)] prfulcroz [(nm);(ng);(np);ns;(nt)]	Они возвращают положение (x, y, z) точки опоры указанного инструмента, в условиях станка на 0: • nm = станок (обязательно) (если задание пусто: использует =1) • ng = группа головки (обязательно) (если задание пусто: использует =1) • np=если оснащение, определенное (ns;nt), уточняет держатель, параметр np указывает положение оснащенного инструмента на держателе • ns = шпиндель (обязательно) • nt=оснащаемый инструмент/держатель на шпинделе ns или шпиндель (если задание пусто: использует =0) Можем рассмотреть те же особые случаи, которые мы рассматривали для функции *prface.* Максимальные значения, используемые для (nm, ng, np, ns, nt), определяются конфигурацией технологий установки, станка, группы и каталога инструментов и держателей. Зависит от конфигурации технологий также порядок оснащения скорее инструмента, чем держателя инструмента. Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов #5; • 130: опущены оба аргумента nt и ns (задание пусто).
pngru[(nm);(ng);(ns);nt;(deltafi)] pnstool[(nm);(ng);(ns);nt;(deltafi)] pntool[(nm);(ng);(ns);nt;(deltafi)]	Они возвращают информацию о замене технологии: • nm = станок (обязательно) (если задание пусто: использует =1) • ng = группа головки (обязательно) (если задание пусто: использует =1) • ns = шпиндель (обязательно • nt = оснащаемый инструмент/держатель на шпинделе ns или шпиндель (если задание пусто: использует =0) • deltafi = допустимое смещение на диаметр ▪ значение 0.0 (или <epsilon) указывает, что замена возможна только с тем же диаметром ▪ положительное значение (напр.: 3.0) указывает, что замена допускает максимальную разность между диаметрами на +/-3.0 (мм/дюймов) ▪ отрицательное значение (напр.: -1.0) указывает, что замена допускает максимальную разность на 1.0 (мм/дюймов), но только в уменьшении по сравнению с исходным диаметром Функции возвращают: • pngru: число Группы взамен • pnstool: число шпинделя взамен

	• pntool: число Инструмента взамен Функции возвращают отрицательное значение (-1), если технология требует замены, котороя невозможна. Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов <4 или>5; • 130: опущены оба аргумента nt и ns (задание пусто).

Функции прямого доступа к матрицам установки

Как уже говорилось, матричный режим адресации предусматривает, что параметры установки могут интерпретироваться в матричной организации: каждый элемент информации имеет свои идентификационные данные (цифровые и буквенные, а также *положение* в матрице.
Функции должны рассматриваться продвинутым программированием.

prmxmac[(nm);nrow;ncol,(vdef)]	Возвращает значение одной общей ячейки матрицы конфигурации станка: • nm = станок (обязательно) (если задание пусто: использует =1) • nrow = строка матрицы (обязательно) (значимо начиная со значения 1) • ncol = столбец матрицы (обязательно) (значимо начиная со значения 1) • vdef = значение по умолчанию (возвращается, если параметр не найден). Если не задано или задано пустой: использует =0 Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов, отличное от 3,4; • 130: опущен аргумент nrow или ncol (задано пустой).
prmxgru[(nm);(ng);nrow;ncol,(vdef)]	Возвращает значение одной общей ячейки матрицы конфигурации групп: • nm = станок (обязательно) (если задание пусто: использует =1) • ng = группа головки (обязательно) (значимо с 1) • nrow = строка матрицы, соответствующая группе ng (обязательно) (значимо начиная с 1) • ncol = столбец матрицы (обязательно) (значимо начиная с 1) • vdef = (возвращается, если параметр не найден). Если не задано или задано пустой: использует =0 Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов, отличное от 4 и 5; • 130: опущен аргумент ng, nrow или ncoll (задано пустой).
prmxtool[(nm);ntool;(nkind);(ncol);(vdef)]	Возвращает значение одной общей информации из матрицы каталога инструментов: • nm = станок (обязательно) (если задание пусто: использует =1) • ntool = номер инструмента (обязательно) • nkind = тип параметра

	• ncol = столбец матрицы (значимо с 1) (возвращается, если параметр не найден). Если не задано или задано пустой: использует =0 Если ntool равен нулю (=0): функция возвращает максимальное идентификационное число инструментов, сконфигурированных в каталоге инструментов. Если для ntool задано ненулевое значение, выбирает инструмент (принимается абсолютное значение). Можно выделить следующие случаи: • nkind =0, ncol =0 (ноль): функция возвращает 1 (отлично от 0), если сконфигурирован инструмент ntool; • nkind #0 (отлично от нуля 0: считывает параметр типа nkind (nkind: рассматривается как абсолютное значение), заданный для инструмента ntool; • nkind =0, ncol #0 (отлично от 0): считывает параметр в столбце ncol (ncol: рассматривается как абсолютное значение), заданный для инструмента ntool. Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов < 2 или > 5; • 130: опущен аргумент nrow или ncol (задано пустой). Примеры: prmxtool[1;0;0;0]: считывает максимальное идентификационное число инструментов, сконфигурированных в каталоге инструментов станка 1 prmxtool[1;3;p\fitool]: считывает диаметр инструмента 3 каталога инструментов станка 1 prmxtool[1;3;0;6]: считывает параметр столбца 6 инструмента 3 каталога инструментов станка 1
prmxhtool[(nm);ntool;(nrow);(nkind);(ncol);(vdef)]	Возвращает значение одной общей информации из матрицы держателей (каталог): • nm = станок (обязательно) (если задание пусто: использует =1) • ntool = номер держателя (обязательно) (значимо начиная с 1) • nrow = строка держателя (обязательно) (значимо начиная с 0) • nkind = тип параметра • ncol = столбец матрицы (значимо с 1) • vdef = значение по умолчанию (в случае недоступной ячейки) - (если не задано или задано пустой: использует =0) Если ntool равен нулю: эта функция возвращает номер держателей, сконфигурированных в каталоге держателей. Если ntool отличен от 0: выбирает держатель (принимается абсолютное значение). Можно выделить следующие случаи: • nkind =0, ncol =0 (ноль): функция возвращает 1 (отлично от 0), если сконфигурирован держатель ntool; • nkind#0 (отлично от нуля 0): считывает параметр типа nkind (nkind: рассматривается

	в абсолютном значении), заданный для инструмента ntool в строке конфигурации nrow; • nkind =0, ncol #0 (отлично от 0): считывает параметр в столбце ncol (ncol: рассматривается как абсолютное значение), заданный для инструмента ntool в строке конфигурации nrow. Если nrow=0: указывает строку конфигурации держателя. Если nrow #0 (принимается абсолютное значение): указывает строку конфигурации nrow оснащенного инструмента (значимо для максимального числа оснащаемых инструментов). Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов < 2 или > 6; • 130: опущен аргумент ntool, nrow или ncol (задано пустой). Примеры: prmxhtool[1;0]: считывает число держателей, сконфигурированных для станка 1 prmxhtool[1;3]: считывает 1, если сконфигурирован держатель 3 станка 1 prmxhtool[1;3;0;6]: считывает параметр, определенный в столбце 6 строки конфигурации держателя 3 станка 1 prmxhtool[1;3;4;p\ofx]: считывает корректирующий параметр x сверла 4 держателя 3 станка 1 prmxhtool[1;3;4;;37]: считывает параметр столбца 37 центра 4 держателя 3 станка 1
prmxstore[(nm);nstore;(nrow);(nkind);(ncol),(vdef)]	Возвращает значение, содержащееся в одной общей ячейке матрицы конфигурации магазина инструментов: • nm = станок (обязательно) (если задание пусто: использует =1) • nstore = номер магазина (значимо с 1) • nrow = строка матрицы магазина, определенного параметром nstore (обязательно) (значимо начиная с 0) • nkind = тип параметра • ncol = столбец матрицы (значимо с 1) • vdef = значение по умолчанию (в случае недоступной ячейки) - (если не задано или задано пустой: использует =0) Если параметр *nstore* равен нулю, возвращает число сконфигурированных магазинов, Если *nstore* отличен от 0: выбирает магазин инструментов (принимается абсолютное значение). Можно выделить следующие случаи: • nkind=0, ncol=0: функция возвращает 1 (т. е. отлична от 0), если положение, определенное nrow, является оснащенным. • nkind#0: возвращает параметр типа nkind (значение nkind рассматривается как абсолютное значение), заданный для положения nrow • nkind=0, ncol#0: возвращает параметр, определенный в столбце ncol (значение ncol рассматривается как абсолютное значение) и строке nrow

	Использование: всегда. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов < 2 или > 6; • 130: опущен аргумент nstore (задано пустой) Примеры prmxstore[1;0]: считывает число магазинов инструментов, сконфигурированных для станка 1 prmxstore [1;3]: считывает максимальное оснащенное положение для магазина инструментов 3 станка 1 prmxstore [1;3;4]: считывает 1, если оснащено положение 4 магазина инструментов 3 станка 1 prmxstore [1;3;4;1200]: считывает параметр типа 1200 из положения 4 магазина инструментов 3 станка 1 prmxstore [1;3;4;0;12]: считывает параметр, определенный в столбце 12 из положения 4 магазина инструментов 3 станка 1

Универсальные функции геометрической библиотеки

Это функция, реализующая большую группу функций, в основном, геометрических: первый аргумент функции указывает значимое имя выбора функции. Затем он указан жирным шрифтом и, в любом случае, документируется отдельно. Для всех рассмотренных случаев действительны приведенные ниже примечания:
Может использоваться:
- всегда в версиях, не использующих имена операций
- Версии, использующие имена операций, не могут использоваться при задании переменных 'o' и 'v', переменных геометрий (ребра фиктивных граней), клиентских функций.

Ситуации ошибки:
- 123: число операндов =0;
- 124: неправильное число операндов.
- 116: неправильный контекст в версиях, использующих имена операций

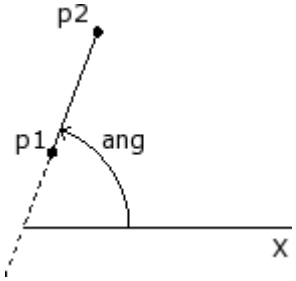
Не сигнализируются ситуации геометрической ошибки, в любом случае, принимается условие по умолчанию.
Геометрический контекст, используемый для решения геометрических ситуаций, соответствует чистой системе прямоугольных координат XYZ. Соответствие аргументов, указанных для оси (X (предпочтительнее), а также Y или Z) с реальными осями грани или детали, как правило, полностью абстрактно. Аргумент, указанный для оси X, может на самом деле соответствовать оси Z наклонной грани или детали: геометрические функции имеют силу геометрической библиотеки, используемой с адаптацией аргументов и результатов к конкретным потребностям.

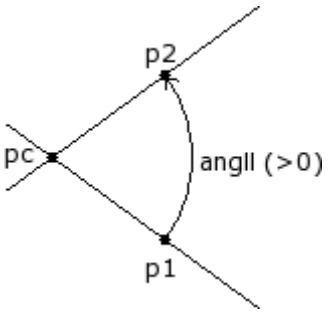
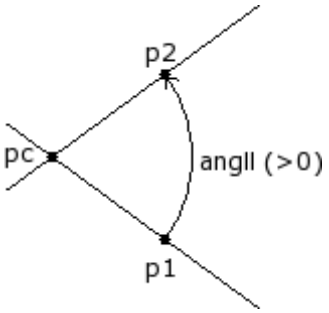
Версии функции, использующие имена операций, ищут операцию, указанную до текущей операции. Поиск прерывается при первом соответствии. Имя операции указано формализмом "wname" и должно быть заключено в двойные кавычки.
К поиску можно суммировать терм. Распознается следующий синтаксис:
- “wname+2”: указывает, что искомая операции находится через две строки после “wname”
- “wname-2”: указывает, что искомая операции находится за две строки до “wname”
- “wname+”;nn: где сдвиг присвоен в дополнительном аргументе (в т. ч. в параметрической форме). Можно использовать также “wname-”.

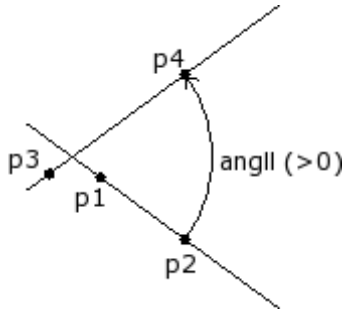
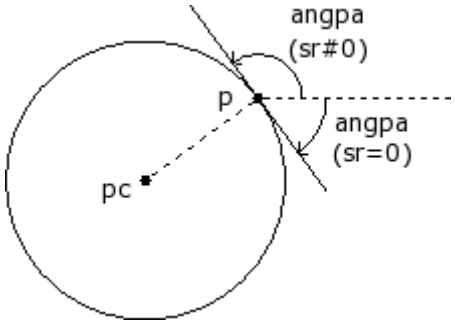
Частично функции должны рассматриваться продвинутым программированием.

Функции расчета углов

geo[**angc**;ang;(sgn)]	Возвращает угол ang, уменьшенный до: 0° – 360°. ang = угол sgn = если значение отлично от 0, сохраняет изначальный знак ang (если поле пустое или не задано, использует =0) Примеры geo[angc;4500] = 180 geo[angc;-450]=270 geo[angc;-450;1]=-90

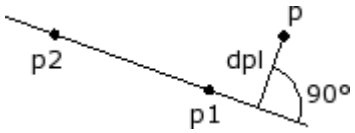
geo[**angm**;ang;angr;(sgn)]	Оценивает, является ли (ang) кратным (angr), и возвращает число кратных. ang = угол (используется уменьшенным до +/-(0° – 360°)) angr = кратный угол (используется в абсолютном значении) sgn = если значение отлично от 0, сохраняет изначальный знак ang (если поле пустое или не задано, использует =0). Возвращает 0, если: • |ang| <= 0.001 • |angr| <= 0.001 • (ang) не кратен (angr). В противном случае возвращает целое значение #0. Возвращенное значение отрицательно, если (ang<0) и (sgn#0). Примеры geo[angm;180;90;0] = 2 geo[angm;-180;90;0] = 2 geo[angm;-180;90;1] = -2 geo[angm;181;90;0] = 0 geo[angm;540;90;0] = 2 (ang) уменьшается до (0° – 360°)
geo[**ang**;x1;y1;x2;y2]	Возвращает угол между осью x и ориентированной линией P1-P2: x1;y1 = абсцисса и ордината точки P1 x2;y2 = абсцисса и ордината точки P2  Угол имеет значение в интервале [0 <= ang < 360], в градусах [°]. Если точки P1 и P2 совпадают, возвращает значение 0. Примеры geo[ang;100;100;400;400] = 45
geo[**ang**; "wname"] geo[**ang**; "wname+nn"] geo[**ang**; "wname",nn"]	Эта функция аналогична функции geo[**ang**;x1;y1;x2;y2]. Ориентированная линия P1-P2 определяется именем операции. Операция "wname+nn" должна соответствовать линейному отрезку, заданному только с одним отрезком. Если операция определена неправильно, функция возвращает значение 0.0.
geo[**angll**;xc;yc;x1;y1;x2;y2]	Возвращает угол между двумя ориентированными линиями Pc-P1 и Pc-P2 (Pc=центр) xc;yc = абсцисса и ордината точки Pc x1;y1 = абсцисса и ордината точки P1 x2;y2 = абсцисса и ордината точки P2 Угол имеет значение в интервале [–180 < angll <= +180], в градусах [°]: • ноль (=0): если точки Pc,P1 и P2 не являются различными, • положительный: если линия Pc-P1 замыкается на Pc-P2 против часовой стрелки; • отрицательный: в других случаях.

	 Примеры geo[angll;100;100;200;0;400;100]= 45 geo[angll;100;100;400;100;200;0]= -45 geo[angll;100;100;400;100; 400;100]= 0
geo[**angll**;xc;yc;zc;x1;y1;z1;x2;y2;z2]	Возвращает угол между двумя ориентированными линиями Pc-P1 и Pc-P2 (Pc=центр) в пространстве:  xc;yc;zc = координаты точки Pc x1;y1;z1 = координаты точки P1 x2;y2;z2 = координаты точки P2 Угол имеет значение в интервале [−180 < angll <= +180], в градусах [°]. Если точки Pc,P1 и P2 не являются различными: угол имеет значение 0. Примеры geo[angll;400;0;-100;400;0;0;450;-20;0] = 28.3 geo[angll;0;0;0;100;0;0;0;100;0]=90 угол между двумя осями координат (ось X и ось Y) geo[angll;0;0;0;100;0;0;0;0;100]=90 угол между двумя осями координат (ось X и ось Z)
geo[**angll**;x1;y1;x2;y2;x3;y3;x4;y4]	Возвращает угол между двумя ориентированными линиями P1-P2 и P3-P4 x1;y1 = абсцисса и ордината точки P1 x2;y2 = абсцисса и ордината точки P2 x3;y3 = абсцисса и ордината точки P3 x4;y4 = абсцисса и ордината точки P4 Угол имеет значение в интервале [−180 < angll <= +180], в градусах [°]: • ноль (=0): если две линии параллельны или совпадают, • положительный: если линия P1-P2 замыкается на P3-P4 против часовой стрелки; • отрицательный: в других случаях.

	 Примеры geo[angll;100;100;200;0;0;100;400;100] = 45
geo[**angll;**"wname1+nn";x3;y3;x4;y4] geo[**angll;**"wname1+nn","wname2+nn"]	Эта функция аналогична функции geo[**angll**;x1;y1;x2;y2;x3;y3;x4;y4]. Возвращает угол между двумя ориентированными линиями, из которых первая или обе линии определяется именем операции. "wname1"=имя операции, задающей первый отрезок "wname2"=имя операции, задающей второй отрезок Операции "wname1" и "wname2" должны определять линейные отрезки, состоящие только из одного отрезка. Если операции правильно не обнаруживаются, функция возвращает значение 0.0.
geo[**angpc**;x;y;xc;yc;(sr)]	Возвращает угол касания точки P на окружности: x;y = абсцисса и ордината точки P xc;yc = абсцисса и ордината центра круга Pc sr = направление вращения на круге (#0 против часовой стрелки; 0=по часовой стрелке= по умолчанию, если не задано). Угол имеет значение в интервале [0 <= ang <= 360], в градусах [°]. Если точки P и Pc совпадают, возвращает значение 0.  Примеры geo[angpc;0;100;0;0;1] = 180 geo[angpc;0;100;0;0] = 0
geo[**angpc**;"wname+nn"]	Эта функция аналогична функции geo[**angpc**;x;y;xc;yc;(sr)]. Возвращает угол касания точки P на дуге круга, заданной именем операции: • операция "wname" должна определить дугу на плоскости xy, состоящую только из одного отрезка; • точка P - это конечная точка дуги. Если операция определена неправильно, функция возвращает значение 0.0.

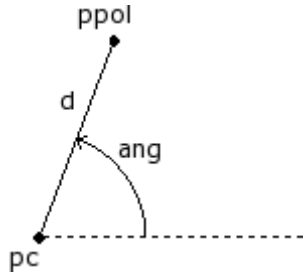
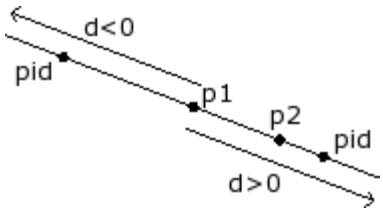
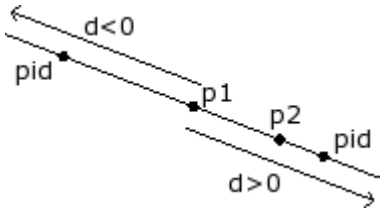
Функции расчета расстояний

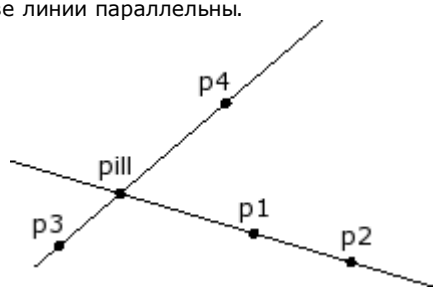
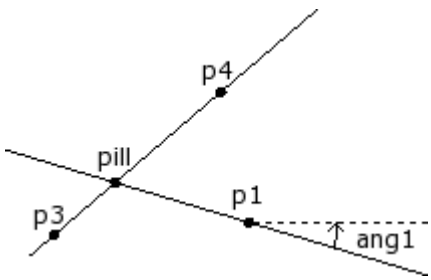
geo[**dist**;x1;y1;x2;y2]	Возвращает расстояние между двумя точками P1, P2 (на плоскости): x1;y1 = абсцисса и ордината точки P1 x2;y2 = абсцисса и ордината точки P2.

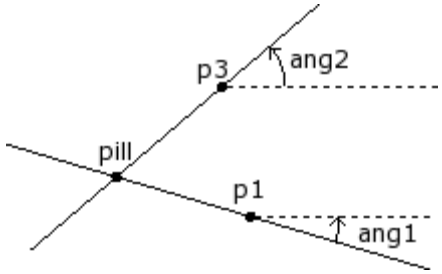
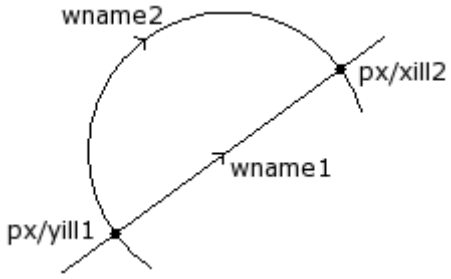
	Примеры geo[dist;0;100;100;-200] = 316.2278
geo[**dist**;x1;y1;z1;x2;y2;z2]	Возвращает расстояние между двумя точками P1, P2 (в пространстве): x1;y1;z1 = абсцисса, ордината и глубина точки P1 x2;y2;z2 = абсцисса, ордината и глубина точки P2. Примеры geo[dist;0;100;10;100;-200;-10] = 316.8596
geo[**dpl**;x;y;x1;y1;x2;y2]	Возвращает минимальное расстояние точки P от линии P1-P2: x;y = абсцисса и ордината точки P x1;y1 = абсцисса и ордината точки P1 x2;y2 = абсцисса и ордината точки P2.  Расстояние равно нулю, если точка P относится к линии. Примеры geo[dpl;0;200;0.;0;100;100] = 141.4214 geo[dpl;50;50;0.;0;100;100] = 0 <- точка (50;50) находится на прямой
geo[**dpl**;"wname1+nn";x1;y1;x2;y2] geo[**dpl;**x;y;"wname2+nn"] geo[**dpl;**"wname1+nn";"wname2+nn"]	Эта функция аналогична функции geo[**dpl**;x;y;x1;y1;x2;y2]. Возвращает расстояние точки P от линии P1-P2: точка и/или линия могут определяться именем операции. "wname1"=имя операции, задающей точку P. Может соответствовать: • точке или подготовке • отрезку профиля (линия или дуга): точка P - конечная точка отрезка; "wname2"=имя операции, задающей линейный отрезок. Операция должна определить линейный отрезок, состоящий только из одного отрезка. Если операция определена неправильно, функция возвращает значение 0.0.
geo[dim;"wname+nn";(nmodo)]	Возвращает длину профиля или элемента профиля: • nModo: 0=(умолчание) возвращает длину профиля от операции ”wname” в конце (но до текущей операции), включая возможные запрограммированные в подготовке входные/выходные отрезки • nModo: 1 (#0) = возвращает длину элемента ”wname” (если подготовка=0)

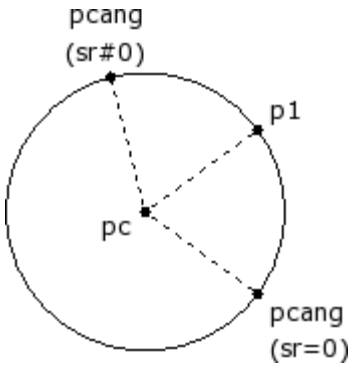
Функции определения точек на геометрических элементах

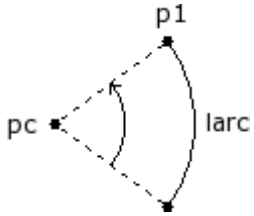
geo[**pxpol**;xc;yc;ang;d] geo[**pypol**;xc;yc;ang;d]	Возвращают абсциссу (x) или ординату (y) точки, заданной с полярными режимами: xc;yc = абсцисса и ордината точки Pc (центра полярной системы) ang = угол d = вектор (применяется в абсолютном значении).

	 Если вектор d равен нулю: функция возвращает изначальную координату точки Pc. Примеры geo[pxpol;0;0;30;100] = 86.6025 geo[pypol;0;0;30;100] = 50
geo[**pxld**;x1;y1;x2;y2;d] geo[**pyld**;x1;y1;x2;y2;d]	Возвращают абсциссу (x) или ординату (y) точки на линии P1-P2 (заданной на плоскости), на расстоянии d от P1: x1;y1 = абсцисса и ордината точки P1 x2;y2 = абсцисса и ордината точки P2 d = расстояние. Если d>0 (положительно): точка вычисляется от P1 к P2; Если d<0 (отрицательно): точка вычисляется от P1 в направлении, противоположном P2; Если d=0 (положительно): точка совпадает с P1.  Примеры geo[pxld;0;0;100;0;200] = 200 geo[pyld;0;0;100;0;200] = 0
geo[**pxld**;x1;y1;z1;x2;y2;z2;d] geo[**pyld**;x1;y1;z1;x2;y2;z2;d] geo[**pzld**;x1;y1;z1;x2;y2;z2;d]	Возвращают абсциссу (x), ординату (y) или координату z точки на линии P1-P2 (заданной в пространстве), на расстоянии d от P1.  x1;y1;z1 = абсцисса и ордината точки P1 x2;y2;z2 = абсцисса и ордината точки P2 d = расстояние. Если d>0 (положительно): точка вычисляется от P1 к P2; Если d<0 (отрицательно): точка вычисляется от P1 в направлении, противоположном P2; Если d=0 (положительно): точка совпадает с P1. Если точки P1 и P2 совпадают, возвращается координата P1.
geo[**pxld**;"wname+nn";d]	Эти функции аналогичны функциям:

geo[**pyld**;"wname+nn";d] geo[**pzld**;"wname+nn";d]	geo[**pxld**;x1;y1;z1;x2;y2;z2;d], geo[**pyld**;x1;y1;z1;x2;y2;z2;d] и geo[**pzld**;x1;y1;z1;x2;y2;z2;d]. Возвращают координату (x, y или z) точки на линии, определенной именем операции, на расстоянии d от P1. Операция "wname1" должна определить линейный отрезок, состоящий только из одного отрезка. Если операция определена неправильно, функция возвращает значение 0.0.
geo[**pxill**;x1;y1;x2;y2;x3;y3;x4;y4] geo[**pyill**;x1;y1;x2;y2;x3;y3;x4;y4]	Возвращают абсциссу (x) или ординату (y) точки пересечения между двумя линиями P1-P2 и P3-P4: x1;y1 = абсцисса и ордината точки P1 x2;y2 = абсцисса и ордината точки P2 x3;y3 = абсцисса и ордината точки P3 x4;y4 = абсцисса и ордината точки P4 в любом случае, возвращают координату точки P1, если: • одна или обе линии заданы с нулевым отрезком (P1=P2 и/или P3=P4); • две линии совпадают; • две линии параллельны.  <u>Примеры</u> geo[pxill;0;0;300;300;0;300;300;0] = 150
geo[**pxill**;x1;y1;ang1;x3;y3;x4;y4] geo[**pyill**;x1;y1;ang1;x3;y3;x4;y4]	Возвращают абсциссу (x) или ординату (y) точки пересечения между двумя линиями P1-ang1 и P3-P4: x1;y1 = абсцисса и ордината точки P1 ang1 = угол наклона первой линии x3;y3 = абсцисса и ордината точки P3 x4;y4 = абсцисса и ордината точки P4 В любом случае, возвращают координату точки P1, если: • вторая линия задана с нулевым отрезком (P3=P4); • две линии совпадают; • две линии параллельны.  <u>Примеры</u> geo[pxill;0;0;45;0;300;300;0] = 150
geo[**pxill**;x1;y1;ang1;x3;y3;ang2] geo[**pyill**;x1;y1;ang1;x3;y3;ang2]	Возвращают абсциссу (x) или ординату (y) точки пересечения между двумя линиями P1-

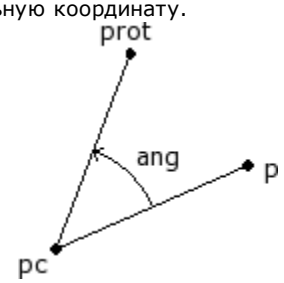
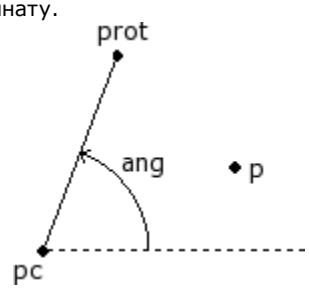
	ang1 и P3-ang2: x1;y1 = абсцисса и ордината точки P1 ang1 = угол наклона первой линии x3;y3 = абсцисса и ордината точки P3 ang2 = угол наклона второй линии.  В любом случае, возвращают координату точки P1, если: • две линии совпадают; • две линии параллельны.
geo[**pxill**;"wname1+nn";x3;y3;x4;y4] geo[**pxill**;"wname1+nn";x3;y3;ang2] geo[**pxill**;"wname1+nn";"wname2+nn"] geo[**pxill**;"wname1+nn";"wname2+nn"(;nsol)] geo[**pyill**;"wname1+nn";x3;y3;x4;y4] geo[**pyill**;"wname1+nn";x3;y3;ang2] geo[**pyill**;"wname1+nn";"wname2+nn"] geo[**pxill**;"wname1+nn";"wname2+nn"(;nsol)]	Эти функции аналогичны функциям: geo[**pxill**;x1;y1;ang1;x3;y3;x4;y4] и geo[**pyill**;x1;y1;ang1;x3;y3;x4;y4]. Возвращают абсциссу (x) или ординату (y) точки пересечения между двумя операциями, определенными именем, соответственно, "wname1" и "wname2". Операции должны определить линейный отрезок или дугу круга, состоящего только из одного отрезка, или дугу конуса (эллипс/овал) или размеченный элемент. nsol = в форме с "wname1" и "wname2", если находятся две точки пересечения, можно определить, какая из двух возвращает: • nsol=1, возвращается первое решение (это умолчание, если не задан nsol), • nsol=2 возвращается второе решение, • nsol=3, 4 возвращается третье/четвертое решение (это может быть значимо в случае пересечения между двумя конусами или дугой и конусом).  В примере на рисунке: ▪ "wname1" - линейный сегмент ▪ "wname2" - дуга. Эти два отрезка имеют 2 точки пересечения: ▪ пересечение, возвращаемое по умолчанию, - ближайшее к начальной точке отрезка "wname1" (nsol=1 или не задан). ▪ второе пересечение возвращается, если nsol=2. Если операции правильно не обнаруживаются, функция возвращает значение 0.0.

geo[**pxme**;x1;y1;x2;y2] geo[**pyme**;x1;y1;x2;y2]	Возвращают абсциссу (x) или ординату (y) средней точки отрезка P1-P2 (на плоскости): x1;y1 = абсцисса и ордината точки P1 x2;y2 = абсцисса и ордината точки P2. Примеры geo[pxme;0;0;400;300] = 200 geo[pyme;0;0;400;300] = 150
geo[**pxme**;x1;y1; z1;x2;y2; z2] geo[**pyme**;x1;y1; z1;x2;y2; z2] geo[**pzme**;x1;y1; z1;x2;y2;z2]	Возвращают абсциссу (x), ординату (y) или глубину (z) средней точки отрезка P1-P2 (в пространстве): x1;y1;z1 = абсцисса, ордината и глубина точки P1 x2;y2;z2 = абсцисса, ордината и глубина точки P2. Примеры geo[pxme;0;0;0;400;300;50] = 200 geo[pyme;0;0;0;400;300;50] = 150 geo[pzme;0;0;0;400;300;50] = 25
geo[**pxme**;"wname+nn"] geo[**pyme**;"wname+nn"] geo[**pzme**;"wname+nn"]	Эти функции аналогичны функциям: geo[**pxme**;x1;y1; z1;x2;y2; z2], geo[**pyme**;x1;y1; z1;x2;y2; z2], geo[**pzme**;x1;y1;z1;x2;y2; z2]. Возвращают координату (x, y или z) средней точки отрезка, определенного именем операции. Операция должна определить линейный отрезок или дугу, состоящую только из одного отрезка. Функция вычисляет линейное расстояние между концами отрезка. Напоминаем, что длина дуги вычисляется функцией geo[**larc**;"wname"]. Если операция определена неправильно, функция возвращает значение 0.0.
geo[**pxcang**;x1;y1;xc;yc;ang;(sr)] geo[**pycang**;x1;y1;xc;yc;ang;(sr)]	Возвращают абсциссу (x) или ординату (y) точки P круга, определенной начиная с точки P1, с прохождением угла по или против часовой стрелки: x;y = абсцисса и ордината точки P1 xc;yc = абсцисса и ордината центра круга Pc ang = угол прохождения sr = направление вращения на круге (#0 против часовой стрелки; 0=по часовой стрелке= по умолчанию, если не задано). Если точки P1 и Pc совпадают: возвращает изначальную координату P1.  Примеры geo[pxcang;0;100;0;0;45] = 70.7107 geo[pxcang;0;100;0;0;45;1] = -70.7107
geo[**larc;**rad;ang]	Возвращает длину дуги круга, определенной начиная с радиуса rad с прохождением угла

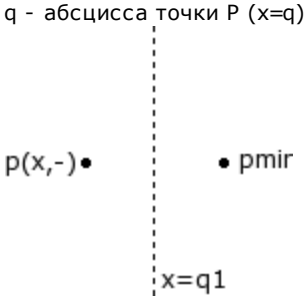
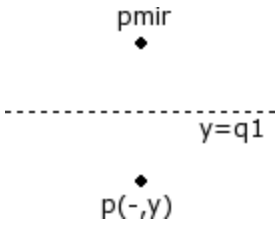
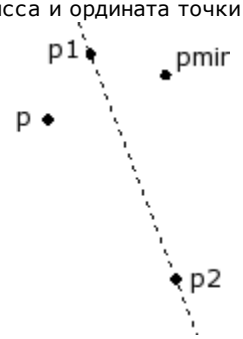
	ang: Если ang=0.0: возвращает нулевое значение. Примеры geo[larc;100;90] = 157.0796 geo[larc;100;360]=628.3185 □ соответствует всей окружности
geo[**larc;x**1;y1;xc;yc;ang]	Возвращает длину дуги круга, определенной начиная с точки P1 с прохождением угла ang:  x1;y1 = абсцисса и ордината точки P1 xc;yc = абсцисса и ордината центра круга Pc ang = угол прохождения Если точки P1 и Pc совпадают, или если ang=0.0: возвращает нулевое значение. Примеры geo[larc;0;0;100;0;90] = 157.0796 geo[larc;0;0;100;0;360]=628.3185 □ соответствует всей окружности
geo[**larc;**x1;y1;x2;y2;xc;yc;(sr)]	Возвращает длину дуги круга, определенной начиная с точки P1, с прохождением до P2, по или против часовой стрелки:  x1;y1=абсцисса и ордината точки P1 x2;y2 = абсцисса и ордината точки P2 xc;yc = абсцисса и ордината центра круга Pc sr = направление вращения на круге (#0 против часовой стрелки; 0=по часовой стрелке= по умолчанию, если не задано) Если точки P1 и Pc или P2 и Pc совпадают, возвращает нулевое значение. Примеры geo[larc;0;0;100;100;100;0] = 157.0796 geo[larc;0;0;100;100;100;0;1] = 471.238
geo[**larc;** "wname"]	Эта функция аналогична функции: geo[**larc**;x1;y1;x2;y2;xc;yc;(sr)]. Возвращает длину дуги круга, определенной именем операции:

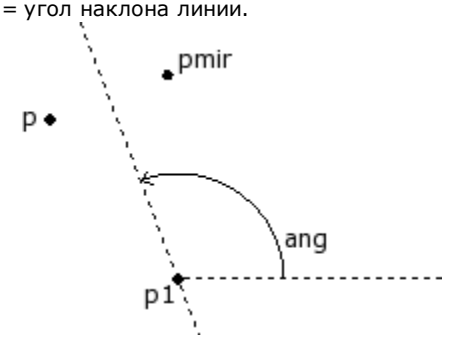
	▪ теперь операция может определить дугу круга, покоящегося на любой плоскости, или дугу конуса; ▪ P1 и P2 - это, соответственно, начальная точка и конечная точка дуги. Длина вычисляется на плоскости развертки дуги. Если операция определена неправильно, функция возвращает значение 0.0.

Функции вращения точки

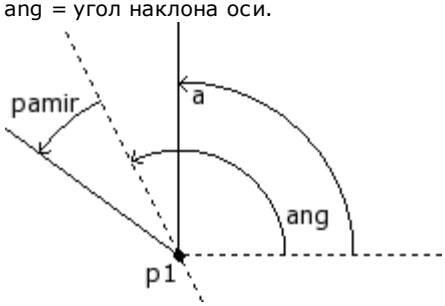
geo[**pxrot**;x;y;xc;yc;ang] geo[**pyrot**;x;y;xc;yc;ang]	Они возвращают абсциссу (x) или ординату (y) точки P, повернутой на угол ang, с центром Pc (дифференциальное вращение): x;y = абсцисса и ордината точки P xc;yc = абсцисса и ордината точки Pc ang = угол дифференциального вращения: • если ang>0 (положительный): точка вращается против часовой стрелки; • Если ang<0 (отрицательный): точка вращается по часовой стрелке. Если точка P совпадает с центром или если ang=0, функция возвращает изначальную координату.  Примеры geo[pxrot;70.7107;70.7107;0;0;-45] = 100 geo[pyrot;70.7107;70.7107;0;0;-45] = 0 geo[pxrot;100;0;0;0;90]=0 geo[pyrot;100;0;0;0;90]=100
geo[**pxrota**;x;y;xc;yc;ang] geo[**pyrota**;x;y;xc;yc;ang]	Они возвращают абсциссу (x) или ординату (y) точки P, повернутой на угол ang, с центром Pc (абсолютное вращение): x;y = абсцисса и ордината точки P xc;yc = абсцисса и ордината точки Pc ang = угол конечного вращения. Если точка P совпадает с центром, функция возвращает изначальную координату.  Примеры geo[pxrota;70.7107;70.7107;0;0;90] = 0 geo[pyrota;70.7107;70.7107;0;0;90] = 100

Функции симметрии

geo[**pmir**;q;q1]	Они возвращают координату зеркального отображения точки P вокруг вертикальной или горизонтальной оси: • если q имеет значение координаты x: ось симметрии вертикальная и имеет уравнение X = q1; ось симметрии вертикальна: имеет уравнение x=q1; q - абсцисса точки P (x=q)  • если q имеет значение координаты Y: ось симметрии горизонтальная и имеет уравнение Y = q1; ось симметрии горизонтальна: имеет уравнение y=q1; q - ордината точки P (y=q)  Пример: geo[pmir;100;500] = 900
geo[**pxmir**;x;y;x1;y1;x2;y2] geo[**pymir**;x;y;x1;y1;x2;y2]	Они возвращают абсциссу (x) или ординату (y) точки P в зеркальном отображении вокруг оси P1-P2: x;y = абсцисса и ордината точки P x1;y1 = абсцисса и ордината точки P1 x2;y2 = абсцисса и ордината точки P2.  Примеры: geo[pxmir;0;0;0;500;500;0]=500 geo[pymir;0;0;0;500;500;0]=500
geo[**pxmir**;x;y;x1;y1;ang] geo[**pymir**;x;y;x1;y1; ang]	Они возвращают абсциссу (x) или ординату (y) точки P в зеркальном отображении вокруг оси P1- ang: x;y = абсцисса и ордината точки P

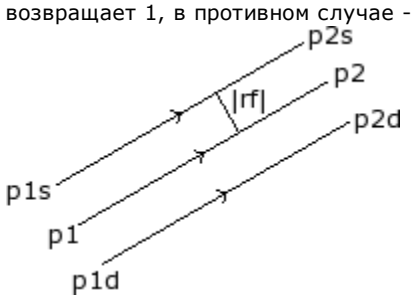
	x1;y1 = абсцисса и ордината точки P1 ang = угол наклона линии.  Примеры: geo[pxmir;0;0;0;500;-45]=500 geo[pymir;0;0;0;500;-45]=500
geo[**pxmir;**"wname1+nn",x1,y1,x2,y2] geo[**pxmir;**"wname1+nn",x1,y1,ang] geo[**pxmir**;x,y,"wname2+nn"] geo[**pxmir**;"wname1+nn","wname2+nn"] geo[**pymir;**"wname1+nn",x1,y1,x2,y2] geo[**pymir;**"wname1+nn",x1,y1,ang] geo[**pymir**;x,y,"wname2+nn"] geo[**pymir**;"wname1+nn","wname2+nn"]	Эти функции аналогичны функциям: • geo[**pxmir**;x;y;x1;y1;x2;y2], geo[**pymir**;x;y;x1;y1;x2;y2] • geo[**pxmir**;x;y;x1;y1;ang], geo[**pymir**;x;y;x1;y1;ang]. Они возвращают абсциссу (x) или ординату (y) точки P в зеркальном отображении вокруг оси. Точка P и/или ось задаются именем операции. • "wname1"=имя операции, задающей точку P. Может соответствовать точке или подготовке или отрезку профиля (линия или дуга: точка P - это конечная точка отрезка); • "wname2"=имя операции, задающей отрезок: она должна определять линейный отрезок, состоящий только из одного отрезка. Если операция "wname1" правильно не обнаруживается, функция возвращает значение 0.0. Если операция "wname2" правильно не обнаруживается, функция возвращает координату точки P.

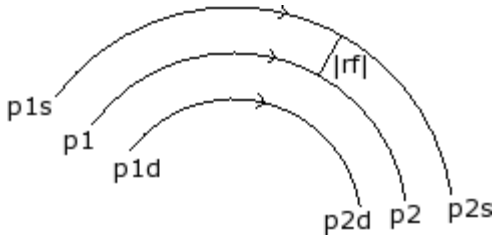
Функции вращения угла

geo[**pamir**;a;ang]	Возвращает угол (a) в зеркальном отражении вокруг оси ang: a = отражаемый угол ang = угол наклона оси.  Примеры geo[pamir;30;90] = 150
geo[**pamir**;a;x1;y1;x2;y2]	Возвращает угол (a) в зеркальном отражении вокруг оси P1-P2: a = отражаемый угол x1;y1 = абсцисса и ордината точки P1 x2;y2 = абсцисса и ордината точки P2. Примеры geo[pamir;30;0;0;0;100] = 150
geo[**pamir**;a;"wname+nn"]	Эта функция аналогична функции geo[**pamir**;a;x1;y1;x2;y2]. Возвращает угол (a) в зеркальном отражении вокруг оси, определенной именем операции. Эта операция должна определить линейный отрезок, состоящий только из одного отрезка. Если

	операция "wname" правильно не обнаруживается, функция возвращает значение a.

Функции коррекции геометрического элемента со сдвигом

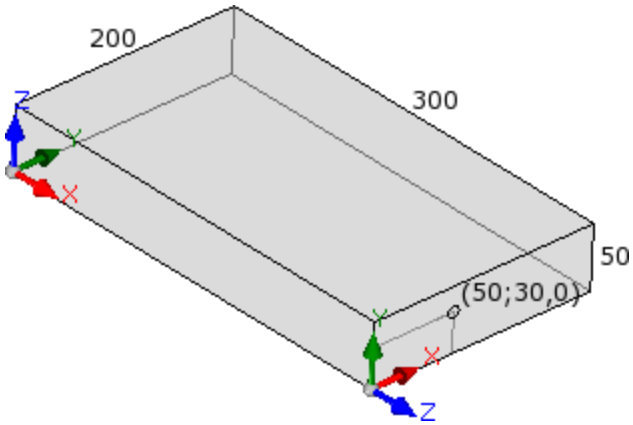
geo[**px1rf**;x1;y1;x2;y2;rf(;nret)] geo[**py1rf**;x1;y1;x2;y2;rf(;nret)] geo[**px2rf**;x1;y1;x2;y2;rf(;nret)] geo[**py2rf**;x1;y1;x2;y2;rf(;nret)]	Они возвращают абсциссу (x) или ординату (y) точки начала (1) или конца (2) линейного сегмента после применения коррекции к отрезку: x1,y1= абсцисса и ордината начальной точки отрезка P1 x2,y2= абсцисса и ордината конечной точки отрезка P2 rf=значение коррекции для применения к отрезку. • положительное значение выполняет корректировку к левой части отрезка, • отрицательное значение выполняет корректировку к правой части отрезка. nret=флаг запроса результата. При задании положительного значения если данные геометрии правильны, функция возвращает 1, в противном случае - 0.  Сегмент p1-p2 - это изначальный отрезок Если значение rf положительно, скорректированный отрезок - p1s-.p2s Если значение rf отрицательно, скорректированный отрезок - p1d-p2d В обоих случаях скорректированный отрезок имеет расстояние от изначального, равное абсолютному значению rf. Если координаты точек P1 и P2 совпадают, функция возвращает координату точки без применения **коррекции**. Примеры geo[px1rf;100;100;300;300;50] = 64.6447 geo[py1rf;100;100;300;300;50] = 135.3553 geo[px2rf;100;100;300;300;50] = 264.6447 geo[py2rf;100;100;300;300;50] = 335.3553
geo[**px1rf**;x1;y1;x2;y2;xc;yc;sr;rf;(nret)] geo[**py1rf**;x1;y1;x2;y2;xc;yc;sr;rf;(nret)] geo[**px2rf**;x1;y1;x2;y2;xc;yc;sr;rf;(nret)] geo[**py2rf**;x1;y1;x2;y2;xc;yc;sr;rf;(nret)]	Они возвращают абсциссу (x) или ординату (y) точки начала (1) или конца (2) дуги после применения коррекции к отрезку: x1,y1= абсцисса и ордината начальной точки отрезка P1 x2,y2= абсцисса и ордината конечной точки отрезка P2 xc, yc=абсцисса и ордината центра дуги sr = направление вращения круга (0= по часовой стрелке, в противном случае - против часовой стрелки). rf=значение коррекции для применения к отрезку. • положительное значение выполняет корректировку к левой части отрезка, • отрицательное значение выполняет корректировку к правой части отрезка nret=флаг запроса результата. При задании положительного значения если данные геометрии правильны, функция возвращает 1, в противном случае - 0.

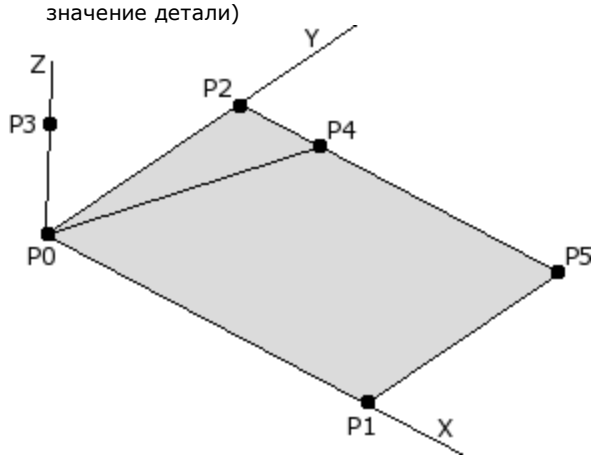
	 Сегмент p1-p2 - это изначальный отрезок Если значение rf положительно, скорректированный отрезок - p1s-.p2s Если значение rf отрицательно, скорректированный отрезок - p1d-p2d В обоих случаях скорректированный отрезок имеет расстояние от изначального, равное абсолютному значению rf. Если дуга неправильна (начальный радиус и конечный радиус разные) или внутренняя требуемая коррекция имеет значение больше радиуса дуги, функция возвращает координату точки без применения коррекции. Примеры geo[px1rf;0;0;200;0;100;0;0;50] = -50 geo[py1rf;0;0;200;0;100;0;0;50] = 0 geo[px1rf;0;0;200;0;100;0;0;-50] = 50 geo[py1rf;0;0;200;0;100;0;0;-50] = 0
geo[**px1rf;**"wname+nn";rf;(nret)] geo[**py1rf;**"wname+nn";rf;(nret)] geo[**px2rf;**"wname+nn",rf;(nret)] geo[**py2rf;**"wname+nn";rf;(nret)]	Они возвращают абсциссу (x) или ординату (y) точки начала (1) или конца (2) отрезка, определенного именем операции, после применения к нему коррекции. Отрезок может линейным сегментом или дугой. "wname"= имя операции, которая задает отрезок для применения коррекции rf=значение коррекции для применения к отрезку. • положительное значение выполняет корректировку к левой части отрезка, • отрицательное значение выполняет корректировку к правой части отрезка. nret=флаг запроса результата. При задании положительного значения если данные геометрии правильны, функция возвращает 1, в противном случае - 0. Выбранная операция должна определять дугу в плоскости xy или линейный отрезок, состоящий только из одного отрезка.

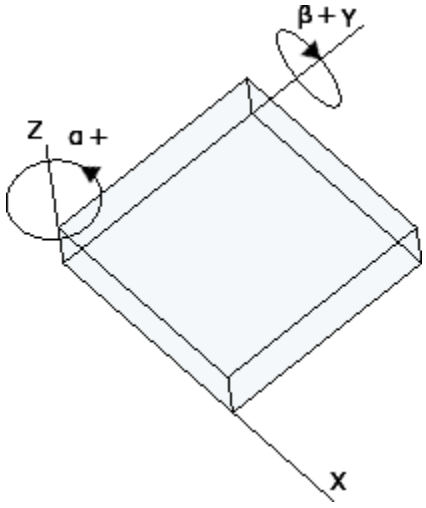
Функции преобразования координат и считывания информации граней

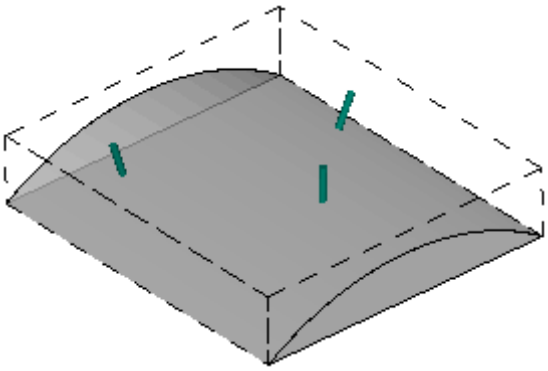
Для всех этих функций, при программировании операции в режиме "грань-деталь", с возможностью задания автоматических граней и поля имени, для параметров задания грани (например, *nside*) допускается следующий синтаксис:
• если *nside*=100, интерпретирует как автоматическую грань, заданную последней до текущей операции;
• распознается также форма с *nside*, замененной *"nameFace"*=имя автоматической грани, все также заданной последней до текущей операции с указанным именем.

geo[**pxp**;x;y;z;nside) geo[**pyp**;x;y;z;(nside)] geo[**pzp**;x;y;z;(nside)]	Дает координату (x; y; z) указанной точки, преобразованной локальной системой грани (nside) в абсолютные координаты детали: x;y;z = координаты точки в локальной системе грани nside = номер грани (если это реальная грань - это клиентский номер). Номер грани факультативен: если он не задан, принимает номер активизированной грани. Эта функция возвращает исходную координату, если: • nside задает неправильное значение грани;

	• nside не указан и активизированная грань - общее значение детали; • при задании переменных (o, v, r) переменной геометрии или клиентского раздела и *nside* указывает нереальную грань. Если *nside* указывает реальную грань, эта функция работает даже если грань не задана на детали. Рассмотрим рисунок:  • длина детали=300; высота детали=200; толщина детали=50 • на грани 4 указана точка с координатами 50;30;0 • в абсолютных координатах детали эта точка имеет координаты: 300;50;30 Примеры: geo[pxp;100;100;-5;1]: возвращает абсциссу точки (100;100;-5) из системы грани 1 в абсолютные координаты детали geo[pxp;100;100;-5]: возвращает абсциссу точки (100;100;-5) из системы активизированной грани в абсолютные координаты детали
geo[**pxf**;x;y;z;nside);(nsorg)] geo[**pyf**;x;y;z; (nside);(nsorg)] geo[**pzf**;x;y;z; (nside);(nsorg)]	Дает координату (x; y; z) указанной точки, преобразованной локальной системой грани (nsorg) в локальную систему грани (nside): x;y;z = координаты точки в локальной системе грани nsorg nside = номер грани-получателя (клиентский номер) (если задание пусто: использует =текущую грань) nsorg = номер исходной грани (клиентский номер) (если задание пусто: использует =общее значение детали) Допускаются уменьшенные формы: • с 4 аргументами. Пример: geo[**pxf**;x;y;z]: • nside: принимает активизированную грань; • nsorg: принимает общее значение детали (-1); • с 5 аргументами. Пример: geo[**pxf**;x;y;z; nside]: • nsorg: принимает общее значение детали (-1). В случае незаданного nsorg или заданного на неправильное значение: функция выполняет преобразование из абсолютных координат детали в локальную систему грани (nside). Эта функция возвращает исходную координату, если: • nside задает неправильное значение грани; • nside не указан и активизированная грань - общее значение детали; • при задании переменных (o, v, r) или переменной геометрии и nside указывает фиктивную грань. Если nside и/или nsorg указывает реальную грань: эта функция работает даже если грань не задана на детали. При программировании операции на грани-детали, если nside=100 или nsorg=100, учитывает последнюю заданную автоматическую грань до текущей операции. Примеры:

	geo[pxf;100;100;-5;1;7]: возвращает абсциссу точки (100;100;-5) из системы грани 7 в систему грани 1; geo[pxf;100;100;-5;;7]: возвращает абсциссу точки (100;100;-5) из системы грани 7 в систему активизированной грани; geo[pxf;100;100;-5;1]: возвращает абсциссу точки (100;100;-5) из абсолютных координат детали в систему грани 1; geo[pxf;100;100;-5]: дает абсциссу точки (100;100;-5) из абсолютных координат детали в систему активизированной грани;
geo[**px**;(np);(nside);(dd);(ndest)] geo[**py**; (np);(nside);(dd);(ndest)] geo[**pz**; (np);(nside);(dd);(ndest)]	Возвращает дает координату (x; y; z) ребра грани (nside) в координатах локальной системы грани (ndest): np = идентификационный номер ребра грани (если задание пусто: использует =0) • 0: начало координат грани (точка P0) • 1: точка по оси x+ (точка P1) • 2: точка по оси y+ (пересчитанная точка P2) • 3: точка по оси z, в направлении z-воздух (точка P3) • 4: изначальная точка P2 • 5: четвертая точка прямоугольника грани (точка P5) nside = номер грани для считывания ребра (клиентский номер) (если задание пусто: использует =текущую грань) dd = расстояние требуемой точки от P0 (незначимо, если np=0) (если задание пусто: не применяет) ndest = номер грани для считывания ребра (клиентский номер) (если задание пусто: использует = общее значение детали)  <u>Допускаются уменьшенные формы:</u> • с 2 аргументами. Пример: geo[**px**;np]: • nside: принимает активизированную грань; • dd: расстояние - по умолчанию (размер, соответствующий грани вдоль оси); • ndest: принимает общее значение детали (-1); • с 3 аргументами. Пример: geo[**px**;np;nside]: • dd: расстояние - по умолчанию (размер, соответствующий грани вдоль оси); • ndest: принимает общее значение детали (-1); • с 4 аргументами. Пример: geo[**px**;np;nside;dd]: • ndest: принимает общее значение детали (-1). Если np имеет неправильное значение: функция работает как с np=0. Если np=4: функция не интерпретирует dd. функция возвращает нулевую координату (=0.0), если: • nside задает неправильное значение грани; • nside не указан и активизированная грань - общее значение детали;

	• при задании переменных (o, v, r) переменной геометрии или клиентского раздела; *nside* указывает нереальную грань. Если nside указывает реальную грань, эта функция работает даже если грань не задана на детали. Если nside указывает грань, заданную как поверхность: • для точек (P0; P2; P4; P3): функция возвращает расчетные значения для первого геометрического элемента, задающего поверхность; • для точек (P1; P5): функция возвращает расчетные значения для последнего значимого геометрического элемента, задающего поверхность Примеры: geo[px;0]: возвращает абсциссу точки P0 активизированной грани, преобразованную в абсолютные координаты детали geo[px;0;7;;1]: возвращает абсциссу точки P0 грани 7, преобразованную в локальную систему грани 1
geo[**альфа**;(nside)] geo[**бета**;(nside)]	Они возвращают, соответственно, угол вращения (альфа) и горизонтального вращения (бета), для задания инструменту для работы перпендикулярно грани nside: nside = номер грани (клиентский номер) (если задание пусто: использует =текущую грань). Эта функция, в любом случае, возвращает 0, если: • nside задает неправильное значение грани; • nside указывает грань, не заданную в детали; • грань задана с неправильной геометрией; • при задании переменных (o, v, r) переменной геометрии или клиентского раздела; *nside* указывает нереальную грань. Если *nside* указывает реальную грань, эта функция работает даже если грань не задана на детали. Если *nside* указывает закругленную грань или поверхность: возврат определяется с учетом плоской вспомогательной грани. Если *nside* указывает грань, заданную как поверхность: функция возвращает расчетные значения для первого геометрического элемента, задающего поверхность  Со ссылкой на рисунок: • бета вращается вокруг оси y • альфа вращается вокруг оси z.

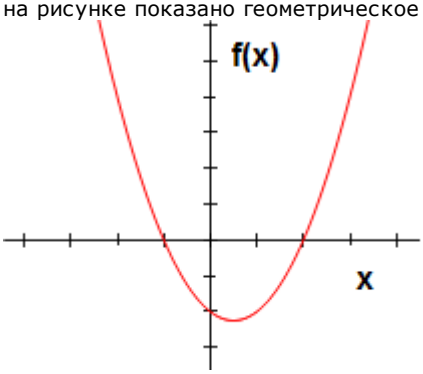
	Со ссылкой на шесть реальных граней параллелепипеда, значения для альфа и бета задаются следующим образом: Грань Бета; Альфа Примечания 1 (0;0) Альфа: любое 2 (180;0) Альфа: любое 3 (-90;90); (90;-90) 4 (-90;180); (90;0) 5 (-90;-90); (90;90) 6 (-90;0); (90;180) Если в конфигурации TpaCAD задается инверсия оси горизонтального вращения, приведенные в таблице значения горизонтального вращения должны менять знак на противоположный.
geo[**alfa**;x;y;z;(nside)] geo[**beta**;x;y;z;nside]	Функции аналогичны предыдущим с возможностью указать точку грани, относительно которой определять значения двух поворотных осей (альфа, бета) для присвоения инструменту для работы перпендикулярно грани nside: x;y;z = координаты точки в локальной системе грани (nside) nside = номер грани (клиентский номер) (если задание пусто: использует =текущую грань). Эта функция возвращает 0, если: • nside задает неправильное значение грани; • nside указывает грань, не заданную в детали; • грань задана с неправильной геометрией; • при задании переменных (o, v, r) переменной геометрии или клиентского раздела; *nside* указывает нереальную грань. Если *nside* указывает реальную или, в любом случае, плоскую грань: функции точно соответствуют форматам без координат точки, а в случае реальной грани работают также если грань не задана на детали. Если *nside* указывает закругленную грань или поверхность: возврат определяется с учетом положения, заданного тремя координатами. На рисунке видно, как вертикальное направление меняется при изменении положения на представленной закругленной грани 
geo[**lface**;(nside)] geo[**hface**;(nside)] geo[**sface**;(nside)] geo[**zface**;(nside)]	Они возвращают, соответственно: длину (lface), высоту (hface), толщину (sface) и ориентацию оси z (zface) грани nside: nside = номер грани (клиентский номер) (если задание пусто: использует =активированную грань). Эта функция, в любом случае, возвращает 0, если: • nside задает неправильное значение грани; • nside указывает фиктивную грань, не заданную в детали;

	• при задании переменных (o, v, r) переменной геометрии или клиентского раздела; *nside* указывает нереальную грань. Если *nside* указывает реальную грань, эта функция работает даже если грань не задана на детали.
geo[**rface**;(nside)] geo[**cxface**;(nside)] geo[**cyface**;(nside)] geo[**czface**;(nside)]	Возвращают особую информацию по закругленной грани. Соответственно: радиус изгиба (rface), координата центра изгиба (cxface; cyface; czface) грани nside: nside = номер грани (если задание пусто: использует =активированную грань). Эта функция, в любом случае, возвращает 0, если: • nside задает неправильное значение грани; • nside указывает незакругленную грань или заданную как поверхность; • при задании переменных (o, v, r) переменной геометрии или клиентского раздела; *nside* указывает нереальную грань.
geo[**plface**;(nside)]	Возвращают особую информацию по закругленной грани или поверхности: • плоскость кривизны (0=xz; 1=yz), если она закругленная грань • ось развития (0=x; 1=y), если поверхность. Эта функция, в любом случае, возвращает 0, если: • nside задает неправильное значение грани; • nside указывает незакругленную грань или поверхность.
geo[**isface**;(nside)]	Возвращает флаг наличия грани в детали (1=существует, 0=не существует): nside = номер грани (клиентский номер) (если задание пусто: использует =активированную грань). Эта функция возвращает 0, если: • nside задает неправильное значение грани; • nside указывает грань, не заданную в детали; • при задании переменных (o, v, r) переменной геометрии или клиентского раздела; *nside* указывает нереальную грань. Если *nside* указывает закругленную грань: функция возвращает 2. Если *nside* указывает грань, заданную как поверхность: функция возвращает 3.
geo[**simil**;(nModo);(nside)]	Возвращает номер реальной грани, которая соответствует критериям подобия для грани, определенной nside: nModo = критерий подобия. Если этот параметр не задан, значение по умолчанию - 0: • 0: проверяет, что инструмент имеет то же направление вертикального входа на грани. Грань (nside) генерируется переносом одной или нескольких осей с возможным вращением на плоскости xy • 1 (отлично от 0): грань (nside) генерируется простым переносом одной или нескольких осей nside = номер грани (клиентский номер). Если этот параметр не задан, используется номер текущей грани. Эта функция, в любом случае, возвращает 0, если: • nside задает неправильное значение грани • nside указывает фиктивную грань, не заданную в детали • nside указывает закругленную грань или заданную как поверхность • при задании переменных (o, v, r) переменной геометрии или клиентского раздела; *nside указывает нереальную грань.* Если получаемое значение значимо (отлично от 0), оно соответствует клиентскому номеру грани. Если nside задает значение реальной грани: функция возвращает nside.

geo[**pr1**;(nside)] geo[**pr2**;(nside)] geo[**pr3**;(nside)] geo[**pr4**;(nside)] geo[**pr5**;(nside)]	Они возвращают дополнительные параметры фиктивной или автоматической грани: nside = номер грани (клиентский номер). Если этот параметр не задан, значение по умолчанию - номер текущей грани. Эта функция возвращает 0, если: • nside задает неправильное значение грани; • nside указывает фиктивную грань, не заданную в детали; • при задании переменных (o, v, r) или переменной геометрии или клиентского раздела.

Алгебраические функции

Алгебраические функции - функции продвинутого программирования

geo[**equ**;a;b;c;(nret)]	Решает алгебраическое уравнение $\mathbf{ax^2+bx+c=0}$ a, b, c=коэффициенты уравнения nret = флаг запроса исхода (если не задан, равен 0): • при заданном нулевом значении возвращает число решений (корней) уравнения (0, если не принимает решений; 1, если принимает одно решение (когда a=0); в противном случае - 2); • задать 1 для получения первого решения, 2 - второго. Если недействительно ни одно решение, решениям присваивается значение 0.0. Если действительно только одно решение, решениям присваивается одинаковое значение. С геометрической точки зрения: • график функции второго графика $\mathbf{f(x)= ax^2+bx+c}$ на плоскости прямоугольных координат представляет собой параболу • решениями функции являются значения **x**, при которых график функции прикасается к оси X. Примеры geo[equ;1;-1;-2] = 2 □ уравнение имеет 2 решения geo[equ; 1;-1;-2;1] = 2 geo[equ; 1;-1;-2;2] = -1 на рисунке показано геометрическое значение примера 

Функции доступа к информации запрограммированной операции

Функции должны рассматриваться продвинутым программированием.

geo[**param**;"wname+nn";"pname";(nret)] geo[**param**;"wname+nn";pID; (nret)]
Возвращает значение информации или параметра запрограммированной операции: • "wname" = имя искомой операции до текущей операции. • "pname" =имя (ASCII) параметра. В соответствии правилами синтаксиса, имя должно быть заключено в двойные кавычки и записываться строчными буквами. • pID =числовой идентификатор параметра. Рекомендуется использовать форму с pID только опытным пользователям.

- nret = флаг запроса исхода (если не задан, равен 0). Если имеет положительное значение (1), функция возвращает:
  - 1, если поиск операции и параметра верен
  - в противном случае получается 0.

Для аргумента "pname" (имя ASCII параметра) предусматривается несколько интересных случаев:
- общая информация об операции:
  - "#cop" - рабочий код операции;
  - "#tip" считывает тип операции (0=точечная; 1=подготовка; 2=отрезок дуги; 3=линейный отрезок; 4=логическая; 5=сложная);
  - "#tips" считывает подтип операции;
  - "#tipt" считывает тип технологии операции;
  - "#prog" - номер по порядку в списке операций;
  - "#list" считывает число операций в перечне операции. Значение может быть значимым (т. е. >0), если операция сложная (подпрограмма, макропрограмма, STOOL) или если это множественный отрезок профиля (например, скругление, множественные дуги, конус);
  - "#vl" считывает значение свойства L;
  - "#vb" считывает значение свойства B;
  - "#vo" считывает значение свойства O;
  - "#vm" считывает значение свойства M;
  - "#vk" считывает значение свойства K;
  - "#vk1" считывает значение свойства K1;
  - "#vk2" считывает значение свойства K2.
- особая информация о сложной операции (подпрограмме или макропрограмме):
  - "#subxi", "#subyi", "#subzi" - соответствующая координата первой обработанной позиции;
  - "#subxe", "#subye", "#subze" - соответствующая координата последней обработанной позиции;
  - "#subxn", "#subxp" - положение, соответствующее координате x минимального/ максимального габарита операции;
  - "#subyn", "#subyp" - положение, соответствующее координате y минимального/ максимального габарита операции;
  - "#subzn", "#subzp" - положение, соответствующее координате z минимального/ максимального габарита операции.

Если функция используется в замечательном строковом формализме:
geo[param;"w1"; "x"]:
- "#name" считывает имя операции

Примеры
geo[param;"w1"; "x"] : возвращает значение координаты X операции "w1" в зависимости от программирования и задания (абсолютного/относительного,..)
geo[param;"w1";31] : возвращает значение параметра ID=31 операции "w1": это координата X центра дуги. Если "w1" соответствует дуге, запрограммированной по 3 точкам, функция считывает координату X, вычисленную для центра (использовать 32 для координаты Y)

geo[**lparam**;"wname+nn";"pname";nlist1;(nlist2);(nlist3);(nlist4);(nlist5);(nlist6);(nret)]
geo[**lparam**;"wname+nn";pID;nlist1;(nlist2);(nlist3);(nlist4);(nlist5);(nlist6);(nret)]

Возвращает значение информации или параметра операции, найденной в расширенном списке указанной операции:

- "wname" = имя искомой операции до текущей операции.
- "pname" =имя (ASCII) параметра. В соответствии правилами синтаксиса, имя должно быть заключено в двойные кавычки и записываться строчными буквами.
- pID =числовой идентификатор параметра. Рекомендуется использовать форму с pID только опытным пользователям.
- nlist1 =номер по порядку операции в расширенном списке "wname". Задать строго положительное значение (>0) не выше значения, возвращаемого функцией:
- geo[param; "wname"; "#list"]
  Этот аргумент обязателен и должен иметь действительное значение.

- nlist2=номер по порядку операции в расширенном списке операции, определенной при помощи:
  geo[lparam; "wname"; "#cop"; nlist1].
  Задать строго положительное значение (>0) не выше значения, возвращаемого функцией:
  geo[lparam; "wname"; "#list";nlist1]
- …
- nlist6=номер по порядку операции в расширенном списке операции, определенной при помощи:
  geo[lparam; "wname"; #cop";nlist1;nlist2;nlist3;nlist4;nlist5].
  Задать строго положительное значение (>0) не выше значения, возвращаемого функцией:

geo[lparam;“wname”;“#list”;nlist1;nlist2;nlist3;nlist4;nlist5]

Оценка аргументов (nlist2,.., nlist6) прерывается при первом не строго положительном значении, останавливая навигацию по расширенным спискам на предыдущем уровне.

- nret = флаг запроса исхода (если не задан, равен 0). Если имеет положительное значение (1), функция возвращает:
  - 1, если поиск операции и параметра верен
  - в противном случае получается 0.

Для аргумента “pname” (имя параметра ASCII) реализованы все значимые случаи, предусмотренные для функции geo[**param**;…].
Основное использование этой функции направлено на **Продвинутое использование Запрограммированных инструментов.**

Примеры
geo[lparam;”w1”;“x”;1]: возвращает значение координаты X первой операции в расширенном списке “w1”
geo[lparam;”w1”;“x”;1;2]: первая операция в расширенном списке “w1” должна иметь свой расширенный

список с минимум двумя операциями в списке и функция возвращает значение координаты X второй

операции в расширенном списке

geo[**sub**;”pname”;(nret)]
geo[**sub**; pID; (nret)]

Возвращает значение параметра или информацию об операции по применению подпрограммы или макропрограммы, использование которых подразумевается в тексте подпрограммы или макропрограммы:

- “pname” =имя (ASCII) параметра. В соответствии правилами синтаксиса, имя должно быть заключено в двойные кавычки и записываться строчными буквами.
- pID= числовой идентификатор параметра.
- nret = флаг запроса исхода (если не задан, равен 0). Если имеет положительное значение (1), функция возвращает:
  - 1, если поиск операции и параметра верен;
  - в противном случае получается 0.

Для аргумента “pname” (имя ASCII параметра) предусматривается несколько интересных случаев:

- ”#cop” - рабочий код операции;
- ”#tips” считывает подтип операции;
- ”#tipt” считывает тип технологии операции;
- ”#prog” - номер по порядку в списке операций.

Те же аргументы, заканчивающиеся цифрой '0' (например, “#cop0”), возвращают информацию по основному применению (т. е. в списке программы) подпрограммы или макропрограммы.
В этих случаях аргумент соответствует не имени параметра ASCII, а информации операции.

Если функция используется в замечательном строковом формализме (“*geo[sub;..]”):

- ”#name” считывает имя операции.
- ”#name0” считывает имя операции, соответствующей основному применению.

Эта функция работает при использовании при создании сложного кода.

Пример:

- программируется код SUB с вызовом подпрограммы SUB1 и точки применения (X=100; Y=200)
- в SUB1 программируется операция HOLE, применимая в:
  - X= 100+geo[sub;”x”]
  - Y= 50+geo[sub;”y”]

Отверстие SUB1 будет позиционироваться по (X=200; Y=250).

Пример:

- в предыдущем примере для кода SUB мы присваиваем имя = "aa" и исследуем некоторые программы в подпрограмме SUB1 (например, для частных переменных r):
  - geo[sub;”#cop”] -> возвращает рабочий код вызывающей операции SUB (=2010)
  - “*geo[sub;”#name”]” -> возвращает имя вызывающей операции SUB (=”aa”). Присвоение является значимым для строковой переменной r

geo[**sub**;”wname”;”pname”;(nret)]

geo[**sub**;"wname";pID;nret]

Как и в случае предыдущих форматов, возвращает значение параметра или информации операции, относящиеся к применению подпрограммы или макроса, и использование которых предназначено в тексте той же подпрограммы или макроса.

Разница между этими форматами заключается в наличии аргумента "wname". Интерпретация аргумента назначается на замечательные строки:

- "prcopsetup1", "prcopsetup2": соответственно идентифицируют первую и вторую операцию, указанную по имени в параметре PRCOPSETUP комплексного кода
- "prcoppoint1", "prcoppoint2": соответственно идентифицируют первую и вторую операцию, указанную по имени в параметре PRCOPPOINT комплексного кода.

Параметры (PRCOPSETUP, PRCOPPOINT) могут использоваться при определении сложных кодов для выполнения автоматической замены технологии, соответственно для точных операций или подготовки. Использование параметра PRCOPSETUP предлагает множество приложений в базовой базе данных и обычно соответствует параметру *Исходный элемент для Подготовки*: использование «prcopsetup2» (или «prcoppoint2») соответствует случаям, когда сложная обработка включает раздельное назначение двух технологий.

ВНИМАНИЕ: автоматическая замена технологии через параметры PRCOPSETUP (PRCOPPOINT) подразумевает, что макропрограмма (или подпрограмма) соблюдает некоторые правила записи. Для получения более подробной информации см. специальную документацию.

Для остальных аргументов применяется то же самое, что и для предыдущих форматов функции:

- "pname" =имя (ASCII) параметра. В соответствии с правилами синтаксиса, имя должно быть заключено в двойные кавычки и записываться строчными буквами.
- pID = числовой идентификатор параметра.
- nret = флаг запроса исхода (если не задан, равен 0). Если имеет положительное значение (1), функция возвращает:
  - 1 если поиск операции и параметра верен;
  - в противном случае получается 0.

Обращает на себя внимание тот факт, что аргумент *nret* является обязательным во втором формате. Отсутствие аргумента интерпретирует формализм *geo[**sub**;"pname";(nret)]*, соответствующий предыдущим форматам.

Для аргумента "pname" (имя ASCII параметра) предусматривается несколько интересных случаев:

- "#cop" считывает рабочий код операции;
- "#tips" считывает подтип операции;
- "#tipt" считывает технологический тип операции;
- "#prog" - номер по порядку в списке операций.

В этих случаях аргумент соответствует не имени параметра ASCII, а информации операции.

Если функция используется в замечательном строковом формализме ("*geo[sub;..]"):

- "#name" считывает имя операции.

Функция работает, если используется при разработке сложного кода.

Пример:
определяется сложный код, который вызывает подпрограмму SUB1 и управляет параметром PRCOPSETUP в строковом формате.
В SUB1 мы используем программирование

"geo[sub;"prcopsetup1";"#cop"]"

например, в частной переменной r мы оцениваем различные назначения для параметра PRCOPSETUP:

- "aa" -> функция "geo[sub,.." возвращает рабочий код операции, назначенной раньше с именем "aa", но только если это соответствует операции setup
- "tec\aa" -> функция "geo[sub,.." возвращает рабочий код операции, соответствующий указанной глобальной технологии (0, если она не назначена)
- "aa; bb" -> позволяет вам использовать "aa" через "prcopsetup1" и "bb" через "prcopsetup2".

Пример:
Продолжая пример и в предположении, что PRCOPSETUP="aa", вы можете увидеть, как читать параметры операции:

- geo[sub;"prcopsetup1";205] -> читает инструмент, запрограммированный в операции с именем "aa"

- geo[sub;”prcopsetup1”;”f”] -> читает запрограммированную в операции скорость ввода с именем “aa”.

Клиентские функции

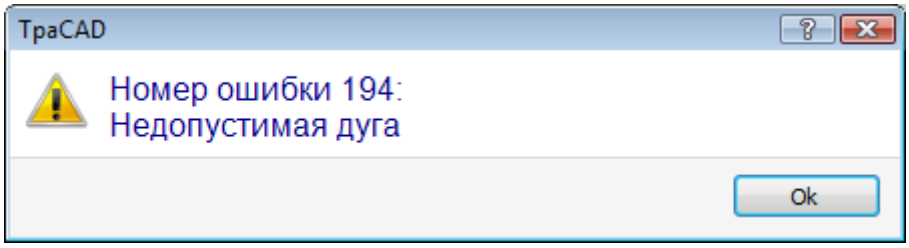


Клиентские функции доступны только в Режиме "Профессиональный".

funxxxxx[n1;..;n30]	Выполняет соответствующую клиентскую функцию, заданную персонализированным именем. Максимальное число аргументов - 30. Ситуации ошибки: • 123: число операндов =0; • 124: число операндов >30; • 134: слишком много встроенных вызовов клиентских функций (максимум: 5); • 135: использование клиентской функции в неправильном контексте (сигнализирует, что запрограммирован вызов клиентской функции, уже находящейся в стеке, или незаконное использование частной клиентской функции).

12 Сообщения об ошибке

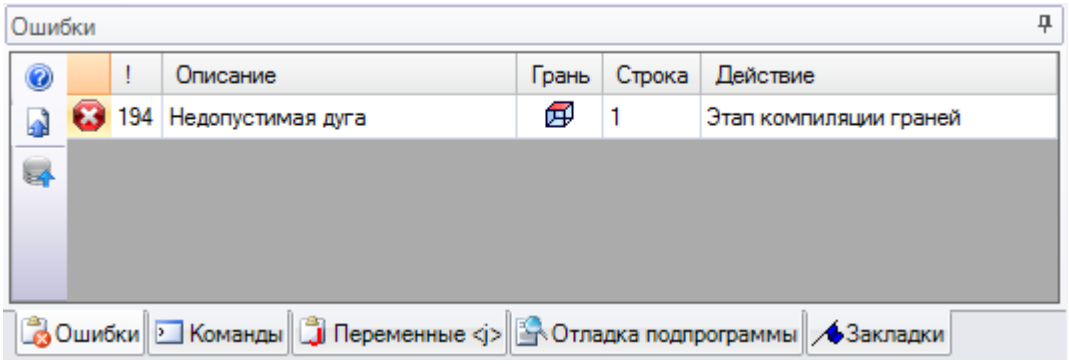
Представленное на рисунке сообщение отображается, например, при ошибке ввода или изменении операции типа дуги:



При нажатии кнопки **[?]** вызывается окно справки, в котором приводится описание ошибки.

В зоне отображения ошибок приводятся ошибки, которые были диагностированы в программе в целом, с указанием, помимо описания ошибки, также грани, программной строки и работ, выполнявшихся в момент, когда произошла ошибка.

При выборе значка **Перейти в соответствующее положение** выполняется переход к программной строке, в которой произошла выбранная ошибка.



12.1 Общие ошибки

Это ошибки, которые возникают во время или после подключение команд приложения. Они могут быть следующими:

- ошибки вследствие неудавшейся процедуры. Примером может быть неудавшаяся загрузка программы из файла.
- сообщения. Примером может быть запрос инструмента применительно к несоответствующим операциям.

1 - Ошибка процедуры

Объяснение:
общая ошибка, не определенная иным образом.

Контекст:
контекст может быть любым. В любом случае, необходимо уточнить, что различным ситуациям ошибки, как правило, соответствует определенная идентификация (т. е. подробная сигнализация ошибки).
Документация по этой общей ошибке, в любом случае, приводится для общего сведения, хотя количество ситуаций, в которых она подается, очень ограничено.

2 - Ошибка распределения памяти

Объяснение:
памяти системы недостаточно для выполнения запрошенной процедуры. Это серьезная ошибка: рекомендуется закрыть приложение и выполнить соответствующие проверки на системе.

Контекст:

любой

5 - Ошибка доступа к файлу

Объяснение:
произошла ошибка доступа к файлу (при считывании или записи). Это может указывать на: неправильный маршрут файла, неразрешенный доступ к файлу, пустой файл, неправильный формат файла.
Этот сигнал может указывать на проблему доступа к папке или отдельному файлу.
При попытке считывания или регистрации программы этот сигнал может соответствовать ситуации блокировки в результате использования программы другим приложением.

Контекст:
этот сигнал может быть следствием запроса:
- загрузки или записи программы
- загрузки или записи матрицы детали
- ввода операций в Локальный буфер (эта ошибка может обуславливаться созданием временного вспомогательного файла)
- загрузки или записи файла клиентских функций (при загрузке может также указывать на определение неправильного формата)

6 - Ошибка доступа к Локальным примечаниям

Объяснение:
произошла ошибка доступа к Локальным примечаниям системы, в результате чего оказалось невозможно сохранение или поиск информации. Это ошибка, связанная с неправильной работой системы.

Контекст:
этот сигнал может быть следствием запроса:
- ввода операции в Локальные примечания (команды изменения операций: Копировать, Удалить)
- поиска операций в Локальные примечания (команды изменения операций: Вставить; общие инструменты: Перенос, Вращение, Симметрично отобразить, Повторения, Разбивка)

7 - Ошибка доступа к временному файлу Undo

Объяснение:
произошла ошибка доступа к одному из временных файлов, созданных в поддержку управления функцией Undo. Эта ошибка может обуславливаться внешним вмешательством во временные файлы или ошибками доступа к периферийным записывающим устройствам. Это ошибка, связанная с неправильной работой системы.

Контекст:
этот сигнал может быть следствием выполнения любой команды модификации программы, предусматривающей возможность отмены:
- команд изменения операций: Правка, Вставить, Вклеить, Удалить, Селективные замены (параметры и/или свойства);
- инструменты

13 - Уровень системы не позволяет выполнить требуемую операцию

Объяснение:
подключенная команда не может быть исполнена, т. к. оператор имеет уровень доступа ниже того, который требуется командой.

Контекст:
этот сигнал может быть следствием запроса:
- загрузки макропрограммы (требуемый уровень доступа: Изготовитель)
- загрузки программы с уровнем доступа считывания выше заданного
- загрузки программы, которая применяла инструменты на профессиональном уровне в системе, работающей с ключом на стандартном уровне.
- регистрации программы с уровнем доступа записи выше заданного

18 - Текущая операция неправильна

Объяснение:
общая ошибка, связанная с применением команды к текущей операции.

Контекст:
этот сигнал может быть следствием запроса:

- команд изменения операций:
- Правка, Вставить, Вклеить, Удалить, Селективные замены (параметры и/или свойства)
- инструменты

36 - Достигнуто максимальное число операций для грани

Объяснение:
на текущей грани больше невозможно вставить операции, т. к. было достигнуто максимальное управляемое число (1000000).

Контекст:
этот сигнал может быть следствием запроса:
- загрузки программы
- применения подпрограммы (или макропрограммы) вследствие: избыточного числа считанных строк или избыточного числа строк вследствие применения повторений или выемки
- команд вставки операций: Вставить, Вклеить
- инструментов, требующих вставки операций вследствие избытка числа строк в результате применения инструмента

На этапе считывания программы эту ошибку можно исправить: лишние строки игнорируются.

38 - Невозможно вставить указанную операцию в текущей грани

Объяснение:
была запрошена вставка операции в грань, где такая операция не активирована.

Контекст:
этот сигнал может быть следствием запроса:
- вставки некоторых типов операций на Грань-Деталь. Обычно для каждой грани отключаются невыполняемые операции, а в режиме Грани-Детали все операции обычно активированы и при попытке вставки неактивированной операции отображается сообщение об ошибке.

39 - Инструмент не может использовать основную операцию

Объяснение:
инструмент не может быть активирован, т. к. среди сконфигурированных для применения операций отсутствует основная операция для работы. Это всегда операция профиля, элементарная по типу отрезка.
Элементарные коды профилей:
- L01 [код = 2201] для линейного отрезка
- A01 [код = 2101] для дуги, заданной в плоскости xy
- A05 [код = 2105] для дуги, заданной в плоскости xz
- A06 [код = 2106] для дуги, заданной в плоскости yz
- A10 [код = 2110] для дуги, заданной в плоскости xyz.

Контекст:
инструменты, изменяющие или генерирующие профили:
- все инструменты профиля (Разбить профиль, Сместить точку подготовки,…)
- продвинутые инструменты профиля (Создание текстов, Выемки зоны, Резка профилей, Построение профилей)

41 - Ошибки в присвоении свойства операции

Объяснение:
было введено неправильное значение свойства операции (Слой, Конструкции, Полей M, O, K, K1, K2). Например, нераспознанное параметрическое задание, значение, выходящее за минимальные и максимальные пределы задания.

Контекст:
Было введено неправильное значение на этапе изменения или ввода операции или на этапе комплексного задания свойств.

42 - Модификации или замены не были внесены

Объяснение:
подключенная команда не определила модификаций.

Контекст:
этот сигнал может быть следствием запроса:
- команд [изменения операций](): Изменить, Вставка, Селективные замены (параметры и/или свойства)
- Инструментов (общих, профиля, продвинутых профиля)

49 - Инструмент применяется только к профилям

Объяснение:
инструмент профиля был активирован для операций, не связанных с профилем.

Контекст:
этот сигнал может быть следствием запроса инструмента профиля с выбранными операциями, не относящимися к профилю.

281 - Чтение файла: неожиданный конец файла

Объяснение:
произошла ошибка при считывании программы. Это сообщение указывает, что файл был признан как файл с правильным форматом, но с неправильным синтаксисом.

Контекст:
был достигнут конец файла с разделом на этапе считывания. Этот сигнал может указывать на то, что в файл были внесены неразрешенные изменения или что его генерация не соблюдает требуемый синтаксис. Эту ситуацию можно исправить, принудительно задавая закрытие каждого открытого раздела и закрывая считывание файла.
Ниже приводится пример конечной части программы TCN: слева конец файла (указан как EOF) выполняется с разделом разомкнутой грани, в центральной колонке раздел завершен правильно

…	…	Открывает раздел грани
SIDE#1{	SIDE#1{	..
…	…	Последняя операция
#2201{ ::WTI #1=532.89 … }W	#2201{ ::WTI #1=532.89 … }W	
EOF	**}SIDE**	
	EOF	

282 - Чтение файла: закрытие раздела не найдено

Объяснение:
произошла ошибка при считывании программы. Это сообщение указывает, что файл был признан как файл с правильным форматом, но с неправильным синтаксисом.

Контекст:
Нет соответствия между строками заголовка и закрытия раздела файла. Этот сигнал может указывать на то, что в файл были внесены неразрешенные изменения или что его генерация не соблюдает требуемый синтаксис.

283 - Чтение файла: неправильный идентификатор грани

Объяснение:
произошла ошибка при считывании программы. Это сообщение указывает, что файл был признан как файл с правильным форматом, но с неправильным синтаксисом.

Контекст:
В частности: соответствует заданию раздела грани, указанной с неправильной нумерацией.
В сообщении может отображаться неправильный номер, как он считан файлом. Эту ситуацию можно исправить: соответствующий раздел игнорируется.

284 - Чтение файла: неправильный идентификатор операции

Объяснение:
произошла ошибка при считывании программы. Это сообщение указывает, что файл был признан как файл с правильным форматом, но с неправильным синтаксисом.

Контекст:
Наблюдается, когда при считывании файла с кодировкой ASCII встречается операция, которая не существует в базе данных операций, или идентификатор, не действительный для считывания файла с внутренней кодировкой.

В первом случае операция вызывается именем ASCII (например, HOLE, SETUP), а во втором случае - рабочим кодом (например, 81, 88).
В сообщении об ошибке может отображаться название операции, как она считывается из файла.
Это ситуация исправима: если в базе данных имеется команда NOP (Not Operation), она задается для данной программной строки, в противном случае эта строка удаляется

286 - Чтение файла: ошибка декодировки файла

Объяснение:
произошла ошибка при считывании программы. Этот сигнал указывает, что файл был распознан как файл зашифрованного формата, но на этапе декодировки произошла ошибка.

287 - Чтение файла: программа не совместима с назначениями среды

Объяснение:
произошла ошибка при считывании программы. Сигнал указывает, что файл содержит неуправляемые программирования на основе текущей конфигурации и соответствует ситуациям, которые могут существенно изменить назначение самой программы. В частности запрограммированы:

- поверхности
- кривые лица

при отсутствии специального разрешения.

12.2 Специфические ошибки при применении инструментов

Это ошибки, которые сообщаются при подключении инструментов приложения. Сигнал указывает, что инструмент не сработал.

50 - Инструмент не обработал трансформации

Объяснение:
инструмент, активированный на основе заданных параметров, не определил каких-либо изменений.

Контекст:
этот сигнал может быть следствием запроса:

- инструментов (общих или профиля);
- инструмента Выемки зоны: не были обнаружены замкнутые зоны.

51 - Инструмент применяется к простому профилю

Объяснение:
инструмент профиля был активирован для непростого профиля

Контекст:
этот сигнал может быть следствием запроса инструментов:

- Линеаризировать глубину профиля
- Применить точки крепления к профилю
- Прерывание профиля
- Замыкание открытого профиля
- Создать маршрут для коррекции на инструмент
- Создание кривых сплайн

53 - Минимизировать профиль: угол сокращения превышает значение 90.0°

Объяснение:
параметр угла сокращения присвоил значение более 90°.

Контекст:
этот сигнал может быть следствием запроса инструмента:

- Минимизировать профиль

54 - Фрагментировать профиль: максимальная длина отрезков равна нулю

Объяснение:
неправильное значение параметра заданной максимальной длины фрагментации (< 5.0 * эпсилон)

Контекст:
этот сигнал может быть следствием запроса инструмента профиля Фрагментировать профили .

55 - Применить точки крепления профиля: неправильное число точек крепления [минимум: 2; максимум: 255]

Объяснение:
заданное число креплений не находится в пределах от 2 до 255.

Контекст:
этот сигнал может быть следствием запроса инструмента профиля Применить точки крепления профиля в режиме автоматического распределения или соответствующих запрограммированных инструментов (коды STOOL).

56 - Применить точки крепления профиля: неправильная длина точек крепления или чрезмерная компенсация на инструмент

Объяснение:
длина креплений задана на нулевое значение (< эпсилон); или при активированном флаге "Компенсация на инструмент" заданная длина креплений имеет значение меньше диаметра инструмента

Контекст:
этот сигнал может быть следствием запроса инструмента Применить точки крепления профиля.

59 - Применить точки крепления профиля: остаточная толщина точек крепления неправильна или не задана

Объяснение:
остаточная толщина точек крепления задана на нулевое значение (< эпсилон)

Контекст:
этот сигнал может быть следствием запроса инструмента Применить точки крепления профиля.

60 - Применить точки крепления профиля: невозможно распределить точки крепления по профилю (уменьшить число точек крепления)

Объяснение:
длина профиля недостаточна для распределения всех точек крепления в указанном количестве. Решением проблемы является задание меньшего числа точек крепления.

Контекст:
этот сигнал может быть следствием запроса инструмента Применить точки крепления профиля с запросом автоматического распределения точек крепления.

61 - Инверсия профиля: получены сложные неинвертируемые коды

Объяснение:
текущий профиль, который необходимо инвертировать, задан со сложными кодами (подпрограммами и/или макропрограммами), которые:
- не могут быть уподоблены профилю и не управляют параметром инверсии; или
- которые не могут быть инвертированы, т. к. они должны отвечать ограничениям, заданным при присвоении базы данных операций для приложения (сами или применяют операции, для которых была исключена возможность инверсии исполнения)

Контекст:
этот сигнал может быть следствием запроса инструмента профиля Инвертировать профили.

62 - Применить инструмент: сложный код конца профиля не завершается с отрезком профиля

Объяснение:
текущий инвертируемый профиль завершается сложным кодом (подпрограммой и/или макропрограммой), развертка которого не завершается отрезком профиля.

Контекст:
этот сигнал может быть следствием запроса инструмента профиля Инвертировать профили, Применить вход к профилю (с выбором добавления отрезка перекрытия в конце).

63 - Сместить подготовку на профиль: указанное положение совпадает с текущей подготовкой

Объяснение:
положение, указанное для смещения точки подготовки, нашло на профиле точку, совпадающую с текущим положением подготовки (разница между значениями значительна, если больше эпсилон)

Контекст:
этот сигнал может быть следствием запроса инструмента профиля Сместить подготовку на замкнутый профиль.

64 - Инструмент применяется к замкнутому профилю

Объяснение:
текущий профиль не является замкнутым профилем. Начальная точка должна совпадать с конечной точкой (разница между значениями значительна, если больше эпсилон)

Контекст:
этот сигнал может быть следствием запроса инструмента профиля Сместить подготовку на замкнутый профиль, Применить вход к профилю (с выбором добавления отрезка перекрытия в конце).

67 - Скругление или фаска профиля: заданный радиус равен нулю

Объяснение:
заданный радиус или фаска равны нулю (< эпсилон);

Контекст:
этот сигнал может быть следствием запроса инструмента профиля Применить скругления к профилю, Применить фаски к профилю, Создать маршрут для коррекции на инструмент или соответствующих запрограммированных инструментов (коды STOOL).

68 - Резать профиль: указанное положение уже совпадает с подготовкой

Объяснение:
указанное положение для резки профиля нашло на профиле точку совпадения, совпадающую с текущим положением подготовительной операции (разница между значениями значительна, если больше эпсилон)

Контекст:
этот сигнал может быть следствием запроса инструмента профиля Разрезать профиль.

69 - Разрезать профиль: указанное положение уже завершает профиль

Объяснение:
указанное положение для резки профиля нашло на профиле точку совпадения, совпадающую с конечным положением профиля (разница между значениями значительна, если больше эпсилон)

Контекст:
этот сигнал может быть следствием запроса инструмента профиля Разрезать профиль.

70 - Начало/ Конец профиля: не задана исходная операция

Объяснение:
инструмент не может быть активирован, т. к. среди сконфигурированных для применения операций отсутствует основная операция для работы. Это всегда операция профиля, элементарная по типу отрезка. Необходимые элементарные коды:
- COPL01 для линейного отрезка;
- COPA17 для кругового отрезка в начале профиля;
- COPA16 для кругового отрезка в конце профиля;

Контекст:

этот сигнал может быть следствием запроса инструментов профиля Применить вход к профилю, Применить выход к профилю.

71 - Применить инструмент: невозможно зацепиться перед профилем

Объяснение:
невозможно задать открытие профиля или зацепление перед текущим профилем, т. к.:
- профиль начинается со сложного кода (подпрограммы или макропрограммы), который, однако, не может быть уподоблен профилю при открытии своей развертки или
- профиль начинается со сложного кода (подпрограммы или макропрограммы), который не управляет параметром зацепления или который не может быть зацеплен, т. к. он должен соблюдать ограничения, определенные при задании базы данных операций для приложения

Контекст:
этот сигнал может быть следствием запроса инструментов профиля Применить подготовку, Применить многократную подготовку, Применить вход к профилю.

72 - Начало профиля: не задано перемещение для точки начала профиля

Объяснение:
положение, указанное для смещения точки подготовки, совпадает с текущим положением подготовки (разница между значениями значительна, если больше эпсилон)

Контекст:
этот сигнал может быть следствием запроса инструмента профиля Применить вход к профилю.

73 - Конец профиля: не задано перемещение для точки конца профиля

Объяснение:
положение, указанное для смещения конечной точки профиля, совпадает с текущим положением конца профиля (разница между значениями значительна, если больше эпсилон)

Контекст:
этот сигнал может быть следствием запроса инструмента профиля Применить выход к профилю.

75 - Соединение профилей: второй профиль определен неправильно

Объяснение:
не было выявлено какого-либо профиля с геометрической непрерывностью по сравнению с первым указанным

Контекст:
этот сигнал может быть следствием запроса инструмента профиля Соединение между соседними профилями.

78 - Соединение профилей: профили разъединены

Объяснение:
указанные профили не имеют геометрической непрерывности, позволяющей обеспечить их соединение в один профиль (разница между значениями значительна, если больше эпсилон)

Контекст:
этот сигнал может быть следствием запроса инструмента профиля Соединение между профилями.

79 - Масштабировать профиль: получаются сложные неизменяемые коды

Объяснение:
текущий профиль задан со сложными кодами (подпрограммами и/или макропрограммами), которые не могут быть изменены выбранным инструментом:
- не уподобляемые профилю и не управляющие параметрами масштаба; или
- которые не могут быть уменьшены или увеличены, т. к. они должны отвечать ограничениям, заданным при присвоении базы данных операций для приложения (сами или применяют операции, для которых была исключена возможность масштабирования исполнения)

- которые выполняют дуги на плоскостях, отличных от xy, и требуется ограниченный масштаб только для плоскости xy

Контекст:
этот сигнал может быть следствием запроса инструмента профиля Масштабировать профиль.

80 - Масштабировать профиль: коэффициент увеличения или уменьшения не задан или равен 1.0

Объяснение:
заданный масштабный коэффициент равен 1.0 или не задан.

Контекст:
этот сигнал может быть следствием запроса инструмента профиля Масштабировать профиль.

82 - Инструмент запрашивает слишком много повторений

Объяснение:
было запрошено слишком большое число повторений: например, общее число вводимых повторений не может быть больше 1000.

Контекст:
этот сигнал может быть следствием запроса инструментов:
- Повторение операций, Прямоугольная серия операций, Круговая серия операций, при числе повторений больше 100000
- Повторения на профиле при числе повторений больше 100000 или если расстояние, вычисленное между последующими повторениями, слишком маленькое (< 10.0 эпсилон координат)

и запрограммированных инструментов (коды STOOL):
- Применить подачу по Z со слишком большим числом проходов (максимум: 1000).

85 - Применить инструмент: профиль задает дуги в плоскости, отличной от xy

Объяснение:
активированный инструмент не может работать на одном или нескольких выбранных профилях вследствие наличия на этих профилях дуг, заданных в плоскости, отличной от xy

Контекст:
этот сигнал может быть следствием запроса инструментов:
- Масштабировать профиль, с запросом масштаба, ограниченного только плоскостью xy
- Линеаризировать глубину профиля
- Создать маршрут для коррекции на инструмент
- Создание кривых сплайн

86 - Конец профиля: невозможно зацепить конец после профиля

Объяснение:
невозможно задать зацепление после этого профиля, т. к. профиль завершается сложным кодом (подпрограммой или макропрограммой) который:
- не может быть уподоблен профилю при закрытии своей развертки; или
- не управляет параметром зацепления или не может быть зацеплен, т. к. он должен соблюдать ограничения, определенные при задании базы данных операций для применения;

Контекст:
этот сигнал может быть следствием запроса инструмента профиля Применить выход к профилю.

88 - Применить инструмент: невозможно задать подготовку из-за отсутствия исходного кода

Объяснение:
активированный инструмент не мог работать вследствие невозможности присвоения профилю исходного кода подготовки.

Контекст:
этот сигнал может быть следствием запроса инструментов, работающих на профилях. Присвоение исходного код подготовки выполняется на этапе конфигурации TpaCAD.

92 - Инструмент не запросил движения по осям

Объяснение:
заданные параметры не вызывают какого-либо переноса.

Контекст:
этот сигнал может быть следствием запроса инструмента Перенос.

93 - Инструмент запросил нулевой поворот

Объяснение:
заданные параметры не вызывают какого-либо вращения.
Может быть, был задан нулевой угол относительного вращения или же был задан абсолютный угол вращения с относительным центром на нуле.

Контекст:
этот сигнал может быть следствием запроса инструмента Вращение.

94 - Инструмент не запросил повторных применений

Объяснение:
общее число вводимых повторений недостаточно.

Контекст:
этот сигнал может быть следствием запроса инструментов:
Повторение операций с требуемым числом повторений меньше 1
Прямоугольная серия операций с общим числом повторений (строки*столбцы) меньше 1
Круговая серия операций с числом исполняемых элементов меньше 2
Этот сигнал может быть следствием использования запрограммированных инструментов (коды STOOL):
Применить подачу по Z с шагом подачи меньше (эпсилон * 10) или нулевым числом проходов.

95 - Создать текст: надпись была обрезана до максимально допустимой длины для построения кривой

Объяснение:
случай распределения надписи вдоль закругленного геометрического элемента (дуге круга или конуса), с суммарной длиной исходной строки, превышающей длину всей кривой, круга или конуса.
Надпись была создана до максимальной развертки, разрешенной для этого геометрического элемента.
Этот сигнал обрабатывается как предупреждение.

Контекст:
этот сигнал может быть следствием запроса продвинутого инструмента Создание текстов или программирования операции создания текста.

96 - Создать текст: неправильное коническое сечение развертки

Объяснение:
дуга конуса (овал или эллипс) для распределения текста задана неправильно.

Контекст:
этот сигнал может быть следствием запроса продвинутого инструмента Создание текстов

98 - Создать текст: высота символов недостаточна (минимум = eps * 100)

Объяснение:
было задано слишком маленькое значение высоты символов. Это значение не может быть меньше (эпсилон * 100).

Контекст:
этот сигнал может быть следствием запроса *п*родвинутого инструмента Создание текстов.

99 - Создать текст: неправильная дуга

Объяснение:
дуга для распределения текста задана неправильно: радиус равен нулю или различны начальный и конечный радиусы.

Контекст:
этот сигнал может быть следствием запроса продвинутого инструмента Создание текстов.

294 - Выемка зоны: профиль не замкнут

Объяснение:
была запрошена выемка незамкнутой зон/зон.

Контекст:
этот сигнал может быть следствием запроса продвинутого инструмента Выемки зоны.

295 - Выемка зоны: профиль не соответствует заданному инструменту

Объяснение:
была запрошена выемка несоответствующей замкнутой зоны (пустая зона) или такой, которая не допускает даже частичного первого круга выемки с заданной технологией.

Контекст:
этот сигнал может быть следствием запроса продвинутого инструмента Выемки зоны.

296 - Выемка зоны: радиус инструмента задан на ноль [минимум: 10*эпсилон]

Объяснение:
было задано неправильное значение радиуса коррекции при выемке недействительной зоны (< 10.0*эпсилон)

Контекст:
этот сигнал может быть следствием запроса:
- продвинутого инструмента Выемки зоны;
- применения цикла выемки (применение подпрограммы или макропрограммы).

297 - Выемка зоны: перекрытие превышает радиус инструмента

Объяснение:
заданное значение перекрытия превышает значение радиуса коррекции

Контекст:
этот сигнал может быть следствием запроса:
- продвинутого инструмента Выемки зоны
- применения цикла выемки (применение подпрограммы или макропрограммы).

298 - Выемка зоны: отклонение по глубине идет от Z=0.0

Объяснение:
начальное и конечное значения по глубине заданы с противоположным знаком в выемке профиля при выполнении последующих проходов

Контекст:
этот сигнал может быть следствием запроса:
- продвинутого инструмента Выемки зоны
- применения цикла выемки (применение подпрограммы или макропрограммы).

299 - Выемка зоны: неправильная координата Z воздуха

Объяснение:
начальное значение глубины и значение координаты в воздухе заданы с одинаковым знаком

Контекст:

этот сигнал может быть следствием запроса:
- продвинутого инструмента Выемки зоны
- применения цикла выемки (применение подпрограммы или макропрограммы).

300 - Выемка зоны: слишком много профилей для оценки (>300)

Объяснение:
была запрошена выемка профилей, определившая оценку слишком большого числа замкнутых зон (максимум = 300).

Контекст:
этот сигнал может быть следствием запроса:
- продвинутого инструмента Выемки зоны
- применения цикла выемки (применение подпрограммы или макропрограммы).

12.3 Ошибки параметрического программирования

Это сигналы неправильных настроек параметрического задания.
При ошибке этого типа соответствующее выражение решается заданием:
- значения 0.0, в случае числовой переменной или параметра;
- решенной строки, совпадающей с запрограммированной строкой, в случае нечисловой переменной или параметра

Для анализа ситуаций подачи различных ошибок см. главу, в которой рассматривается Параметрическое программирование.

101 - Параметрическое программирование: слишком длинная строка

Объяснение:
было написано выражение, состоящее из слишком большого числа символов. Максимальное число символов - 100.

102 - Параметрическое программирование: неправильный синтаксис

Объяснение:
синтаксис, использованный в параметрическом программировании, неправильный.
Ниже приводятся некоторые правила программирования, которые могут помочь истолковать ошибку синтаксиса:
- допускаются символы значения между пробелом (' ') и закрытой фигурной скобкой ('}');
- пробел может использоваться только при задании строки (переменной или параметра типа строки или аргумента типа строки). Примеры:
  - "**strcmp[r5;"pippo 1"]**" - правильно
  - "**120+ 12**" определяет ошибку синтаксиса
- символ " используется для задания прямых сообщений (начинает и закрывает сообщение). Пример: strcmp[r5;"pippo"];
- символ ' используется для задания значения символа (начало и конец). Пример: 120+'a';
- использование синтаксиса применительно к символическим именам должно задать:
  - непустое символическое имя: "o\" определяет ошибку;
  - длина имени не должна превышать 16 символов: "o\abracadabraaaaaaaaaaaa" определяет ошибку;
  - должна присваиваться переменная типа, указанного приведенным именем: "o\aaa" определяет ошибку синтаксиса, если не задана переменная "o" с символическим именем "aaa".
- использование функций или переменных, относящихся к составлению клиентских функций, определяет ошибку синтаксиса, если они используются в программе

Пример неправильного синтаксиса:
Примеры неправильного синтаксиса:
"100+16-" -> становится действительным при исправлении на "100+16"
"32*(r0+r3" -> становится действительным при исправлении на "32*(r0+r3)"
"abs(r5)" -> становится действительным при исправлении на "abs[r5]"

"o\aaa" -> если нет переменной "o" с символическим именем "aaa"

103 - Параметрическое программирование: переменная <r>, вызванная по имени, не найдена

Объяснение:
используется переменная <r> с символическим именем, которое не было задано.
В зависимости от конфигурации TpaCAD это может быть серьезная или несерьезная проблема (предупреждение): в случае предупреждения применяемой переменной задается нулевое числовое значение (0.0).

105 - Параметрическое программирование: численное значение превышает допустимый диапазон (-3.4E+30; 3.4E+30)

Объяснение:
значение, вычисленное для числового выражения, превышает интервал, разрешенный для значения с запятой

106 - Параметрическое программирование: решение параметра в виде слишком длинной строки (макс.: 260 символов)

Объяснение:
строка, полученная в результате параметрического задания, превышает макс. число символов, допустимых для параметра в типа строки, которое составляет 260 символов

109 - Параметрическое программирование: неправильный контекст для использования аргументов подпрограммы

Объяснение:
ошибка, связанная с использованием переменных аргументов, связанных с применением подпрограммы или макропрограммы: subx, suby,…., subface.

Контекст:
неправильные контексты использования:
- присвоение переменной 'o', 'v';
- присвоение клиентских функций;
- присвоение переменных геометрий (ребра фиктивных граней) или клиентских разделов.

111 - Параметрическое программирование: неправильный контекст для использования переменных <$>

Объяснение:
ошибка, связанная с использованием переменных или функций, связанных с переменными <$>: $0-$299, p$[.], min$[.], max$[.], ave$[.], sum$[.].

Контекст:
неправильные контексты использования:
- присвоение переменной 'r', 'o', 'v';
- присвоение клиентских функций;
- присвоение переменных геометрий (ребра фиктивных граней) или клиентских разделов;
- текст программы.

112 - Параметрическое программирование: неправильный контекст для использования переменных <r>

Объяснение:
ошибка, связанная с использованием переменных или функций, связанных с переменными <r>: r0-r299, pr[.], minr[.], maxr[.], aver[.], sumr[.], strlen, getat[.],strcmp[.], toolex[.]. tooltip[.].

Контекст:
неправильные контексты использования:
- присвоение переменной 'o', 'v';
- присвоение клиентских функций.

113 - Параметрическое программирование: неправильный контекст для использования переменных <v>

Контекст:
неправильные контексты использования:
- присвоение переменной 'o', 'v';
- присвоение клиентских функций.

114 - Параметрическое программирование: неправильный контекст для использования переменных <o>

Контекст:
неправильные контексты использования:
- присвоение переменной 'o', 'v';
- присвоение клиентских функций.

115 - Параметрическое программирование: неправильный контекст для использования переменных <j>

Объяснение:
ошибка, связанная с использованием переменных или функций, связанных с переменными <J>: j0-J99, pj[.], minj[.], maxj[.], avej[.], sumj[.].

Контекст:
неправильные контексты использования:
- присвоение переменной 'o', 'v';
- присвоение клиентских функций;
- присвоение переменных геометрий (ребра фиктивных граней) или клиентских разделов.

116 - Параметрическое программирование: неправильный контекст для использования имени операции

Объяснение:
ошибка, связанная с использованием функции геометрической библиотеки с синтаксисом, в котором используется имя операции.

Контекст:
неправильные контексты использования:
- присвоение переменной 'o', 'v';
- присвоение клиентских функций;
- присвоение переменных геометрий (ребра фиктивных граней) или клиентских разделов.

117 - Параметрическое программирование: неправильный индекс переменной <r>

Объяснение:
указан или вычислен неправильный индекс для переменной <r>: правильные значения находятся в интервале от 0 до 299. Однако в программировании переменной <r> интервал значений более ограниченный: переменная <r> может использовать только переменные меньшего индекса; тогда, например, r10 может использовать r9, но не r11.

Примеры:
Примеры неправильного программирования:
"r400", "pr[400]", "*pr[400]" максимальный индекс переменной r - 299
"r20" используемый, например, при задании переменной r10

118 - Параметрическое программирование: неправильный индекс переменной <j>

Объяснение:
указан или вычислен неправильный индекс для переменной <j>: правильные значения находятся в интервале от 0 до 99.

119 - Параметрическое программирование: неправильный индекс переменной <$>

Объяснение:
указан или вычислен неправильный индекс для переменной <$> : правильные значения находятся в интервале от 0 до 299.

120 - Параметрическое программирование: неправильный индекс переменной <v>

Объяснение:
указан или вычислен неправильный индекс для переменной <v>: правильные значения находятся в интервале от 0 до 15 с максимальным приемлемым значением, которое зависит от числа переменных <v>, которыми может управлять приложение:
- максимальное значение равно 15 при 16 управляемых переменных
- максимальное значение равно 6 при 7 управляемых переменных
- …
- неправильно ни одно значение без какой-либо управляемой переменной <v>.

121 - Параметрическое программирование: неправильный индекс переменной <o>

Объяснение:
указан или вычислен неправильный индекс для переменной <o>: правильные значения находятся в интервале от 0 до 15 с максимальным приемлемым значением, которое зависит от числа переменных <o>, которыми может управлять приложение:
- максимальное значение равно 15 при 16 управляемых переменных
- максимальное значение равно 6 при 7 управляемых переменных
- …
- неправильно ни одно значение без какой-либо управляемой переменной <o>

122 - Параметрическое программирование: функция имеет слишком много операндов (максимум: 30)

Объяснение:
была вызвана функция с числом операндов более 30, что является максимальным пределом операндов, управляемых одной функцией.

123 - Параметрическое программирование: функция без операндов

Объяснение:
была вызвана функция с несколькими операндами без операндов. Пример неправильного программирования: "ifcase[]"

124 - Параметрическое программирование: функция с ошибочным числом операндов

Объяснение:
была вызвана функция с несколькими операндами с неправильным числом операндов. Пример неправильного программирования: "ifelse[r0;l/2]

125 - Параметрическое программирование: деление на ноль

Объяснение:
в математической операции было выполнено деление на ноль. Эта ошибка является следствием использования математических операторов деления (/, %, #, ?)

126 - Параметрическое программирование: значение тригонометрической функции (sin, cos) не попадает в интервал от -1 до +1

Объяснение:
была выполнена обратная тригонометрическая функция (asin, acos), если операнд не попадает в интервал от –1.0 до +1.0.

127 - Параметрическое программирование: квадратный корень из отрицательного числа

Объяснение:
была выполнена функция sqrt расчета квадратного корня с отрицательным операндом

128 - Параметрическое программирование: возведение в степень с неправильным показателем [минимум = 0; максимум = 10]

Объяснение:
была выполнена функция pown возведения в степень со 2-м операндом (показателем степени) не в интервале от 0 до 10.

129 - Параметрическое программирование: неправильная функция геометрической библиотеки

Объяснение:
была выполнена универсальная функция геометрической библиотеки geo[..] с 1-м операндом, не соответствующим правильному имени. Пример неправильного программирования - "geo[aaa;l/2]": "aaa" не соответствует правильному имени

130 - Параметрическое программирование: функция с опущенным обязательным аргументом

Объяснение:
был опущен обязательный аргумент функции. Примеры неправильного программирования: "primp[;100.0]" опущен 1-й аргумент, считающийся обязательным

132 - Параметрическое программирование: неправильный угол для расчета тангенса

Объяснение:
ошибка обуславливается исполнением тригонометрической функции tan при присвоении операнду (углу) неправильного значения для расчета тангенса. Присвоенный угол, уменьшенный до числового интервала (0° - 360°), не может иметь значение 90° или 270°, т. к. в этом случае расчет касательной лишается смысла.

134 - Параметрическое программирование: слишком много встроенных вызовов клиентских функций (максимум: 5)

Объяснение:
число встроенных вызовов клиентских функций больше 5. Эта ошибка может быть только следствием неправильного задания клиентских функций на этапе конфигурации приложения.

135 - Параметрическое программирование: неправильное использование клиентской функции

Объяснение:
сигнализирует, что запрограммирован вызов клиентской функции, уже находящейся в стеке, или незаконное использование частной клиентской функции

136 - Параметрическое программирование: неправильное использование аргументов arg# res# var#

Объяснение:
ошибка, связанная с использованием аргументов, отведенных для клиентских функций в программе

137 - Параметрическое программирование: неправильный индекс или имя аргумента arg#

Объяснение:
указан или вычислен неправильный индекс для переменной <arg> или не представляется заданной ни одна переменная с указанным символическим именем. Эта ошибка может сигнализироваться только на этапе записи клиентских функций

138 - Параметрическое программирование: неправильный индекс или имя аргумента res#

Объяснение:
указан или вычислен неправильный индекс для переменной <res> или не представляется заданной ни одна переменная с указанным символическим именем. Эта ошибка может сигнализироваться только на этапе записи клиентских функций

139 - Параметрическое программирование: ошибка из-за вызова клиентской функции

Объяснение:
возникла ошибочная ситуация, диагностированная на уровне клиентской функции. Более подробно см. в документации, посвященной клиентским функциям, задаваемым на этапе конфигурации приложения.

140 - Параметрическое программирование: ошибка в использовании функций, отведенных для клиентских функций

Объяснение:
были использованы функции, отведенные для клиентских функций в программе

141 - Параметрическое программирование: неправильный индекс аргумента var#

Объяснение:
указан или вычислен неправильный индекс для переменной <var> или не представляется заданной ни одна переменная с указанным символическим именем. Эта ошибка может сигнализироваться только на этапе записи клиентских функций

12.4 Ошибки при обработке изменяемых геометрий

Это сигналы ошибки, связанные с заданием изменяемых Геометрий.

22 - Невозможно удалить грань, если для нее заданы операции

Объяснение:
невозможно удалить фиктивную грань, на которой были запрограммированы операции.

Действия:
удалить все операции, запрограммированные на фиктивной грани, а затем удалить грань.

144 - Изменяемые геометрии: исходная грань неверна или не задана

Объяснение:
исходной грани был присвоен неправильный номер. Это всегда номер фиктивной грани (более 6) и может означать, что указанная грань:

- не задана
- имеет номер, больший или равный текущей фиктивной грани
- имеет неправильную геометрию (точки не различны или на одной линии).

145 - Изменяемые геометрии: не все вершины грани различны

Объяснение:
не все три вершины определяемой фиктивной или автоматической грани заданы различными.

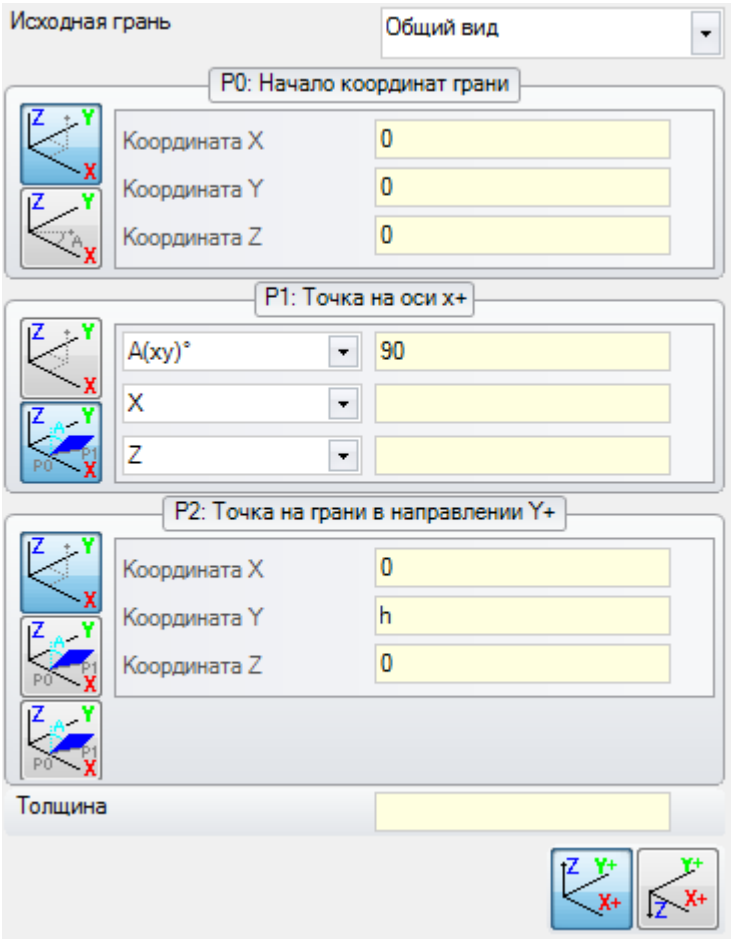
146 - Изменяемые геометрии: вершины грани находятся на одной линии

Объяснение:
три вершины определяемой фиктивной или автоматической грани заданы на одной линии.

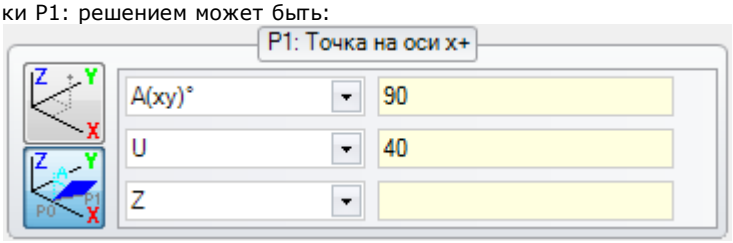
147 - Изменяемые геометрии: неправильная полярная геометрия грани

Объяснение:
произошла ошибка геометрического типа при задании ребра грани в полярных координатах. В частности, эта ошибка сигнализируется при возникновении условия вектора полярной системы, вычисленного бесконечным.

Пример:
Случай неправильного программирования соответствует заданиям:



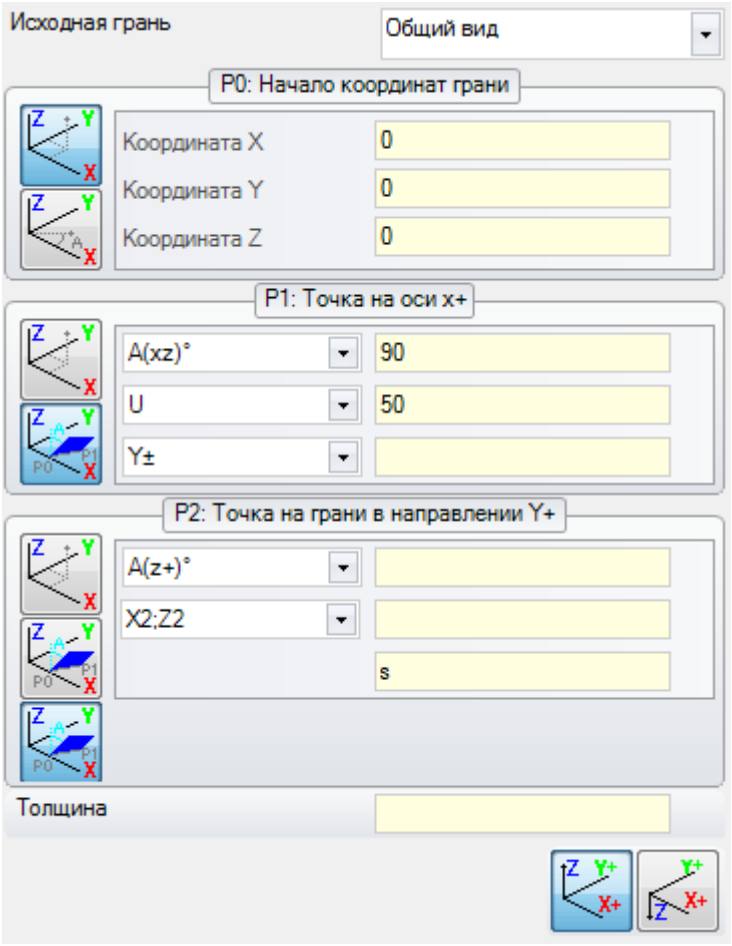
Ошибка касается точки P1: решением может быть:



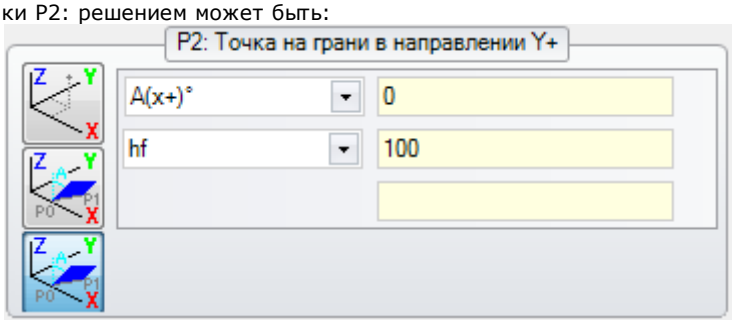
148 - Изменяемые геометрии: неправильная плоскость вращения

Объяснение:
произошла ошибка при задании плоскости вращения грани (точки не различны или на одной линии).

Пример:
Случай неправильного программирования соответствует заданиям:



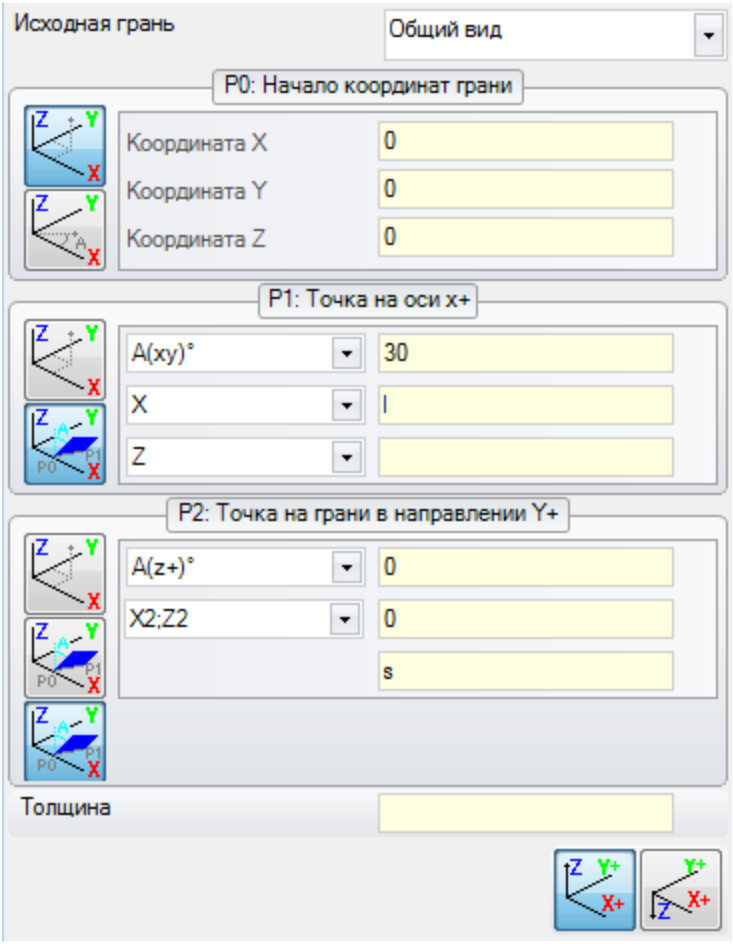
Ошибка касается точки P2: решением может быть:



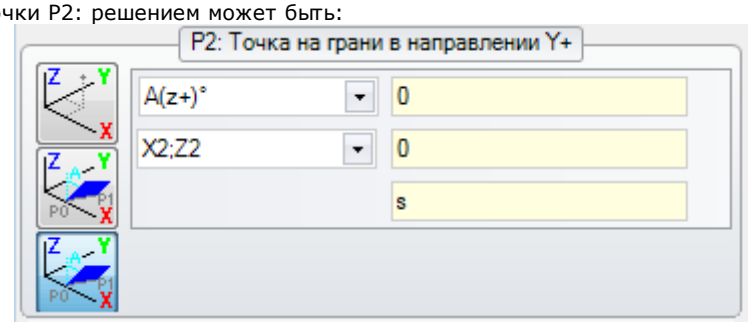
149 - Изменяемые геометрии: невозможно задать третью точку грани

Объяснение:
произошла ошибка при задании третьего ребра при помощи плоскости вращения грани

Пример:
Случай неправильного программирования соответствует заданиям:



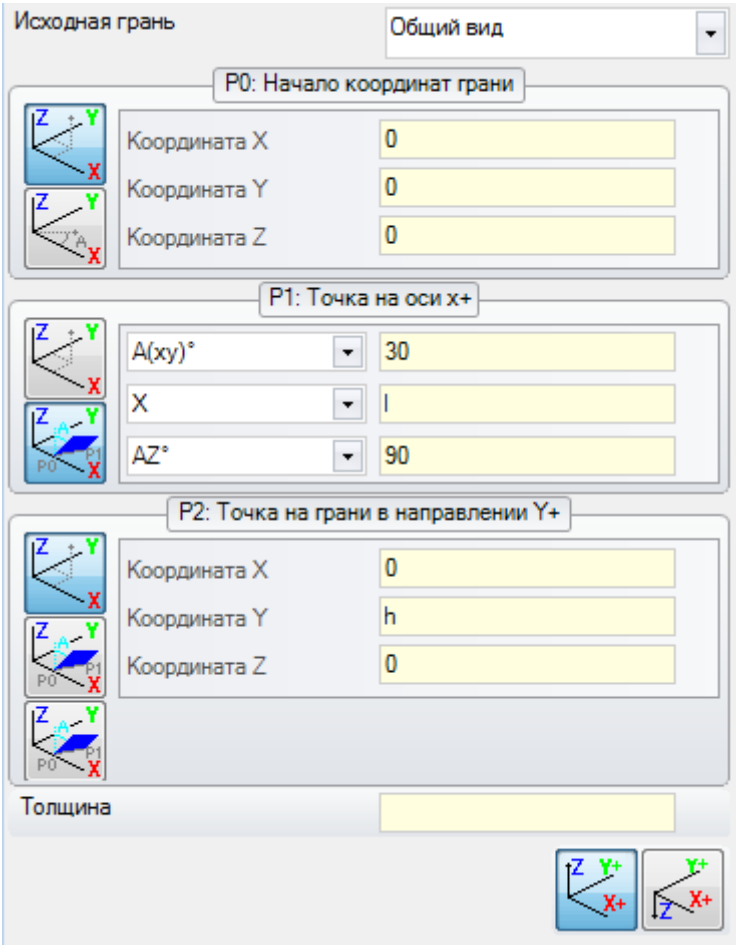
Ошибка касается точки P2: решением может быть:



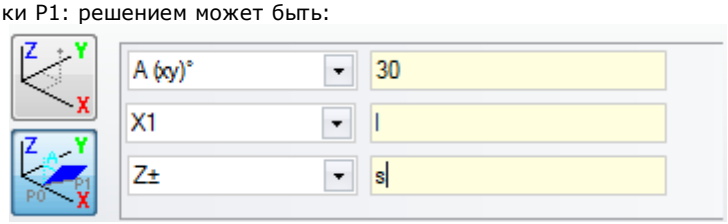
150 - Изменяемые геометрии: неправильная глубина точки

Объяснение:
произошла ошибка при задании координаты третьей оси точки в полярных координатах. В частности, эта ошибка возможна при задании координаты в режиме углового приращения, если угол имеет абсолютное значение, равное 90°.

Пример:
Случай неправильного программирования соответствует заданиям:



Ошибка касается точки P1: решением может быть:



165 - Изменяемые геометрии: неправильный радиус закругления грани

Объяснение:
был задан ненулевой радиус закругления грани (радиус принимается нулевым, если <=эпсилон координаты *2.0), но меньше расстояния между точками P0 и P1.

166 - Изменяемые геометрии: ошибка в решении геометрии поверхности

Объяснение:
невозможно решить геометрическую непрерывность между фиктивными гранями.

167 - Изменяемые геометрии: максимальное число элементов на поверхности

Объяснение:
на поверхности больше невозможно вставить другие элементы, т. к. было достигнуто максимальное управляемое число (300).

12.5 Ошибки на этапе компиляции программы грани

Это ошибки, возникающие при компиляции программы грани. Только в некоторых случаях это серьезные ошибки, не допускающие выполнения программы.
Всегда принимается решение по умолчанию (подробное рассмотрение каждой отдельной ситуации см. в документации по операциям).

151 - Неправильный код операции <имя рабочего кода>

Объяснение:
указанный рабочий код (взамен <имя рабочего кода>) не задан в базе данных операций, заданной для приложения, или же он задан с другой типологией.

Действия:
- соответствующая операция решена распространением рабочих координат с предыдущей операции
- это **серьезная ошибка**, не позволяющая выполнять программу:
- для устранения ошибки необходимо заменить данную обработку другой правильной (например, заменой рабочего кода).

Контекст:
эта ошибка может возникнуть при считывании программы с файла

152 - Параметр <имя параметра>: неправильное значение

Объяснение:
указанному параметру (взамен <имя параметра>) было присвоено значение, не входящее в интервал, определенный при конфигурации

Действия:
- это **серьезная ошибка**, не позволяющая выполнять программу

153 - Параметр <имя параметра>: задать формат "$nn"

Объяснение:
в команде FOR не был правильно задан первый терм одного из трех выражений

Контекст:
Эта ошибка может возникнуть только при программировании макропрограммы.

Пример:
Пример правильного задания:

FOR (**$0** = r1 to **$0** <= r2; **$0** = $0+r3)
{
.
} ENDFOR
где первый терм каждого выражения выделен жирным шрифтом. Для каждого терма, как требуется, используется форма $0 (не обязательно, чтобы всегда указывалась одна и та же переменная).

Пример неправильного задания:

FOR (**$0** = r1 to **$0+5** <= r2; **$0** = $0+r3)
{
.
} ENDFOR
где неправильное задание подчеркнуто.

155 - Свойство <имя поля>: недопустимое значение

Объяснение:
указанному свойству было присвоено значение, не входящее в интервал, определенный в конфигурации.
<Имя поля> может быть заменено на:
"L" для Слоя
"B" для Конструкции
"O", "M", "K", "K1", "K2" для соответствующего поля

Действия:

если было задано отрицательное значение, свойству задается значение 0. Если заданное значение превышает максимальное значение, заданное в конфигурации, свойству задается максимально допустимое значение.

156 - Поле <имя поля>: значение не соответствует заданному минимуму

Объяснение:
этот сигнал относится к полю в клиентском разделе. Он приводится в группе общих ошибок программирования по подобию сигнала. Указанному полю было задано значение ниже минимального, определенного в конфигурации. В любом случае, это несерьезная проблема (предупреждение).

Действия:
в любом случае, заданное значение сохраняется. Учитывая особенности клиентских разделов, все решения по задаваемым полям определяются при последующем толковании (т. е. на этапе оптимизации).

157 - Поле <имя поля>: значение превышает заданный максимум

Объяснение:
этот сигнал относится к полю в клиентском разделе, точно так, как предыдущая ошибка. Указанному полю было задано значение, превышающее максимальное, определенное в конфигурации. В любом случае, это несерьезная проблема (предупреждение).

Действия:
в любом случае, заданное значение сохраняется. Учитывая особенности клиентских разделов, все решения по задаваемым полям определяются при последующем толковании (т. е. на этапе оптимизации).

158 - Моделирование: код или последовательность кодов неправильна

Объяснение:
этот сигнал относится к строке в разделе моделирования и указывает неправильный код или неправильную последовательность кодов.

Действия:
в любом случае, заданное значение сохраняется. Для стирания сигнала ошибки необходимо удалить или изменить соответствующий раздел.

161 - Автоматических граней слишком много или они недоступны

Объяснение:
невозможно задать начало координат автоматической грани вследствие слишком большого числа созданных граней (макс. число: 400) или потому, что они не представляются заданными.

Действия:
- Соответствующая операция решена без задания автоматической грани
- это **серьезная ошибка**, не позволяющая выполнять программу
- для устранения ошибки необходимо изменить программирование (удалить операцию или уменьшить общее число заданных автоматических граней или использовать неавтоматическую исходную грань).

Контекст:
Этот сигнал может подаваться при обработке кода операции, задающего автоматическую грань.

162 - Поле F: недопустимое значение

Объяснение:
было задано неправильное значение свойству операции Грань применения.

Действия:
- Это **серьезная ошибка**, не позволяющая выполнять программу:
- для устранения ошибки необходимо изменить программирование.

Контекст:
Эта ошибка может возникнуть только при программировании Грани детали и может соответствовать одному из следующих случаев задания:
- автоматическая грань с неразрешенным режимом
- автоматическая грань, которая не представляется заданной

- неавтоматическая грань (реальная или фиктивная), которая не представляется заданной.

190 - Операции за пределами применения (ось <имя оси>)

Объяснение:
создание операции превышает зону или длину применения. Сообщение указывает ось или оси, для которых был подключен сигнал.

Действия:
- Если сигнал соответствует **ошибке**, выполнение программы невозможно.

Контекст:
Этот сигнал может подаваться при применении операции SUBNEST, если создание операций находится за пределами запрограммированной зоны габарита.

12.6 Ошибки в операциях профиля

192 - Расчетный радиус бесконечен

Объяснение:
было вычислено бесконечное значение радиус-вектора полярной системы.

Контекст:
Это распространяется на следующие операции:
- L04 [код = 2204]
- L05 [код = 2205]
- L06 [код = 2206]
- L07 [код = 2207]

193 - Нулевой радиус

Объяснение:
было вычислено нулевое значение для радиуса полярной системы или дуги. Этот сигнал обрабатывается как **предупреждение**, а не как ошибка.

194 - Неправильная дуга

Объяснение:
дуга была задана неправильно или недостаточно (центр не задан, различны начальный и конечный радиусы).

195 - Неправильная секущая

Объяснение:
неправильно задана секущая (не задана или точки не различны, или неправильна геометрически). Секущая должна задаваться:
- двумя различными точками или
- одной точкой и одним углом.

196 - Неправильная касательная в начале

Объяснение:
неправильно определена касательная в начале (не задана или неправильна геометрически). Секущая должна задаваться:
- углом или
- двумя различными точками.

Этот сигнал обрабатывается как **предупреждение**, а не как ошибка.

197 - Неправильная касательная в конце

Объяснение:
неправильно задана касательная в конце (не задана или неправильна геометрически). Напоминаем, что секущая должна задаваться:
- углом или
- двумя различными точками.

Этот сигнал обрабатывается как **предупреждение**, а не как ошибка.

198 - Расчетная точка вне отрезков

Объяснение:
в фаске или в скруглении решение находится за пределами изначально запрограммированного отрезка.

199 - Несуществующая точка пересечения

Объяснение:
сигнал в случае двойных дуг, если не находится решение.

200 - Неправильная дуга (точки не различны)

Объяснение:
дуга задана неправильно, причина состоит в совпадении точек дуги и/или точек с центром. Случаи ошибки:
- дуга, заданная тремя точками: три точки не различны;
- дуга, заданная двумя точками и центром: центр совпадает с точкой дуги

201 - Неправильная дуга (точки на одной прямой)

Объяснение:
при определении дуги точками точки были заданы на одной прямой. Если дуга имеет пространственную развертку, эта ошибка сигнализирует также случаи круга или начальной, конечной точки и центра на одной прямой.

202 - Овал: неправильный радиус

Объяснение:
при конфигурации овала меньший радиус задан большим или равным меньшей полуоси.
Этот сигнал обрабатывается как **предупреждение**, а не как ошибка.

203 - Овал превратился в круг

Объяснение:
при построении овала две полуоси определены одинаковыми. Этот сигнал обрабатывается как **предупреждение**, а не как ошибка.

204 - Овал: ось/оси нулевые или неправильные

Объяснение:
при построении овала одна или обе полуоси нулевые (разница между значениями значительна, если больше эпсилон).

205 - Эллипс/Овал: начальная точка вне габаритов конуса

Объяснение:
при построении овала или эллипса начальная точка оказывается за пределами габаритного прямоугольника, заданного для конического сечения.

206 - Прямоугольник: неправильная ось/оси или радиус

Объяснение:
при построении прямоугольника одна или обе оси заданы нулевыми (разница между значениями значительна, если больше эпсилон) или радиус скругления задан таким, что он выходит за габариты прямоугольника.

207 - Многоугольник: неправильное число сторон

Объяснение:
в определении операции многоугольника было задано недействительное число сторон: принимается значение от 3 до 99. В любом случае, это несерьезная проблема (предупреждение): недействительное значение приводится в указанный интервал.

12.7 Ошибки в подпрограмме или макропрограмме

208 - Запрограммированный инструмент: совпадение не найдено

Объяснение:
случай соответствующего кода типа STOOL (запрограммированный инструмент), чтобы указать, что действительная обработка не была идентифицирована. Этот сигнал обрабатывается как **предупреждение**, а не как ошибка.
Ситуации ошибки:
- в поле **Имя** запрограммированного Инструмента ничего не было назначено
- имена, указанные в поле **Имя**, не учитываются при запланированной операции на начальном этапе.
- операция применяемого инструмента не действует.

209 - Неправильное применение зашифрованной программы

Объяснение:
заданная подпрограмма (или макропрограмма) не соблюдает критерии, определенные для криптографии.
Ситуации ошибки:
- файл не соответствует макропрограмме;
- код применения - общего типа SUB;
- файл не имеет соответствия с подписью базы данных клиентских операций;
- файл имеет атрибуты, не соответствующие настройке в базе данных операций.

Контекст:
Этот сигнал может означать одну из следующих ситуаций:
- криптография подпрограммы выполнена после использования файла;
- текущая структура приложения не выдала подписи, соответствующей базе данных клиентских операций;
- файл, соответствующий зашифрованной программе, был взломан вручную.

210 - Неправильное имя подпрограммы

Объяснение:
имя подпрограммы (или макропрограммы) задано неправильно. Ситуации ошибки:
- задано неправильными символами: "#%;/\;"
- задано более, чем с одним символом "."

211 - Подпрограмма не существует

Объяснение:
подпрограмма (или макропрограмма) не существует или не может быть считана.

212 - Указанный файл имеет неправильный формат для подпрограммы

Объяснение:
указанная подпрограмма (или макропрограмма) имеет неправильный формат. Этот сигнал может также указывать, что делается попытка применить макропрограмму с общим кодом подпрограммы, не имея на это права.

213 - Неправильный номер грани

Объяснение:
было запрошено применение грани с неправильным идентификатором (номер меньше 1 или больше 99).

Контекст:
Этот сигнал может означать одну из следующих ситуаций:
- задан номер грани меньше 1 или больше 99;
- в программе грани детали:
  - на автоматической грани применен вложенный вызов подпрограммы;
  - операция SSIDE (операция запрограммированного вложенного вызова) определила неправильную грань применимой подпрограммы;
  - операция SSIDE (операция запрограммированного вложенного вызова) определила неправильную грань для применения.

214 - Исходный технологический элемент не применен

Объяснение:
случай сложного кода с параметром задания операции технологии по имени с нереализованной операцией. Этот сигнал может указывать, что не была найдена операция, точечная или подготовки, с указанным именем или что операция недействительна по компиляции или рабочему коду.
Этот сигнал обрабатывается как предупреждение.

216 - Не удалось прочитать подпрограмму

Объяснение:
была выявлена ошибка на этапе прочтения подпрограммы (или макропрограммы)

217 - Не задано имя подпрограммы

Объяснение:
не было задано имя подпрограмме (или макропрограмме)

218 - Создание кривой не применимо

Объяснение:
оказалось невозможным создать кривую сплайн, так как не были опознаны профили, к которым применять трансформацию. Если не требуется удалить первоначальные операции, этот сигнал обрабатывается как **предупреждение**, а не как ошибка.

219 - Выемка не применима

Объяснение:
оказалось невозможным применить выемку, так как не были опознаны профили, к которым ее применить. Если не требуется удалить первоначальные операции, этот сигнал обрабатывается как **предупреждение**, а не как ошибка.

220 - Вращение не применимо

Объяснение:
оказалось невозможным применить требуемое вращение вследствие:
- внутренних пределов развертыванию подпрограммы (или макропрограммы): развертывание, в свою очередь, применяет подпрограмму (или макропрограмму), для которых трансформация не допускается (в конфигурации базы данных операций);
- того, что для этого же приложения не допускается трансформация (в конфигурации);
- развертывания подпрограммы (или макропрограммы) включает дуги, заданные на плоскостях, отличных от xy, но не сконфигурирована вспомогательная операция A10 [код = 2110].

221 - Инверсия не применима

Объяснение:
оказалось невозможным применить требуемую инверсию вследствие:
- внутренних пределов развертыванию подпрограммы (или макропрограммы): развертывание, в свою очередь, применяет подпрограмму (или макропрограмму), для которых трансформация не допускается (в конфигурации базы данных операций);
- того, что для этого же приложения не допускается трансформация (в конфигурации).

222 - Зеркальное отражение по x не применимо

Объяснение:
оказалось невозможным применить требуемую симметрию вследствие:
- внутренних пределов развертыванию подпрограммы (или макропрограммы): развертывание, в свою очередь, применяет подпрограмму (или макропрограмму), для которых трансформация не допускается (в конфигурации базы данных операций);
- того, что для этого же приложения не допускается трансформация (в конфигурации).

223 - Зеркальное отражение по y не применимо

Объяснение:
оказалось невозможным применить требуемую симметрию вследствие:
- внутренних пределов развертыванию подпрограммы (или макропрограммы): развертывание, в свою очередь, применяет подпрограмму (или макропрограмму), для которых трансформация не допускается (в конфигурации базы данных операций);
- того, что для этого же приложения не допускается трансформация (в конфигурации).

224 - Растягивание не применимо

Объяснение:
оказалось невозможным применить требуемое растягивание вследствие:
- внутренних пределов развертыванию подпрограммы (или макропрограммы): развертывание, в свою очередь, применяет подпрограмму (или макропрограмму), для которых трансформация не допускается (в конфигурации базы данных операций);
- того, что для этого же приложения не допускается трансформация (в конфигурации);
- развертывание подпрограммы (или макропрограммы) включает дуги, заданные на плоскостях, отличных от xy, но растягивание требуется только в плоскости xy.

225 - Запрограммированный инструмент: одна или несколько операций были исключены

Объяснение:
при выполнении кода Запрограммированного инструмента (например, STOOL) одна или несколько операций из указанных по имени была исключена из трансформации, т. к. несовместима с этим инструментом.
Этот сигнал обрабатывается как предупреждение, а не как ошибка.

226 - Чрезмерное число встроенных вызовов подпрограммы (максимум: 5)

Объяснение:
развертывание подпрограммы (или макропрограммы) превысило число вложенных приложений (или макропрограммы). Максимальное число вложенных приложений:
- 5: в случае типологии программы;
- 4: в случае типологии подпрограммы и макропрограммы.

Этот сигнал может соответствовать также рекурсивным вызовам программируемых инструментов (коды STOOL). В этом случае вложение вызовов имеет развертку, которую можно назвать *вертикальной*, вдоль текста программы (подпрограммы или макропрограммы).

227 - Клиентская ошибка номер <код клиентской ошибки>

Объяснение:
развертывание подпрограммы (или макропрограммы) распознало клиентскую ошибку по толкованию инструкции запрограммированной ошибки:
- ERROR [код = 2009];
- BREAK [код = 2005], может использоваться только в тексте макропрограммы.

Номер запрограммированной ошибки заменяет <код клиентской ошибки> в сообщении.
Этот сигнал может также обуславливаться инструкцией ERROR, запрограммированной непосредственно в тексте программы. В этом случае этот сигнал генерируется на этапе регистрирования программы и сигнализирует невозможность выполнения программы.

228 - Невозможно задать шрифт (неправильное имя)

Объяснение:
неправилен заданный шрифт. Случаи ошибки:
- не задано имя шрифта;
- имя недействительно вследствие чрезмерной длины (макс.: 32 символа);
- недействительное имя для задания шрифта, управляемого системой Windows ®;
- недействительное имя для задания шрифта TrueType;
- недействительное имя для задания клиентского шрифта (файл не найден);
- клиентский шрифт не задает действительное значение высоты шрифта (минимум: эпсилон * 100).

Контекст:
этот сигнал может подаваться в случае применения подпрограммы (или макропрограммы) создания текста

229 - Невозможно задать устройство для создания шрифта

Объяснение:
возникла системная ошибка при задании устройства отображения для создания шрифта.

Контекст:
этот сигнал может подаваться в случае применения подпрограммы (или макропрограммы) создания текста

12.8 Ошибки логических условий

Это ошибки, связанные с проверкой применимых к программе логических условий. Это всегда серьезные ошибки, не допускающие выполнения программы.

230 - Число закрывающих ELSE или ENDIF больше числа открывающих IF

Объяснение:
Эта ошибка может указывать, что не задано предыдущее условие IF, в соответствии с указанной командой. Или было определено больше ELSE или ENDIF, чем IF. Проверить соответствие между IF, ELSE и ENDIF. Стоит помнить, что открытый IF не должен закрываться ENDIF.

Контекст:
Ошибка, которая может возникнуть, помимо определения IF, также при применении подпрограммы или макропрограммы.

Пример:
Случай неправильного программирования соответствует:

```
IF …
 IF…
 …
 ENDIF
..
ENDIF
…
ENDIF   <- Endif без соответствующего If
```

231 - Число закрывающих ENDIF меньше числа открывающих IF

Объяснение:
было определено больше IF, чем ENDIF. Проверить соответствие между IF, ELSE и ENDIF.

Контекст:
Ошибка, которая может возникнуть, помимо определения IF, также при применении подпрограммы или макропрограммы.

Пример:
Случай неправильного программирования соответствует:

```
IF …   <- If без соответствующего Endif
 IF…
 …
 ENDIF
..
```

232 - Неправильный код после открытого IF

Объяснение:
операция после открывающего IF неправильна. Открывающий IF может определять операцию:
- точечную
- клиентскую логическую
- сложную (подпрограмма или макропрограмма).

Только если в тексте макропрограммы:
- подготовки
- профиля (дуги или линии)
- неклиентская логика присвоения переменных ($, j).

Контекст:
Ошибка, которая может возникнуть, помимо определения IF, также при применении подпрограммы или макропрограммы.

233 - Число закрывающих ENDFOR больше числа открывающих FOR

Объяснение:
было определено больше ENDFOR, чем FOR. Проверить соответствие между FOR и ENDFOR.

Контекст:

Эта ошибка может возникнуть только при программировании макропрограммы.

234 - Число закрывающих ENDFOR меньше числа открывающих FOR

Объяснение:
было определено больше FOR, чем ENDFOR. Проверить соответствие между FOR и ENDFOR.

Контекст:
Эта ошибка может возникнуть только при программировании макропрограммы.

235 - Число команд FOR больше максимально допустимого (максимум: 500)

Объяснение:
было задано более 500 циклов FOR, что представляет собой максимальный предел, управляемый для программы грани.

236 - Число итерраций при выполнении циклов FOR превышает максимально допустимое (максимум: 100000)

Объяснение:
применение макропрограммы определило исполнение более 100000 циклов FOR. Эта проверка подключается для исключения возможности ввода бесконечных циклов, что заблокировало бы приложение.

237 - Команда ENDIF используется для завершения цикла FOR

Объяснение:
проверить соответствие между IF- ELSE и ENDIF, FOR и ENDFOR. Напоминаем два простых общих правила, которые должны соблюдаться при контекстуальном задании циклов IF и FOR:
- если внутри цикла IF: цикл FOR ..ENDFOR должен полностью определяться внутри только одной ветви IF..ELSE..ENDIF;
- если внутри цикла FOR: цикл IF..ELSE..ENDIF должен полностью определяться внутри цикла FOR.

Пример:
Случай неправильного программирования соответствует:
```
FOR …
 IF…
 …
 ENDIF
..
ENDIF   <- Endif, который закрывает FOR
```

238 - Команда ENDFOR используется для завершения цикла IF

Объяснение:
проверить соответствие между IF- ELSE и ENDIF, FOR и ENDFOR. См. предупреждения, приведенные для предыдущей ошибки.

Пример:
Случай неправильного программирования соответствует:
```
IF …
 IF…
 …
 ENDIF
..
ENDFOR  <- Endfor, который закрывает If
```

12.9 Ошибки при задании глобальных функций

239 - Команда ELSEIF используется в цикле IF после ELSE

Объяснение:
В цикле IF была определена инструкция ELSEIF после инструкции ELSE: в цикле IF, при использовании инструкции ELSE, она должна быть последней цикла, до ENDIF.

Контекст:

Ошибка, которая может возникнуть, помимо определения IF, также при применении подпрограммы или макропрограммы.

240 - Имя клиентской функции не задано

Объяснение:
имя функции не задано.

Контекст:
этот сигнал соответствует неправильному заданию операции в базе данных операций.

241 - Недопустимое имя клиентской функции

Объяснение:
имя функции неправильно.

Контекст:
Этот сигнал может соответствовать одной из следующих ситуаций:
- неправильное задание операции в базе данных операций;
- неправильное задание клиентской функции на этапе конфигурации приложения.

242 - Ошибка исполнения клиентской функции: не заданы возвраты

Объяснение:
развертывание глобальной функции неправильно.

Контекст:
Этот сигнал соответствует ситуации ошибки параметрического программирования, выявленной на этапе интерпретации и развертывания функции.
Кроме того, указывает, что не было выполнено какого-либо задания переменных (j).

12.10 Ошибки подготовки мультипрофилей

Эти ошибки сигнализируются только при запросе оптимизации программы или создании матрицы детали. Это серьезные ошибки, не допускающие выполнения программы.

245 - Расчет мультипрофилей превышает максимальное число операций, которые можно задать для грани

Объяснение:
на грани невозможно завершить этап расчета многократных подготовок ввиду достижения максимального числа операций.

12.11 Ошибки технологического задания профилей и точечных операций

Эти ошибки сигнализируются только при запросе оптимизации программы или создании матрицы детали. Это серьезные ошибки, не допускающие выполнения программы.

250 - Невозможно применить подготовку к разомкнутому профилю из-за отсутствия исходного кода

Объяснение:
невозможно завершить данный этап вследствие отсутствия необходимой операции setup. В частности, отсутствует как технологическая, так и геометрическая setup. Технологическая setup задается на уровне конфигурации приложения.

251 - Невозможно применить технологическую точку из-за отсутствия исходного кода

Объяснение:
невозможно завершить данный этап вследствие отсутствия точечной технологической операции, необходимой для замены геометрических точек. Точечная технологическая операция задается на уровне конфигурации приложения.

252 - Невозможно задать открытые профили

Объяснение:
были выявлены профили, открытые или начинающиеся с геометрической подготовки, в то время как конфигурация приложения исключает их исполнение. В этом случае необходимо решить задание профилей, задавая для каждого из них технологическую подготовку.

253 - Была выполнена замена технологии

Объяснение:
сигнализация не является серьезной (это предупреждение) и указывает, что была выполнена одна или несколько технологических замен, обусловленная состоянием износа запрограммированных инструментов.

254 - Не удалось выполнить замену технологии

Объяснение:
сигнализация указывает, что не удалось выполнить технологическую замену, обусловленную состоянием износа запрограммированного инструмента. В этом случае необходимо решить, сделав инструмент действительным с заменой или разрешив найти действительную замену.

12.12 Ошибки задания начального/конечного отрезков профиля

271 – Вход/Выход профиля: невозможно построить трехмерную дугу

Объяснение:
процедура не может решить требуемый тип отрезка т.к. среди сконфигурированных для применения операций отсутствует операция части окружности с пространственной разверткой.
В контексте TpaCAD эта процедура решает линейный отрезок и этот сигнал соответствует "Серьезному предупреждению".
В контексте выполнения: этот сигнал соответствует ошибке.

272 - Вход/Выход профиля: запрограммированная геометрия несовместима с запросом коррекции на инструмент

Объяснение:
процедура не может решать отрезки типа "**Приближение"/"Удаление"/"Перекрытие**", т.к. несовместима с одновременным запросом коррекции на инструмент. Этот сигнал генерируется в случае, когда отрезок не оказывается на непрерывной касательной с начальным/конечным отрезком профиля.
В TpaCAD эта процедура не решает какой-либо входной / выходной отрезок и этот сигнал соответствует "Серьезному предупреждению".
Этот сигнал соответствует ошибке при выполнении программы на машине.

273 – Начало / Конец профиля: невозможно построить отрезок перекрытия, если профиль не замкнут

Объяснение:
процедура не может решить тип конечного отрезка с "**Перекрытием**", т.к. исходный профиль не замкнут. Если исходный профиль начинается линейным отрезком или дугой в плоскости xy, достаточно, чтобы замыкание выполнялось на координатах (x;y). Если исходный профиль начинается дугой в плоскости # xy, замыкание также должно выполняться на координате глубины.
В TpaCAD эта процедура не решает какой-либо входной / выходной отрезок и этот сигнал соответствует "Серьезному предупреждению".
Этот сигнал соответствует ошибке при выполнении программы на машине.

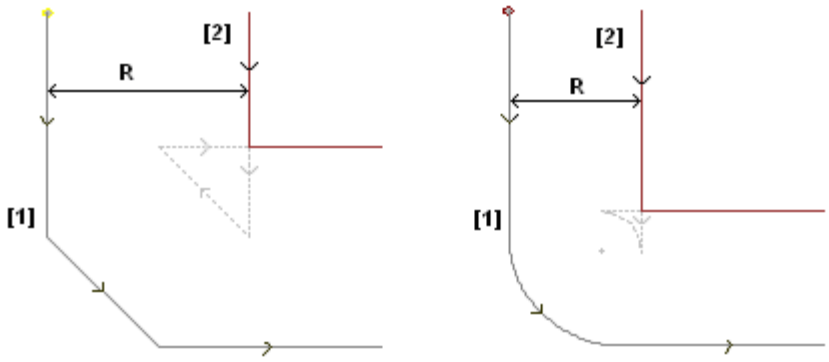
12.13 Ошибки при применении коррекции на инструмент

Это серьезные ошибки, не допускающие выполнения программы.

261 - Коррекция на инструмент: чрезмерная коррекция на радиус дуги

Объяснение:
требуется внутренняя коррекция для дуги со значением коррекции больше радиуса дуги. Если требуется применять коррекцию, ошибку можно устранить путем подключения Сокращения профиля.

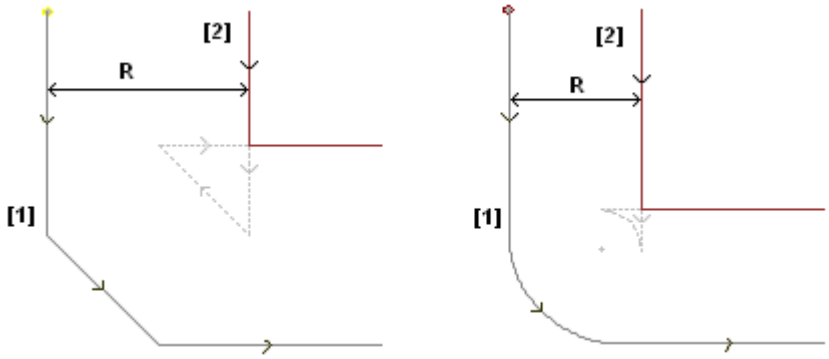
Возвращаясь к тому, что было сказано в пункте, касающемся применения *Коррекции на инструмент*, ситуация соответствует случаю справа на рисунке:



262 - Коррекция на инструмент: чрезмерная коррекция на отрезок

Объяснение:
требуется коррекция линейного отрезка с таким значением коррекции, которое меняет направление его прохождения. Если требуется применять коррекцию, ошибку можно устранить путем подключения Сокращение профиля.

Возвращаясь к тому, что было сказано в пункте, касающемся применения *Коррекции на инструмент*, ситуация соответствует случаю слева на рисунке:



263 - Коррекция на инструмент: применяется сокращение к профилю

Объяснение:
Коррекция на инструмент профиля потребовала применения сокращения.
Отчет управляется как **предупреждение**, а не как ошибка. Цель отчета - подчеркнуть возможную необходимость проверки результатов исправления.

265 - Коррекция на инструмент: ошибка из-за коррекции в плоскости, отличной от xy, с вариантом пересечения отрезков

Объяснение:
требуется коррекция кругового отрезка, заданного в плоскости, отличной от xy, и возникло условие решения пересечения внутри дуги. Эту ошибку можно устранить путем удаления отрезка или прерывания коррекции отрезка.

266 - Коррекция на инструмент: ошибка из-за коррекции в плоскости, отличной от xy

Объяснение:
требуется коррекция для кругового отрезка, заданного в плоскости, отличной от xy, и возникло условие первоначальной дуги, которая инвертирует одну из двух координат x или y при выполнении отрезка. Эту

ошибку можно устранить путем удаления отрезка или прерывания коррекции отрезка или изменения геометрии или амплитуды дуги.

267 - Коррекция на инструмент: инверсия поправки должна устранить проблему пересечения или возобновить отрезок после прерывания

Объяснение:
сигнализирует, что в профиле была запрошена инверсия коррекции на неинвертированном отрезке или не задано возобновление коррекции после прерывания или невозможно решить пересечение скорректированных отрезков. Эту ошибку можно устранить путем вставки отрезка с прерыванием коррекции.

268 - Коррекция на инструмент: была запрошена приостановка коррекции без дальнейшего возобновления

Объяснение:
в профиле была выполнена приостановка коррекции без дальнейшего возобновления Эту ошибку можно устранить путем задания возобновления коррекции в требуемой точке.

269 - Коррекция на инструмент: приостановка и дальнейшее возобновление коррекции не помогает выполнить скругление

Объяснение:
в профиле была выявлена неправильная приостановка и возобновление коррекции: решение коррекции между двумя задействованные отрезками не устраняет пересечение.

270 - Коррекция на инструмент: приостановка и дальнейшее возобновление коррекции должно обеспечить проверку геометрической непрерывности отрезков

Объяснение:
в профиле была выявлена неправильная приостановка и возобновление коррекции: два задействованных отрезка профиля должны иметь геометрическую непрерывность (первый завершается там, где начинается второй).

12.14 Ошибки фрагментации и линеаризации дуг в плоскостях, отличных от xy

Эти ошибки сигнализируются только при запросе оптимизации программы или создании матрицы детали. Это серьезные ошибки, не допускающие выполнения программы.

255 - При линеаризации трехмерных дуг превышено максимальное число линий

Объяснение:
на грани невозможно завершить данный этап ввиду достижения максимального числа операций.

256 - Линеаризация трехмерных дуг невозможна из-за отсутствия исходного линейного кода

Объяснение:
невозможно завершить данный этап вследствие отсутствия операции рабочего кода L01 [код= 2201].

13 Индивидуализация TpaCAD

Для персонализации TpaCAD выбрать пункт **Индивидуализировать** меню Применение.

13.1 Среда

Запуск

При запуске:
выбирает порядок работы TpaCAD при запуске.
- **Создать новую программу:** создает новую программу на основе файла прототипа.
- **Загрузить последнюю программу:** загружает последнюю открытую программу
- **Показать пустое окружение:** не загружено никакой программы

- **Поддерживать каталог в Файле Открыть:** при подключении окно Открыть Деталь предлагает папку и тип файла последнего открытого файла. Если умолчание отключено, окно Открыть Деталь предлагает папку архивирования программ и тип программ-детали.
- **Восстановить фильтры Слоев:** при подключении при запуске TpaCAD восстанавливаются состояния вида и блокировки Слоев на момент последнего закрытия приложения. В противном случае не задано никакого ограничения вида или изменения Слоев. Умолчание - дезактивировано. Эта опция недоступна, если не реализовано свойство Слоя.
- **Восстановить Специальные фильтры** при подключении при запуске TpaCAD восстанавливаются состояния вида или блокировки Специальных фильтров на момент последнего закрытия приложения. В противном случае не задано никакого ограничения вида или изменения Специальных фильтров. Умолчание - дезактивировано. Эта опция недоступна, если не реализовано свойство Специальные фильтры.
- **Восстановить закладки:** при подкл:ючении при запуске TpaCAD восстанавливаются закладки так, как они были заданы при последнем закрытии приложения. Умолчание - дезактивировано. Эта опция недоступна, если не реализован раздел Закладки.
- **Выбрать рабочую среду при запуске:** при подключении при запуске TpaCAD можно выбрать, какую среду запускать: Станок или Чертеж. Эта опция недоступна, если не реализована функция двойной рабочей среды.

Клиентская версия: содержит строку, которая поможет в определении инсталлированной версии клиентской конфигурации. Информация считывается из файла версии, который должен управляться ответственным за индивидуализации и который служит для определения версий базы данных клиентских операций.

Действие

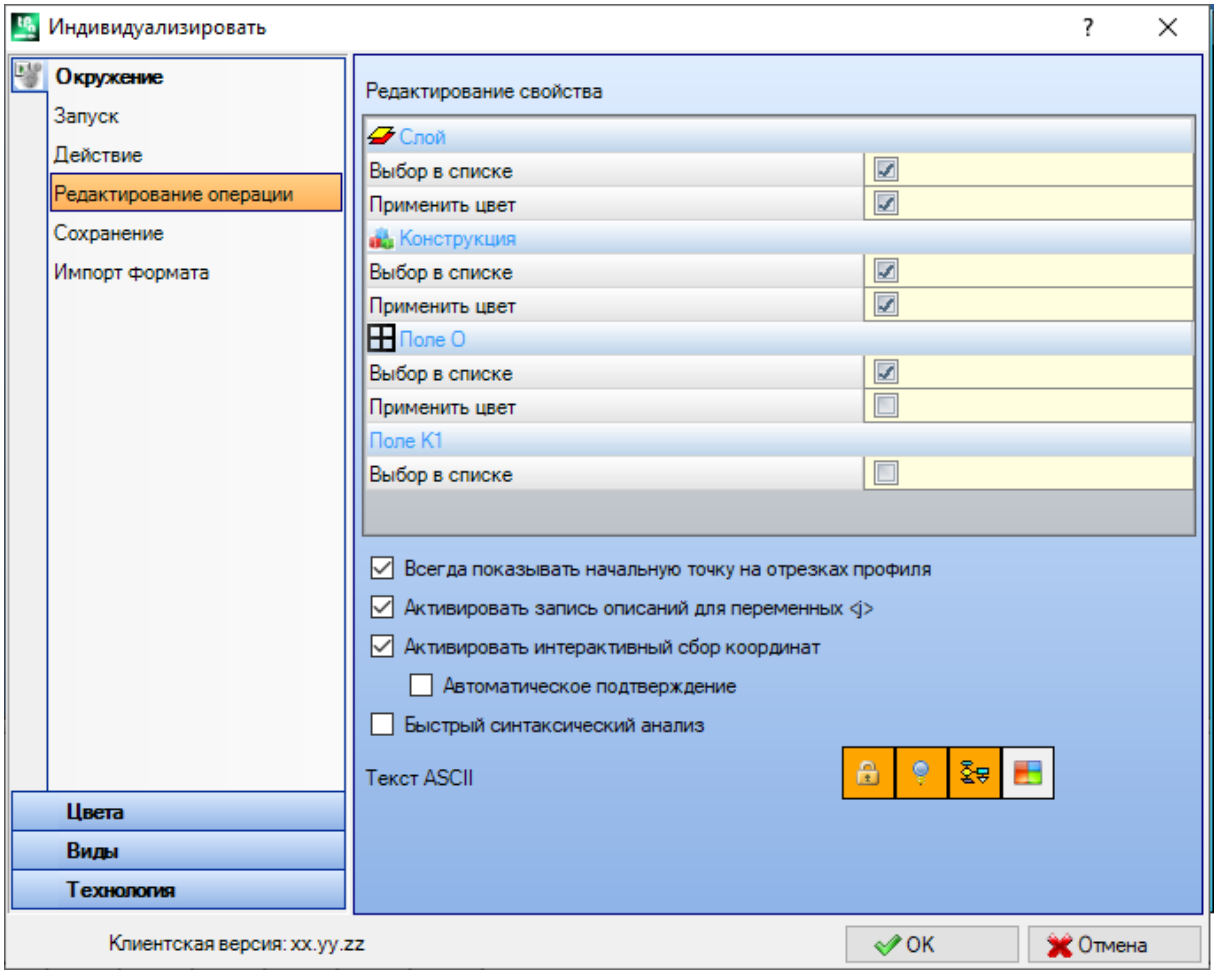
Подтвердить изменение информации о программе
задает порядок управления изменением общей информации о детали (размеры, переменные, клиентский раздел,..), не подтвержденным специально. Имеется 3 опции:
- **Подтверждение по запросу:** система запрашивает, должна ли она принимать изменения
- **Автоматическое подтверждение:** изменения принимаются автоматически
- **Не автоматическое подтверждение:** изменения, не подтвержденные специально, теряются.

Подтвердить изменение операции
задает порядок управления изменением операции в грани программы, не подтвержденным специально. Имеется 3 опции:
- **Подтверждение по запросу:** система запрашивает, должна ли она принимать изменения
- **Автоматическое подтверждение:** изменения принимаются автоматически
- **Не автоматическое подтверждение:** изменения, не подтвержденные специально, теряются.

Форматы Edicad/TpaEdi32: уставки из файла прототипа
задает порядок управления заданиями из файла прототипа при открытии программы в формате Edicad или tpaEdi32. Имеется 3 опции:
- **Подтвердить применение**: система запрашивает, должна ли она принимать задания
- **Применять по умолчанию**: задания считываются автоматически
- **Не применять по умолчанию**: задания не выполняются.

- **Запрашивать подтверждение при изменении операций командой редактирования**: при подключении каждый раз, когда пользователь выбирает команду Удаление, Вставить, Перемещение, Дублирование операции, отображается запрос подтвердить эту команду.
- **Подтвердить изменение операции перехода к следующей строке**: при подключении четкое подтверждение изменения операции (т. е. подтверждение выбором кнопки задания операции) перемещает текущую операцию на следующую строку в списке.
- **Допустить подтверждение операции, составленной с ошибкой**: при подключении можно подтвердить ввод или прямое изменение операции также в случае, если при операции был подан сигнал об ошибке. Исключение составляют ситуации серьезных неисправимых ошибок: крайней ситуацией является отсутствие системной памяти.
- **Разрешить автоматическое включение операций**: при подключении ввод определенной операции завершается автоматически без необходимости подтверждения. Этот выбор касается ввода операции, выбранной из графической палитры или из списка предпочтительных операций. Разрешение на прямой ввод - это особая характеристика каждой отдельной операции.
- **Добавить выборы**: при подключении выборы в списке грани не обнуляются при выборе операции или при прямом наведении в графической зоне (щелчок в зоне) или в тексте ASCII.
- **Палитра операций: автоматическое расширение групп**: при подключении кнопки (группы) палитры операций открываются, приводя мышь на зону этой кнопки. В противном случае для открытия группы необходимо щелкнуть на ней.

Редактирование операции



Редактирование свойства

- **Выбор в списке:** выбрать пункт для управления выбором в соответствующем списке свойств, в противном случае управляется клетка редактирования. Выбор возможен только если свойство задало максимальное значение не выше 16.
- **Применить цвет:** выбрать этот пункт для применения цвета, задаваемого соответствующему свойству, в графике операций.
  Выборы по умолчанию:

- поле "L": выбор в списке, применяет цвет;
- поле "B" (конструкция): выбор в списке, применяет цвет;
- поле "O": выбор в списке, не применяет цвет;
- поле "K": выбор не в списке, не применяет цвет (недоступен);
- поле "K1": выбор не в списке, не применяет цвет (недоступен);
- поле "K2": выбор не в списке, не применяет цвет (недоступен).

При подключенном выборе в списке образующие список пункты могут задаваться:
- на этапе индивидуализации (в таблицах цветов) для полей "L" и "B";
- на этапе конфигурации для полей "O", "K", "K1" и "K2". В этом случае сообщения переводятся на другой язык.

При графическом представлении операции с несколькими запросами выбора Применить цвет применяются следующие критерии: поле "L" превалирует над полями "B" и "O"; Поле "B" превалирует над полем "O".

- **Исходное растровое изображение для поля O:** опция доступна только если Поле O может принимать значение максимум 1. Можно сделать выбор из двух опций, с отображением значения, присвоенного значимым значениям для поля (0/1):
  - Левая сторона / Правая сторона
  - Нижняя сторона / Верхняя сторона

  ВНИМАНИЕ: управление этим пунктом, в любом случае, зависит от подключения определенных параметров в конфигурации TpaCAD.
- **Всегда показывать начальную точку на отрезках профиля:** этот пункт касается отображения параметров, соответствующих координатам начальной точки отрезка профиля (дуги или линии). Выбрать этот пункт для постоянного отображения и возможности управления этими полями; оставить этот пункт не выбранным для сохранения координат невидимыми, но только если они не заданы. Если этот пункт не выбран, поля, тем не менее, отображаются, даже если они не заданы, если операция (дуга или линия) фактически начинает профиль: она первой запрограммирована в списке грани или предшествующий операцией, которая не относится к профилю.
  Статус поля также можно изменить, используя кнопку, присутствующую в командной строке текущего задания, с применением тех же критериев.
  Этот режим оставляет без изменений функции интерактивной вставки геометрических элементов, которые напрямую считывают начальную точку применения, в то время как можно моделировать программирование только конечной точки отрезка профиля, вставленного при прямом выборе из палитры операций. Умолчание активировано.
- **Активировать запись описаний для переменных <j>:** выбрать этот пункт, чтобы разрешить редактирование полей, соответствующих описательным сообщениям переменных <j>, при вводе инструкций задания переменных. Если этот пункт не выбран, поля остаются видимыми, но их нельзя редактировать: все уже выполненные настройки будут отображены. Умолчание активировано.
- **Активировать интерактивный сбор координат:** выбрать этот пункт, чтобы разрешить интерактивный сбор координат из ввода данных операции в соответствии с конфигурацией базы данных операций (поля со значком ). Умолчание активировано.
  - **Автоматическое подтверждение:** выбрать этот пункт для автоматического подтверждения изменения операции, после интерактивного считывания координат из ввода данных операции. Умолчание - дезактивировано. На этапе закрытия интерактивного считывания можно отказаться от выбранного здесь условия, удерживая нажатой клавишу Shift.

ВНИМАНИЕ: управление этими пунктами, в любом случае, зависит от подключения определенных параметров в конфигурации TpaCAD.

- **Быстрый синтаксический анализ:** выбрать этот пункт для выполнения контроля действительности параметрического задания при каждом изменении параметра операции. При неправильном задании немедленно сигнализируется ошибка. Умолчание - дезактивировано.
- **Текст ASCII:** эта группа графического выбора задает состояние подключения вспомогательных столбцов в таблице текста ASCII, соответственно, для:
  - **Состояние изменения**: этот столбец указывает состояние изменения операции
  - **Состояние отображения**: этот столбец указывает, соответствует ли операции отображение в зоне графического представления
  - **Логическое состояние**: этот столбец указывает логическое состояние операции, если подключена опция *Вид логических условий*
  - **Цвет**: этот столбец указывает основной присвоенный операции цвет в зависимости от типа операции (точечная, подготовка, отрезок профиля) или рабочего кода. Задание цвета в зависимости от типа выполняется в группе последующих заданий или в базе данных операций. Если код сложный (вызов подпрограммы или код макропрограммы) и не задан персонализированный цвет в базе данных операций и если развертка соответствует профилю, в программной строке задается цвет, заданный для элементов профиля.
    Это подключение применяется также в расширенном списке операций и в окне "Последовательности".

Сохранение

- **Сохранить программы в формате ASCII:** выбрать этот пункт для сохранения программ в формате ASCII. Под форматом ASCII здесь подразумевается не тип регистрируемого файла, который, в любом случае, остается текстовым файлом типа ASCII, а кодировка, с которой регистрируется информация программы и, в основном, - операции. Формат ASCII обеспечивает более интуитивное прочитывание и может использоваться также для создания программ для TpaCAD. Ниже приводится строка, соответствующая регистрации операции сверления (рабочий код: 81; Имя ASCII: HOLE), соответственно, для формата ASCII и для внутреннего формата:
  - HOLE WS1 EG0 X100 Y100 Z-12 TD10 TMC1 TR1 TP1
  - W#81{::WTp WS=1 #8015=0 #1=100 #2=100 #3=-12 #1002=10 #201=1 #203=1 #1001=1 }W.

  При считывании программы в формате ASCII может наблюдаться удаление исходных строк, в случае, если не будет найдено соответствия между загруженными операциями и операциями, определенными в базе данных операций. Умолчание - дезактивировано.
  Выбор этой опции не применяется к сохранению макропрограммы.
- **Оптимизировать программы:** при активизации после записи программы в память выполняется ее оптимизация. Умолчание - дезактивировано. При подключении:
  - если не сконфигурировано приложение оптимизации, проверка ограничивается выполнением комплексного анализа программы с применением: логических условий, технологических заданий по умолчанию, коррекции на инструмент). Этот этап анализа может завершаться подачей диагностических сигналов;
  - если приложение оптимизации сконфигурировано, оно выполняет проверку детали для выполнения (с возможным применением оптимизации маршрута и/или отсчетов и/или смен инструментов). В любом случае, перед оптимизацией программы выполняется комплексный анализ в соответствии с предыдущим пунктом.
- **Зарегистрировать матрицу детали:** при подключении, оптимизация программы на диске записывает файл, соответствующий оптимизированной версии (которая иначе называется "матрица детали", с расширением файла .TXN и/или .TXM). Этот пункт отображается только если сконфигурировано приложение оптимизации, а выбор применяется только если выбрано: **Оптимизировать программы**. Умолчание - дезактивировано.
- **Запрашивать подтверждение оптимизации для большой программы:** при подключении, после записи программы в память, запрашивает подтверждение для выполнения оптимизации или экспорта формата, если размер программы больше заданного значения. Должно задаваться значение в Кб в диапазоне от 1 до 50000. Умолчание - дезактивировано.
  ВНИМАНИЕ! Эта настройка используется также на этапе открытия программ:
  1. для управления просмотром графического представления.
  При выборе программы размерами, превышающими заданные здесь, предварительный просмотр автоматически временно отключается. Для запроса просмотра графического представления необходимо подключить пункт **Предварительный просмотр** в окне Открыть Файл.
  2. для управления графикой программы. (См.: **Индивидуализация TpaCAD-> Виды -> Индивидуализировать графическое представление**, настройка **Отключить графические элементы, добавленные для большой программы**).

- **Автоматическое сохранение:** при подключении для программы автоматически создается копия автоматического сохранения в указанном временном интервале. Должно задаваться значение в минутах в диапазоне от 1 до 60. Автоматическое сохранение выполняется в папке временных файлов TpaCAD во временном файле с фиксированным названием и расширением TBK. Умолчание - дезактивировано.
  В условиях нормального функционирования файл автоматического сохранения стирается при закрытии TpaCAD.
  Если этого не происходит, например, вследствие ошибки, вызывающей непредвиденное закрытие TpaCAD, или при отключении подачи электропитания в ходе изменения программы, можно будет восстановить последнее выполненное сохранение. При запуске TpaCAD сообщение сигнализирует, что была найдена копия файла автоматического сохранения и что можно восстановить его содержание: файл копируется в резервный невременный файл (в той же папке и с таким же названием, но с расширением SBK), позволяя затем открыть его.
  ВНИМАНИЕ! Автоматическое сохранение может точно не соблюдать указанный временной интервал, если выполняется интерактивная процедура или любая команда. В этом случае сохранение выполняется как только это становится возможным, по завершении выполняемой функции.
  ВНИМАНИЕ! Автоматическое сохранение не заменяет команду **Сохранить**. Все также необходимо сохранять программу по завершении работы.
  ВНИМАНИЕ! Автоматическое сохранение не может использоваться для восстановления программы, закрытой без сохранения.
  Конфигурировать модули экспорта
  - отображаются модули экспорта, определенные на этапе конфигурации производителем станка, для которых можно выбрать конфигурацию на этом уровне. При щелчке на значке  можно определить критерии и параметры, которые будут использоваться на этапе экспорта программы. Если не может конфигурироваться ни один модуль экспорта, эта зона не отображается.

Импорт формата

Конфигурировать модули импорта

- отображаются модули импорта, определенные на этапе конфигурации производителем станка, для которых можно выбрать конфигурацию на этом уровне. При щелчке на значке ... можно определить критерии и параметры, которые будут использоваться на этапе импорта программы. Если не может конфигурироваться ни один модуль импорта, эта зона не отображается.

Общие активации программы

Таблица присваивает функции автоматических заданий, выполняемых при открытии программы с подключенным импортом формата. Большая часть приведенных функций имеет полное соответствие в уже рассмотренных *Общих инструментах программы*:

- **Открыть прототип:** при подключении определяет инициализацию программы на основе файла прототипа по умолчанию (файл PIECE.TCN, архивированный в папке TPACADCFG\CUSTOM). Если этот файл существует, программа инициализируется, считывая из файла прототипа режимы выполнения, переменные 'o' и 'v', реализованные клиентские разделы (Специальные уставки, Дополнительная информация, Привязки, Настройки оптимизации) и имена граней. При выборе значка ... открывается окно, в котором приводится полный маршрут файла, из которого инициализирована импортированная программа.
  При управлении гранью-деталью может предлагаться подключаемая опция:
  Грань-деталь: Всегда обновлять из файла прототипа: выбрать это поле для запроса также задания грани-детали из файла прототипа.

ВНИМАНИЕ! Инициализации из файла прототипа, в любом случае, выполняются с соблюдением всех блокировок раздела, если таковые заданы в файле, генерированном процедурой импорта.

- **Применить технологию к профилям:** при подключении присваивает код операции setup, открывающей профили, импортированные как открытые (т. е. без подготовки) или начинающиеся с чисто геометрического кода подготовки. При выборе значка ... можно задать операцию подготовки и технологические задания как требуется. Среди заданий можно задать также цифровые свойства (уровень, поле M,…) за исключением поля B (конструкция). Никакого задания не выполняется, если это поле не подключено
- **Применить технологию к точечным операциям:** при подключении присваивает код точечных операций, которые импортируются с чисто геометрическим кодом операции. При выборе значка ... можно задать операцию setup и технологические задания как требуется. Среди заданий можно задать также цифровые свойства (слой, поле M,…) за исключением поля B (конструкция). Никакого задания не выполняется, если это поле не подключено.
- **Сократить профили:** при подключении уменьшает число последовательных отрезков профилей, в соответствии с правилами, определенными в окне, которое открывается при выборе значка ....
- **Фрагментировать профили:** при подключении выполняет фрагментацию отрезков профиля по правилам, определенным в окне, которое открывается при выборе значка ....
- **Соединить профили:** при подключении активизирует процедуру соединения профилей на основе геометрической непрерывности этих профилей, в соответствии с правилами, определенными в окне, которое открывается при выборе значка ... .
- **Подтвердить профили:** при подключении активизирует процедуру подтверждения профилей на основе программирования подготовки в ненейтральной точке в соответствии с правилами, определенными в окне, которое открывается при выборе значка ... .
- **Различные процедуры:** при подключении подключает процедуры, как они определены в окне, которое открывается при выборе значка . В частности:
- **инвертировать ось глубины:** при подключении инвертирует знак программирования оси глубины грани.

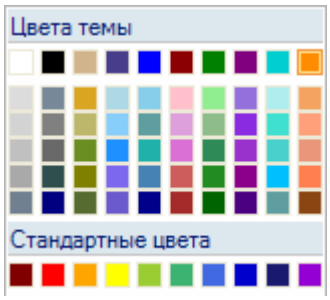
Подтвердить применение: при открытии программы подключает или отключает запрос подтверждения перед применением инструментов, подключенных в предыдущей таблице.

13.2 Цвета

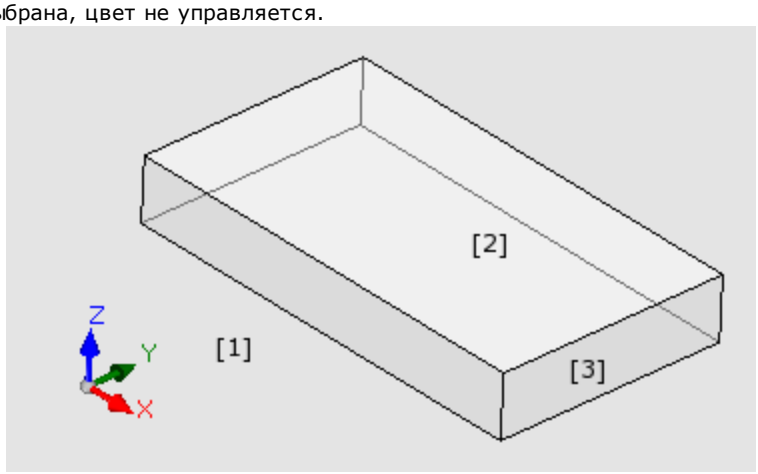
Графическое представление

Цвета графики

При перемещении выбора на список обновляется нижний контрольный индикатор, указывающий надпись и цвет, соответствующий сделанному выбору. Для задания цвета щелкнуть значок



Для каждого пункта приводится одна клетка выбора, состояние которой может меняться только для некоторых пунктов (см. ниже):
- если клетка выбрана, соответствующий цвет применяется обычным образом;
- если клетка не выбрана, цвет не управляется.



- **Фон окна:** цвет фона графической зоны (рисунок: цвет **1**).
- **Панель:** цвет фона панели (рисунок: цвет **3**). Клетка выбора может быть отключена: в этом случае не выполняется заполнение панели заданным цветом. **ВНИМАНИЕ**: для получения эффекта прозрачности панели относительно фона окна необходимо отключить также заполнение графическим рисунком (см. ниже);
- **Активированная грань:** цвет фона текущей грани (рисунок: цвет **2**). Клетка выбора может быть отключена: в этом случае не выполняется заполнение активной грани заданным цветом. **ВНИМАНИЕ**: для получения эффекта прозрачности грани относительно панели, а также фона окна необходимо отключить также заполнение графическим рисунком (см. ниже);
- **Выбор:** цвет представления выбранных операций
- **Текущая операция:** цвет представления текущей операции. Клетка выбора может быть отключена: в этом случае текущая операция представлена своим цветом, определенным в программировании (свойством слоя, конструкции или поля "O") или цветом, определяющим тип операции.

- **Точечная операция:** цвет представления точечных операций. Цвет используется, если для операции не признается одна из трех значимых ситуаций (с приоритетом, определенным порядком в списке):
  - выбранная или текущая операция;
  - выполняется в воздухе или сквозная;
  - имеет заданный цвет свойств (слой, поле B, поле O);
  - имеет заданный цвет представления, персонализированный в базе данных операций.
- **Операция setup:** цвет представления операций подготовки. Цвет используется, если для операции не признается одна из трех значимых ситуаций, как указано для точечных операций
- **Операция профиля:** цвет представления отрезков профиля (линейных сегментов, дуг). Цвет используется, если для операции не признается одна из трех значимых ситуаций, как указано для точечных операций.

- **Исправленный профиль:** цвет представления профилей с примененной коррекцией на инструмент

- **Операция в воздухе:** цвет представления операций, выполняемых в воздухе: оценка воздуха выполняется по оси глубины (координата Z) операций. В случае дуги или линейного сегмента или, полностью или частично выполненного в воздухе, соответствующая часть представляется этим цветом. Клетка выбора может быть отключена: в этом случае запрограммированная в воздухе операция представляется так, как будто она запрограммирована в детали.
  Этот пункт может быть недоступен в зависимости от конфигурации TpaCAD.
  Примеры операций, выполняемых в воздухе:
- сверление или подготовка на координате Z=10,
- линия от Z=10 до Z=0
- линия от Z=10 до Z=-10 (частично выполняется в воздухе)
- **Сквозная операция:** цвет представления операций, выполняемых за пределами толщины грани (сквозная операция). В случае дуги или линейного сегмента, полностью или частично выполненного сквозным, соответствующая часть представляется этим цветом. Клетка выбора может быть отключена: в этом случае сквозная операция представляется так, как будто она запрограммирована в детали.

Этот пункт может быть недоступен в зависимости от конфигурации TpaCAD.
Примеры операций, выполняемых сквозными (с толщиной детали 18 мм):

- сверление или подготовка на координате Z=-25,
- линия от Z=10 до Z=-22 (от Z=10 до Z=0 выполняется в воздухе; от Z=-18 до Z=-22 выполняется сквозной) цвет графического представления операций, выполненных за пределами толщины грани (сквозная операция): если дуга или линейный сегмент полностью или частично выполняется сквозным, соответствующая часть отображается этим цветом. **ВНИМАНИЕ**: Распознавание сквозной операции превалирует над возможным цветом свойств (слой, поле B, поле O).

- **Габаритные отрезки в трехмерном графическом представлении:** цвет представления габарита операций в трехмерном изображении. Клетка выбора может быть отключена: в этом случае габарит имеет тот же цвет, который задан для операции.
- **Чертеж и вспомогательные графические средства:** цвет, используемый в:
  - чертежных функциях
  - инструментах взаимодействия с мышью (например, точка переноса, ось симметрии)
  - управлении расширенным списком в тексте ASCII программы
- **Фон исходной грани:** цвет фона исходной грани в окне задания фиктивной или автоматической грани.
- **Закладки:** цвет представления графических элементов закладки.
- **Графическое заполнение:** цвет заполнения замкнутых графических элементов, в соответствии с продвинутым программированием макропрограммы. Клетка выбора может быть отключена: в этом случае, заполнение выполняется с использованием собственного цвета конструкции (поле "B").

 Выбрать для задания цветов графики на набор цветов по умолчанию.

Слой

Эта вкладка недоступна, если это свойство не реализовано.

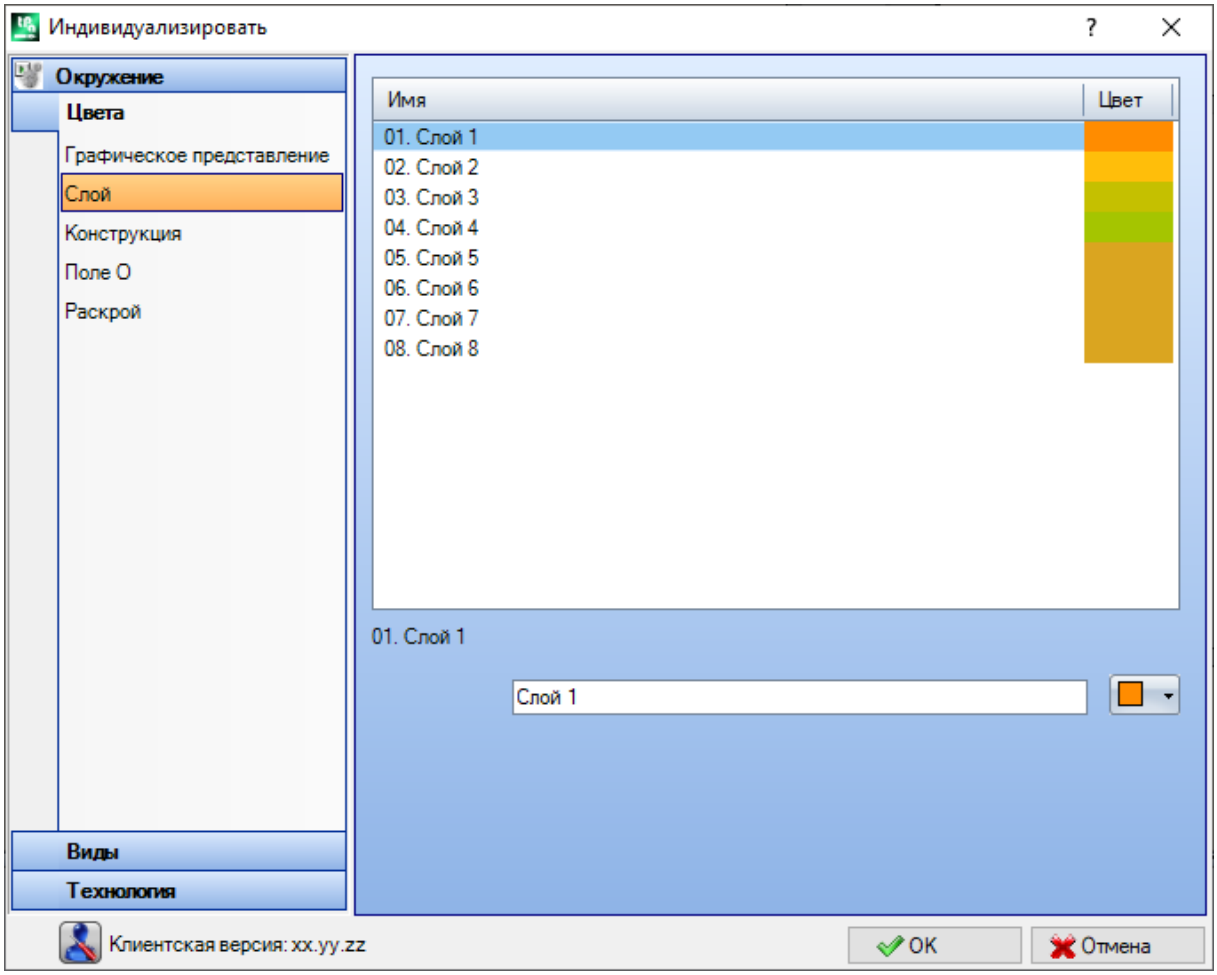


Таблица имеет определенное число строк (8 на рисунке), равное максимальному задаваемому слоям значению при максимальном числе 16. Значения свойств свыше 16 применяют цвет, заданный здесь для значения 16.

- **Заголовок:** номер слоя по порядку начиная со значения 1.
- **Имя:** имя для присвоения слою. Если не задается, присваивается имя по умолчанию: в этом случае имя переводится на текущий язык. Если же имя меняется, оно продолжает предлагаться без изменений перевода на другой язык.
- **Цвет:** для задания цвета слоя щелкнуть ячейку  .

Все операции, имеющие поле "L" отличное от 0, могут отображаться цветом, присвоенным здесь значению слоя. Это выполняется в зависимости от использования умолчания слоя, дополнительно в графическом представлении.
Можно также задавать значение слоя, которое исключает графику операции (см. ниже).

Конструкция

Эта вкладка недоступна, если это свойство не реализовано.
Таблица имеет определенное число строк, равное максимальному задаваемому значению для конструкций при максимальном числе 16. Значения свойств свыше 16 применяют цвет, заданный здесь для значения 16.

- **Заголовок:** номер конструкции по порядку начиная со значения 1.
- **Имя:** имя для присвоения конструкции. Если не задается, присваивается имя по умолчанию: в этом случае имя переводится на текущий язык. Если же имя меняется, оно продолжает предлагаться без изменений перевода на другой язык.
- **Цвет:** для задания цвета конструкции щелкнуть ячейку  .

Все операции, имеющие поле "B" отличное от 0, могут отображаться цветом, присвоенным здесь значению конструкции.

Можно также задавать значение конструкции, которое исключает графику операции (см. ниже).

Поле O

Эта вкладка недоступна, если это свойство не реализовано.
Таблица имеет определенное число строк, равное максимальному задаваемому значению для свойства при максимальном числе 16. Значения свойств свыше 16 применяют цвет, заданный здесь для значения 16.

- **Заголовок:** номер свойства по порядку начиная со значения 1.
- **Имя:** имя для присвоения свойству. Это поле не может быть изменено.
- **Цвет:** щелкнуть ячейку  для задания цвета свойства.

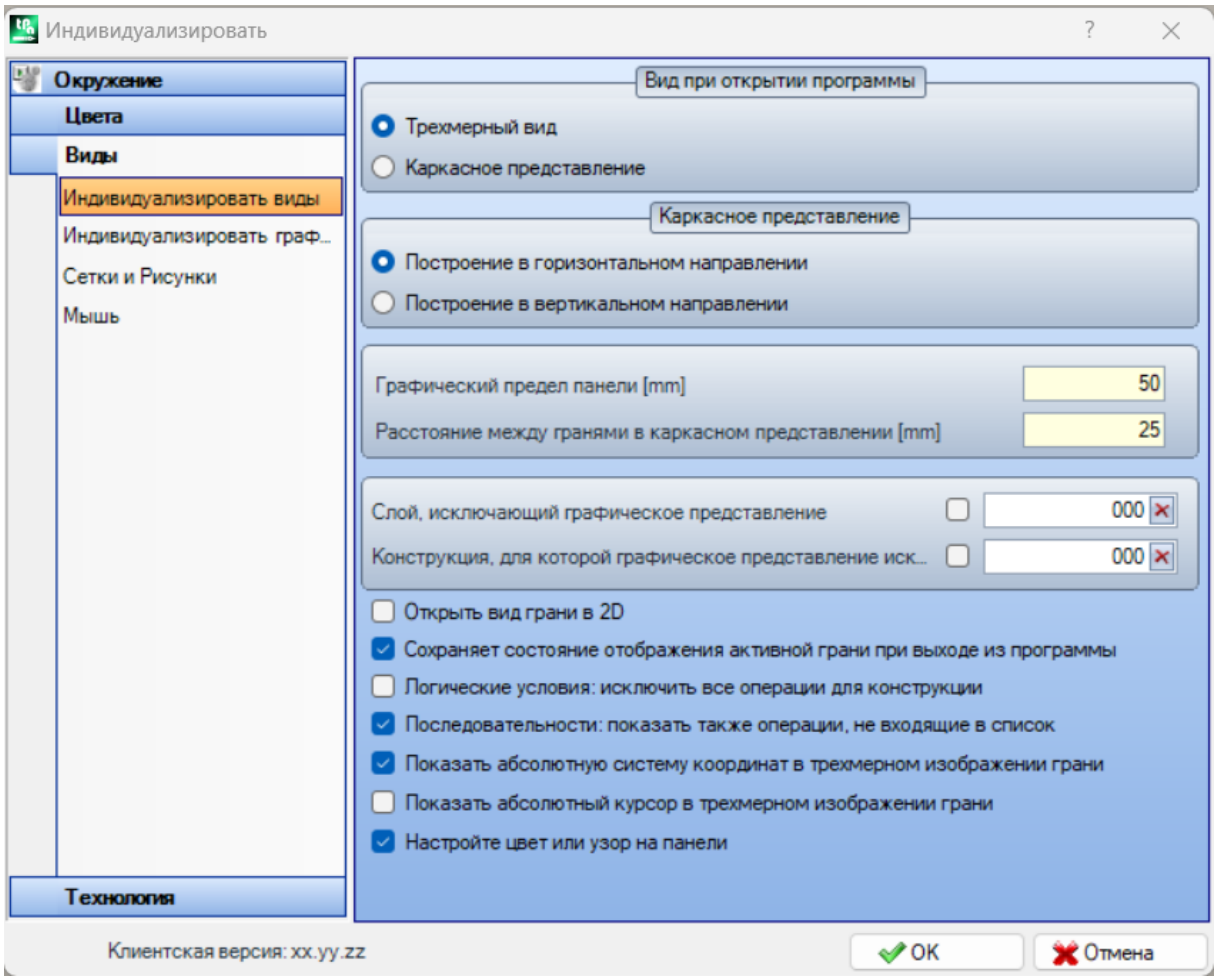
Имена, присвоенные значениям поля O, используются при выборе в списке значения свойства.

Раскрой

Эта вкладка недоступна, если функциональность *Раскроя* не управляется.
Размер таблицы определяется числом строк, равных 25, помеченных от « ID1» до « ID25 » , в соответствии с первыми 25 деталями, которые можно назначить в проекте Раскроя. В случае назначения последующих строк эти же цвета используются с модификацией для разных оттенков.

13.3 Виды

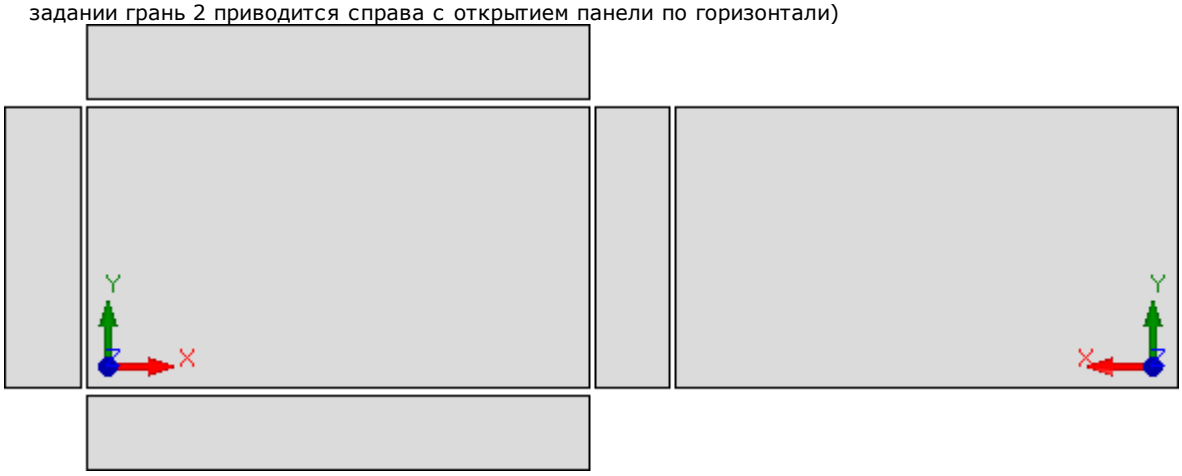
Индивидуализировать виды



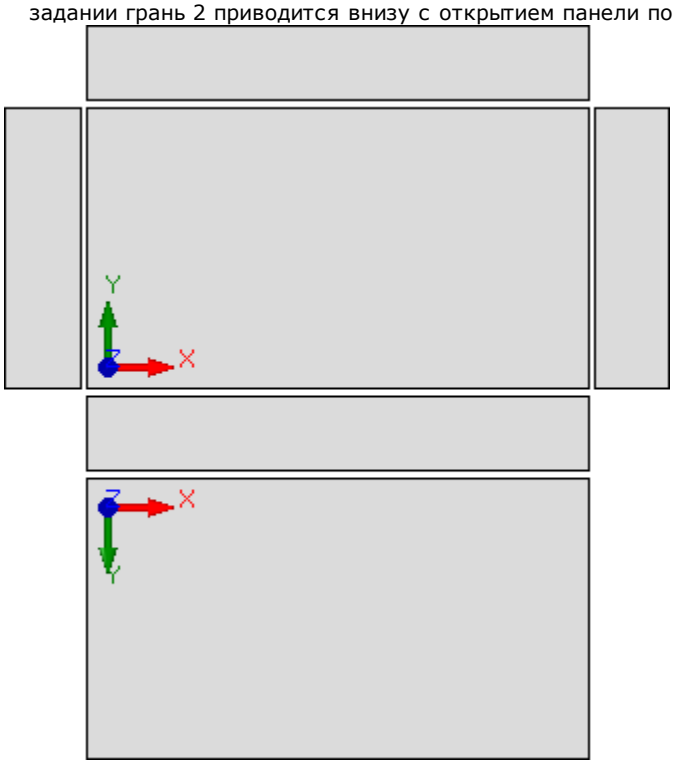
Вид при открытии программы
выбирает режим представления, заданный при открытии программы. Предусматривается два пункта:
- **Трехмерный вид**: панель отображается в трехмерном виде.
- **Каркасное представление**: панель отображается с разбивкой (с выбором следующего пункта).

Выбор может быть недоступен в зависимости от конфигурации TpaCAD.

Каркасное представление
выбирает режим представления, заданный при отображении панели с разбивкой. Предусматривается два пункта:
- **Построение в горизонтальном направлении:** панель разбивается с разверткой по горизонтали (при задании грань 2 приводится справа с открытием панели по горизонтали)



- **Построение в вертикальном направлении:** панель разбивается с разверткой по вертикали (при задании грань 2 приводится внизу с открытием панели по вертикали)



Этот выбор не имеет влияния в следующих случаях:
- грань 2 не задана
- грань 1 не задана. В этом случае грань 2 представляется вместо грани 1.

Боковые грани представлены, даже если частично они не заданы.
Выбор может быть недоступен в зависимости от конфигурации TpaCAD.

- **Графический предел панели:** свободное пространство, оставленное вокруг графического представления детали. Задается в: [мм] или [дюймах] (единицы измерения конфигураций). Это поле принимает значения от 0 мм до 100 мм.

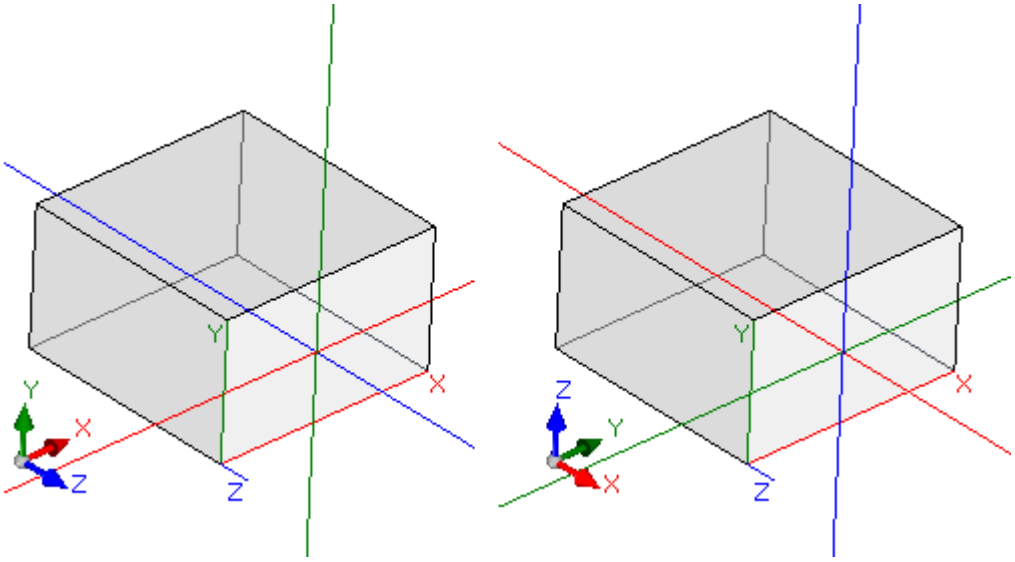
- **Расстояние между гранями в каркасном представлении**: свободное пространство, оставленное между гранями в случае каркасного представления детали. Задается в: [мм] или [дюймах] (единицы измерения конфигураций). Это поле принимает значения от 0 мм до 100 мм. Выбор может быть недоступен в зависимости от конфигурации TpaCAD.
- **Слой, исключающий графическое представление**: значение слоя (поле "L"), исключающее графическое представление операции. Этот параметр может быть полезен, например, в случаях, в которых графика операции полностью поручена использованию конструкций. Принимаются значения от 0 (в этом случае не действует никогда) до максимально используемого значения. Чтобы применить графическое исключение, нужно поставить флажок. Для стирания заданного значения щелкнуть на значке . Эта опция недоступна, если не реализовано свойство Слоя.
- **Конструкция, для которой графическое представление исключено**: значение конструкции (поле "B"), исключающее графическое представление операции. Этот параметр может быть полезен в случаях, в которых операция конструкции используется для создания других операций, например, при использовании после операций, применяющих геометрические трансформации (коды типа STOOL). Принимаются значения от 0 (в этом случае не действует никогда) до максимально используемого значения. Чтобы применить графическое исключение, нужно поставить флажок. Для стирания заданного значения щелкнуть на значке . Эта опция недоступна, если не реализовано свойство Конструкция.
- **Открыть вид грани в 2D**: выбрать для задания вида на плоскости XY при открытии вида грани. Если этот пункт не выбран, режим представления детали остается без изменений при открытии вида грани. Умолчание - не активировано. Графическое представление остается без изменений при открытии вида грани-детали даже при выбранном этом пункте.

- **Сохраняет состояние отображения активной грани при выходе из программы**: подключает запомнить состояние отображения активной грани в момент закрытия TpaCAD. При следующем запуске TpaCAD сохраненная грань, если она существует, будет активирована, и к ней будут применены сохраненные состояния отображения и режим отображения грани (Трехмерный вид, двумерный вид или каркасное представление).

- **Логические условия: исключить все операции для конструкции**: подключает возможность исключения из вида *Логические условия* всех операций конструкции. Если этот пункт не выбран, в виде Логические ограничения операции конструкции (с чисто положительным значением поля "B") оцениваются на основе прямых логических ограничений (IF.. ENDIF, EXIT). Умолчание - не активировано.
- **Последовательности: показать также операции, не входящие в список:** подключает или отключает полное отображение в виде последовательностей. Если этот пункт не выбран, на виде последовательностей отображаются только операции, для которых можно задавать последовательность; поэтому не отображаются открытые профили и операции, для которых напрямую отключено управление последовательностью. Умолчание - не активировано.

- **Показать абсолютную систему координат в трехмерном изображении грани:** подключает или отключает отображение абсолютной системы координат в трехмерном изображении грани. Если не активизируется, представленная система трех координат отражает ориентацию грани. Умолчание - не активировано.
- **Показать абсолютный курсор в трехмерном изображении грани:** подключает или отключает отображение абсолютного курсора в трехмерном изображении грани. Если не активизируется, представленный курсор - курсор, отражающий ориентацию грани. Умолчание - не активировано.

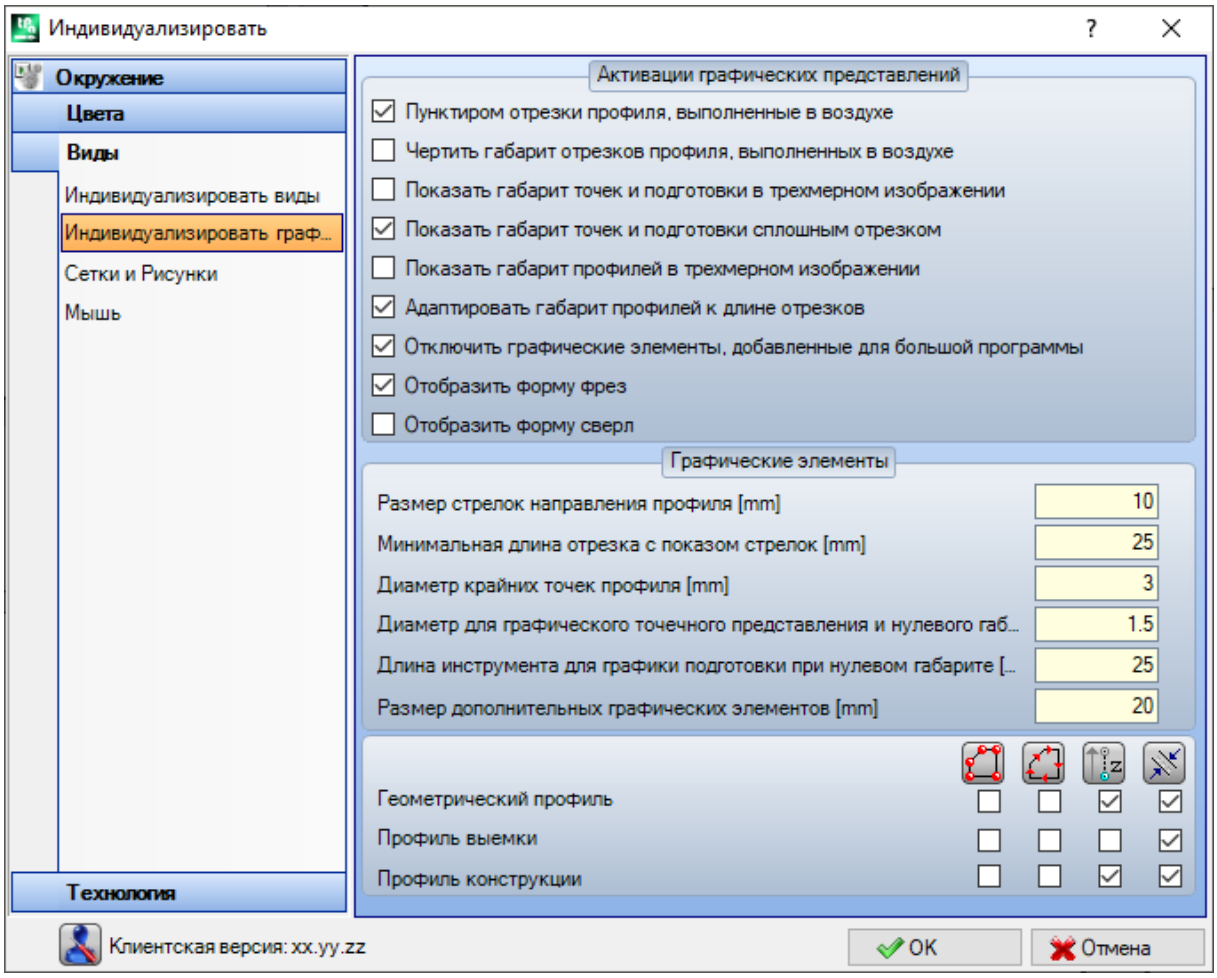
При активном виде грани 4 посмотрим, как меняются представления системы координат и курсора:



[слева] отключены оба пункта: отображается локальная система координат и курсор грани
[справа] подключены оба пункта: отображается абсолютные система координат и курсор на детали.

- **Настройте цвет или узор на панели:** подключает или отключает применение к текущей программе цвета или рисунка, если они заданы в *Специальных уставках* программы. Умолчание - не подключено и пункт может быть невидимым для задания.

Индивидуализировать графическое представление



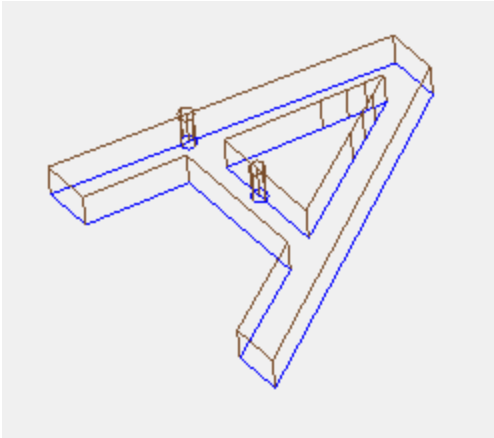
Активации графических представлений

- **Пунктиром отрезки профиля в воздухе:** при выборе активизирует отображение пунктиром отрезков профиля, выполненных в воздухе. Умолчание активировано. Выбор может быть неизменяемым в зависимости от конфигурации TpaCAD.
- **Чертить габарит отрезков профиля, выполненных в воздухе:** выбрать для подключения отображения габарита инструмента также для отрезков профиля, выполненных в воздухе. Напоминаем, что габарит инструмента на отрезках профиля приводится на виде *Коррекция на инструмент*. Умолчание - дезактивировано. Выбор может быть неизменяемым в зависимости от конфигурации TpaCAD.
- **Показать габарит точек и подготовки в трехмерном изображении:** подключает или отключает отображение габаритов в трехмерном представлении точечных и подготовительных операций. Габарит выделяет использование инструмента на детали с оценкой по запрограммированной глубине с фактическим входом в деталь.



На рисунке представлен пример трехмерного изображения точечных (или изолированных подготовительных) операций с подключенным отображением габаритов. Для каждой операции представлен один цилиндр высотой, равной запрограммированной глубине. Заполнение цилиндра определяется выбором **Показать габарит точек и подготовки сплошным отрезком**. Умолчание - не активировано.
Применение выбора зависит от состояния пункта **Габариты трехмерного графического изображения** в меню Показать и локального меню ассоциированное с самым пунктом: если этот пункт не выбран, опция **Показать габарит точек и подготовки в трехмерном изображении** игнорируется.

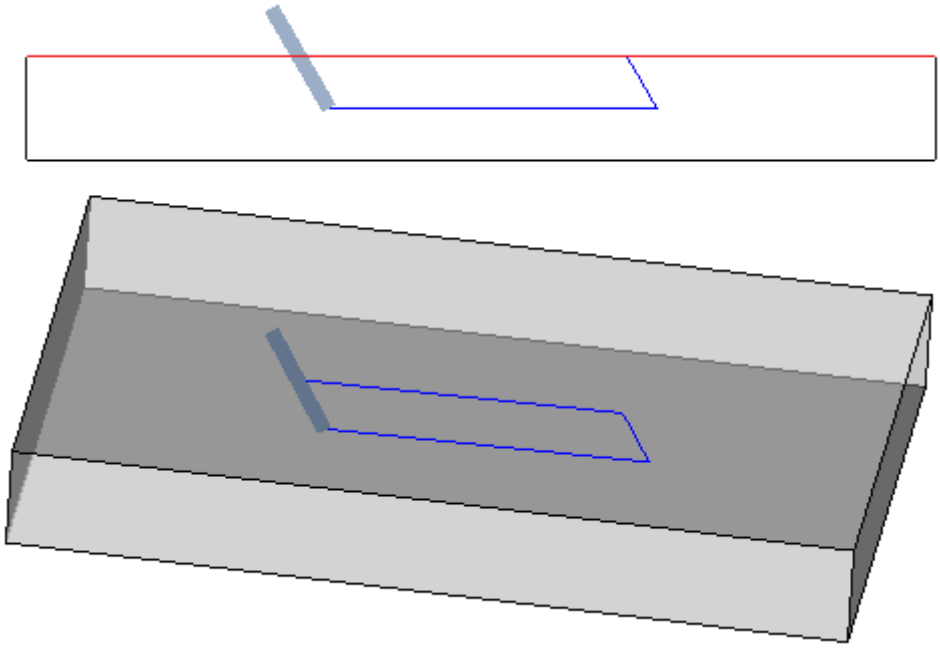
- **Показать габарит профилей в трехмерном изображении:** подключает или отключает отображение габаритов в трехмерном представлении операций профиля. Габарит выделяет использование инструмента на детали с оценкой по запрограммированной глубине.



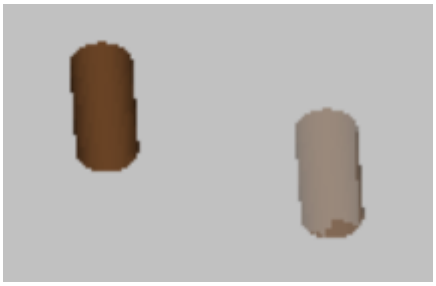
На рисунке представлен пример трехмерного изображения профилей с активированным отображением габаритов: для каждого отрезка профиля выделен габарит инструмента в детали. Умолчание - дезактивировано.
Применение выбора зависит от состояния пункта **Габариты трехмерного графического изображения** в меню Показать. Значение выбора может меняться из локального меню, которое реализовано в пункте **Габариты трехмерного графического изображения** в меню "Показать".
Если профиль программируется с ориентируемой подготовкой, подготовка представлена вычерченным ориентированным цилиндром, а ассоциированный с ним профиль отображается отрезками, ориентированными на основе параметров, заданных в этой подготовке.

На двух чертежах показан пример трехмерного представления ориентированного профиля с возможностью отображения габаритов. Подготовка показана слева с кончиком инструмента, обращенным вниз. Отрезок профиля выделяет:

- габарит инструмента в детали (отрезок внизу);
- отрезок контакта с плоскостью грани (отрезок вверху, с видом детали спереди).



- **Показать габарит точек и подготовки сплошным отрезком:** если эта опция подключена, отображение габаритов в трехмерном представлении точечных и подготовительных операций содержит показанный сплошным отрезком цилиндр (на рисунке слева). Если эта опция отключена: отображение содержит контуры габарита цилиндра в прозрачности (на рисунке справа).



- **Адаптировать габарит профилей к длине отрезков:** если эта опция подключена, отображение габаритов профилей ограничивается исходя из длины отрезков. Выбор позволяет уменьшить общее число графических элементов на профилях, фрагментированных на множество маленьких отрезков. В частности:
  - для длины отрезков применено значение, заданное в пункте **Минимальная длина отрезка с показом стрелок** (см. следующие поля в окне)
  - применен совокупный критерий на последующих отрезках
  - для отрезков, считающихся "короткими", отображение габаритов исключает конечную часть

- **Отключить графические элементы, добавленные для большой программы**: если эта опция подключена, отображение дополнительных графических элементов (стрелки, крайние точки, габарит в трехмерном изображении, габарит профили) задано как отключенное при открытии большой программы. При оценке большой программы рассматриваются два элемента:
  1. размер в кб файла, заданного на странице **Среда ->Сохранение** (поле, связанное с пунктом **Запрашивать подтверждение оптимизации для большой программы**)
  2. число операций, обрабатываемых программой, если их больше 100000. (используемое значение является переменным, с отклонением от 5000 до 100000, в зависимости от размера в килобайтах набора файлов).

- **Отобразить форму фрез:** выберите для отображения инструментов типа фрезы с реальной формой, например, конической или в целом фигурной. Выбор осуществляется как в случае программирования инструмента так и по диаметру, с реальным применением, совместимым с используемой технологией. Если элемент не выбран, инструменты типа фрезы отображаются с цилиндром.

- **Отобразить форму сверл:** выберите для отображения инструментов типа сверла с реальной формой, например, расклешенной или в целом фигурной. Выбор осуществляется как в случае программирования инструмента так и по диаметру, с реальным применением, совместимым с используемой технологией. Если элемент не выбран, инструменты типа сверла отображаются с цилиндром.

Графические элементы

- **Размер стрелок направления профиля:** длина двух отрезков отображения стрелки направления на отрезках профиля. Задается в: [мм] или [дюймах] (единицы измерения конфигураций). Это поле принимает значения от 0,5 мм до 100 мм.
- **Минимальная длина отрезка с показом стрелок:** минимальная длина, которую должен иметь отрезок профиля для обеспечения отображения стрелки направления. Задается в: [мм] или [дюймах] (единицы измерения конфигураций). Это поле принимает значения от 0,5 мм до 100 мм.
- **Диаметр крайних точек профиля:** диаметр отображения крайних точек на отрезках профиля. Представление крайних точек определяется минимальной длиной, которую должен иметь отрезок профиля, установленную на двойную величину заданного здесь значения. Задается в: [мм] или [дюймах] (единицы измерения конфигураций). Это поле принимает значения от 0,5 мм до 20 мм.
- **Диаметр для изображения точек и подготовки при нулевом габарите:** диаметр отображения точечных и подготовительных операций, для которых задан нулевой диаметр инструмента. Если операция - конструкция, это значение не используется. Задается в: [мм] или [дюймах] (единицы измерения конфигураций). Это поле принимает значения от 0,5 мм до 20 мм.
- **Длина инструмента для графики подготовки при нулевом габарите**: длина представления цилиндра инструмента подготовки при задании нулевой длины для инструмента или при незаданной технологии. Если операция - конструкция, это значение не используется. Задается в: [мм] или [дюймах] (единицы измерения конфигураций). Это поле принимает значения от 0.0 мм до 50 мм.
- **Размер дополнительных графических элементов:** размер графических элементов, добавленных при выполнении интерактивной процедуры (чертеж, инструменты). Это значение используется, например, для отображения следующих элементов:
  - закладки;
  - элемент выбора профилей;
  - точки крепления к профилю;
  - значимый элемент на этапе позиционирования;
  - стрелки перемещения оси.

  Задается в: [мм] или [дюймах] (единицы измерения конфигураций). Это поле принимает значения от 5 мм до 100 мм.

Последняя группа выбора позволяет персонализировать графическое представление особых профилей:

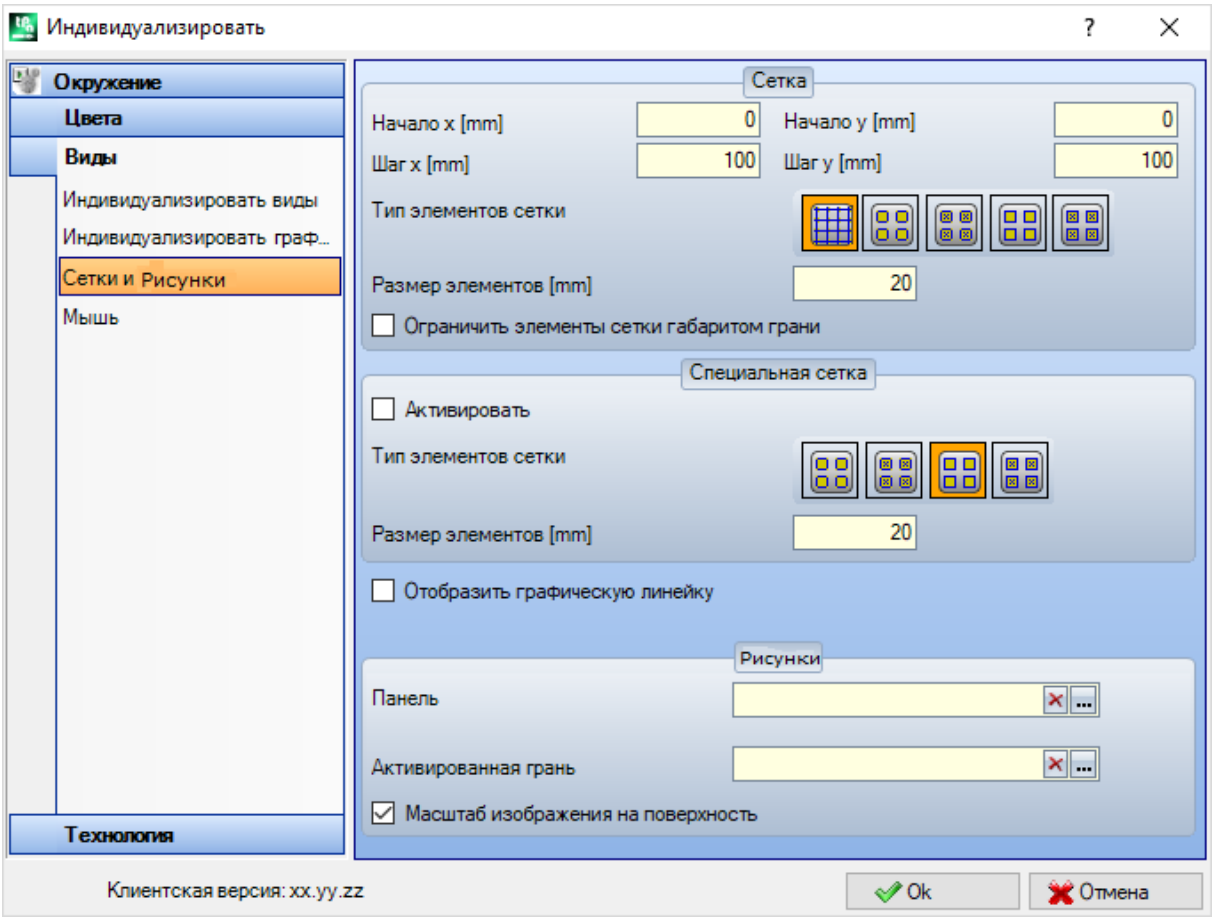
- **Геометрический профиль:** - это профиль, который при подготовке выбрал параметр **Геометрический профиль**
- **Профиль выемки:** - это профиль, который при подготовке выбрал параметр **Профиль выемки**
- **Профиль конструкции:** - это профиль, который при подготовке задал **поле B** с ненулевым значением (чисто положительным).

Для каждого из этих типов профиля можно подключить представление следующих графических элементов:

- крайние точки,
- стрелки,
- габарит в трехмерном изображении,
- габарит профилей.

Умолчание для всех этих пунктов соответствует неактивным пунктам.
Профиль может задаваться геометрического типа для сокращения процедуры графики. Типовым применением является разработка кривой ISO. Профиль задается профилем выемки, если он был добавлен при помощи процедуры выемки.

Сетка и Рисунки



Сетка

Эти настройки задают сетку прямоугольных координат с разверткой решеткой в плоскости XY текущей грани. Сетка не отображается на *Общем виде детали* или при активном *Трехмерном виде*.

- **Начало координат x:** начало координат x сетки. Задается в: [мм] или [дюймах] (единицы измерения конфигураций). Умолчание - 0.0.
- **Начало координат y:** начало координат y сетки. Задается в: [мм] или [дюймах] (единицы измерения конфигураций). Умолчание - 0.0.
- **Шаг x:** шаг сетки по оси x грани. Задается в: [мм] или [дюймах] (единицы измерения конфигураций). Это поле принимает минимальное значение, соответствующее 1 мм.
- **Шаг y:** шаг сетки по оси y грани. Задается в: [мм] или [дюймах] (единицы измерения конфигураций). Это поле принимает минимальное значение, соответствующее 1 мм.
- **Тип элементов сетки:** отображены опции отображения элементов сетки:
  - **Линии:** сетка представлена горизонтальными и вертикальными линиями, расстояние между которыми определяется заданными шагами. Точки пересечения линий - это точки сети. Это опция по умолчанию
  - **Пустые круги:** сетка представлена с пустыми кругами, расположенными по центру точек сетки.
  - **Перечеркнутые круги:** сетка представлена с перечеркнутыми кругами, расположенными по центру точек сетки
  - **Пустые квадраты:** сетка представлена с пустыми квадратами, расположенными по центру точек сетки
  - **Перечеркнутые квадраты:** сетка представлена с перечеркнутыми квадратами, расположенными по центру точек сетки
- **Размер элементов:** размер элементов сетки. В случае кругов задает диаметр кругов, в случае квадратов задает сторону квадратов. Задается в: [мм] или [дюймах] (единицы измерения конфигураций). Это поле принимает минимальное значение, соответствующее 1 мм.
- **Ограничить элементы сетки габаритом грани**: выбрать для запроса представления элементов сетки внутри зоны грани. Этот выбор активен в случае, если сетка представлена не линиями. Даже если этот пункт не выбирается, отображение за пределами габаритов грани ограничено.

Специальная сетка

Эти настройки задают сетку, заданную непосредственно для отдельных точек, определенную на этапе конфигурации производителем станка. Специальная сетка также определяется в плоскости xy текущей грани, но только в случае граней 1 или 2.
Кроме того, представление всегда ограничивается элементами внутри зоны грани.
Этот выбор недоступен для функции *Essential*.

- **Активировать:** подключает или отключает управление специальной сеткой. Умолчание - дезактивировано.
- **Тип элементов сетки:** отображены опции отображения элементов сетки:
  - **Пустые круги:** сетка представлена с пустыми кругами, расположенными по центру точек сетки Это опция по умолчанию
  - **Перечеркнутые круги:** сетка представлена с перечеркнутыми кругами, расположенными по центру точек сетки
  - **Пустые квадраты:** сетка представлена с пустыми квадратами, расположенными по центру точек сетки
  - **Перечеркнутые квадраты:** сетка представлена с перечеркнутыми квадратами, расположенными по центру точек сетки
- **Размер элементов:** размер элементов сетки. В случае кругов задает диаметр кругов, в случае квадратов задает сторону квадратов. Задается в: [мм] или [дюймах] (единицы измерения конфигураций). Это поле принимает минимальное значение, соответствующее 1 мм. Значение по умолчанию - 20 мм.

- **Отобразить графическую линейку:** подключает или отключает отображение линеек рядом с двухмерным отображением или каркасное представление текущей грани. Линейка не отображается в случае *Трехмерного вида.*

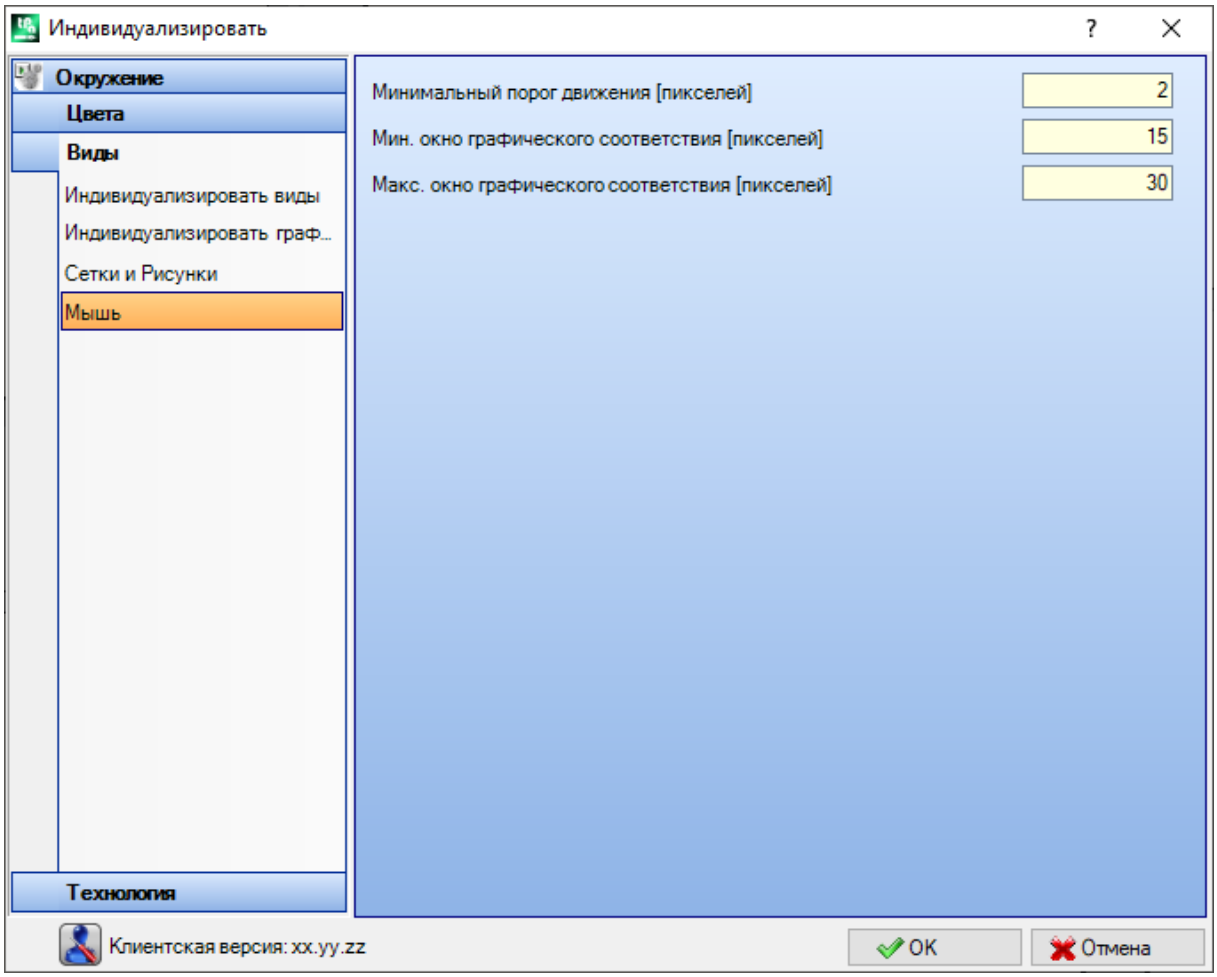
Рисунки
Может выбираться графический рисунок для заполнения **Панели** и **Активной грани.** Название рисунка может редактироваться в клетке редактирования или при щелчке на значке  . Открывается окно с файлами изображений, записанными в папке конфигурации (TPACADCFG\CUSTOM\DBPATTERN): действительны следующие форматы: *.PNG, *.JPG, *.BMP, и требуется выбрать файл в заданной папке. Для удаления имени заданного рисунка щелкнуть значок  .

Некоторые дополнительные соображения должны быть сделаны при использовании графическго рисунка панели в программе с определенным направлением зерна, горизонтального или вертикального.
Если найден файл с тем же именем + "_gx" или "_gy", автоматически определяется, какой файл загрузить.
Например, используйте файл *patternA.jpg*:
- ✓ с доступным файлом *patternA_gx.jpg*: *patternA_gx.jpg* будет использоваться с горизонтальным зерном, *patternA.jpg* для других случаев
- ✓ с доступным файлом *patternA_gy.jpg*: *patternA_gy.jpg* будет использоваться с вертикальным зерном, *patternA.jpg* для других случаев
- ✓ с доступными файлами *patternA_gx.jpg* и *patternA_gy.jpg*: *patternA_gx.jpg* будет использоваться с горизонтальным зерном, *patternA_gy.jpg* будет использоваться с вертикальным зерном, *patternA.jpg* в случае отсутствия заданного зерна.

Если вместо этого доступен толтько файл *patternA.jpg*:
- ✓ *patternA.jpg* будет использоваться без изменений с горизонтальным зерном или без заданного зерна
- ✓ *patternA.jpg* будет использоваться повернутым на 90 ° с вертикальным зерном.

- **Масштаб изображения на поверхность:** указывает режим позиционирования графических рисунков. При подключении этого пункта адаптирует изображение к полезной площади (панели или активной грани), в противном случае воспроизводит изображение сбоку до заполнения полезной площади.

Мышь



- **Минимальный порог движения [пикселей]:** расстояние в пикселах, которой должен пройти указатель на экране до определения изменения состояния. Эта настройка позволяет, например, избежать непроизвольных поворотов детали. Кроме того, эта настройка используется в интерактивном считывании (например, в функции чертежа) в качестве фильтра для подключения поиска объекта привязки. Умолчание - 2, могут вводиться значения в интервале [1; 10].

- **Мин. окно графического соответствия [пикселей]:** минимальный размер окна поиска графического соответствия. Умолчание - 10, могут вводиться значения в интервале [1; 50].

- **Макс. окно графического соответствия [пикселей]:** максимальный размер окна поиска графического соответствия. Умолчание - 20, могут вводиться значения в интервале [1; 50], но не меньше, чем в предыдущем поле.

Эти два значения определяют графическую зону поиска, которая применяется при графическом обучении (при поиске объектов привязки или текущей операции). Со ссылкой на пример, приведенный в параграфе *Ввод геометрических элементов из меню "Начертить"*, при задании значений (10 и 20 для двух полей) выполняется максимум две попытки графического поиска: первая - в зоне размерами 10 пикселов, отцентрованной по положению мыши, вторая - в зоне размерами 20 пикселов. Если заданы значения (7 и 20), выполняется максимум три попытки на зонах размерами, соответственно, 7, 14, 20. Переход между зонами осуществляется с шагами, кратными минимальному размеру.

13.4 Технология

Данные, отображаемые на страницах этой группы, могут меняться только при закрытой рабочей программе и в рабочей среде *Станка*.

Коды по умолчанию

В первой таблице заданы технологии, принятые для подготовительных или точечных операций в случаях, когда необходимо выполнить задание по умолчанию.
Примеры применения этих заданий:
- выполнение инструментов, требующих ввода подготовки при открытии нового профиля
- применение сложных операций, например, создание текстов или выемок
- ввод геометрического элемента *Точка* из меню Чертеж
- направление на выполнение открытых профилей или профилей, начинающихся геометрическим кодом подготовки, или операций с точечным геометрическим кодом (вытекающих, например, из процедуры импорта внешнего формата).

Процедура, выполняемая для определения технологии:
- выбрать строку грани, которой необходимо задать технологию

Грань	Код
Все	задает общую технологию для всех граней. Применяется, когда грань не имеет задания своей технологии
Грань 1	задает технологию для грани 1 (верхней), а также для возможных фиктивных и автоматических граней, отвечающих критерию подобия
...	
Грань 6	задает технологию для грани 6 (конечной), а также для возможных фиктивных и автоматических граней, отвечающих критерию подобия
Фиктивные грани	задает комплексную технологию для всех фиктивных и автоматических граней или только для граней, не отвечающих критерию подобия с одной из шести реальных граней

- выбрать в таблице строку грани для работы. При щелчке на значке ... одного из двух полей под таблицей открывается окно, в котором можно выбрать операцию из предусмотренных и задать на ней технологические данные. Из списка выбираемых операций исключаются:
  - операции, недоступные в палитре операций;
  - операции с программированием в полярной системе координат.

Кроме того:
- для подготовки: перечислены только операции подготовки;
- для точечной операции: перечислены точечные и подготовительные операции.

Должно уделяться особое внимание при задании размерных технологических параметров (координат и скоростей), т. к. они должны задаваться с единицей измерения, определенной в конфигурации ([мм] или [дюймы] для координат; [м/мин.] - [мм/мин.] или [дюймы/с] - [дюймы/мин.] для скоростей. Задания могут выполняться в числовом или параметрическом формате: в любом случае, сигнализируется возможная ошибка параметрического программирования. В точечной операции по умолчанию задание параметра "Диаметр" выполняется по следующим правилам:
- если операция точечного геометрического кода не имеет заданное значение диаметра, выполняется замена
- если операция точечного геометрического кода имеет заданное значение диаметра, замена **не** выполняется

Для стирания настройки в главной таблице щелкнуть значок клетки кода по умолчанию.

Таблица **Глобальные технологии** задает максимум 8 технологий, значимых для операций подготовки, используемых в форме параметрического программирования. Каждая строка в таблице может задавать подготовку по обычной процедуре, которая рассматривалась по применению технологической подготовки.
Для подключения подготовки выбрать соответствующую клетку в столбце ВКЛ и задать символическое имя подготовке для использования в программировании: как указано в заголовке столбца, распознаваемая параметрическая форма - типа “tec\namesetup”. Для каждой активированной подготовки имя обязательно и должно быть однозначным.
Для изменения технологических заданий подготовки два раза щелкнуть мышью (или нажать кнопку F2) на правой клетке соответствующей строки для открытия окна.
Для отключения уже заданной подготовки убрать флажок выбора в соответствующей клетке в столбце ВКЛ.
Глобальная технология может использоваться во всех операциях, предусматривающих возможность задания технологии подготовки при помощи ???ИМЕНИ, понимаемого как поле ИМЯ запрограммированной подготовки: теперь подготовка вызывается параметрическим именем и программирование не требуется.

Технология по умолчанию

Эта вкладка отображается только если в базе данных операций имеется операция, использованная для задания технологии по умолчанию. Выбор технологии не дифференцирован для грани и касается заданий технологии общей подготовительной или точечной операции.

Можно задать технологические параметры, соответственно, для:

- **Уставки по умолчанию для setup:** касается операций подготовки и сложных кодов типа профиля (кодов, которые в базе данных операций имеют заданный Подвид операции, равный 1; например, операции: Паз, Створка).
- **Уставки по умолчанию для точечных операций:** касается точечных операций и сложных кодов типа сверления (которые в базе данных операций имеют заданный Подвид операции, равный 0; например, операции: Ряд отверстий для сверления, Распределение отверстий на круге).

Задания в этом окне меняются при необходимости изменения задания по умолчанию параметра по сравнению с тем, что предлагается при вводе, делая его, например, неизменяемым.
Например, при задании значения = 1 параметру Станок для Подготовки:
- при каждом новом приращении операции setup (или: Паза, Створки) поле Станок будет предлагаться заданным на 1, в любом случае, изменяемым;
- эта настройка не меняет программирование или интерпретацию уже введенных операций.

Если необходимо принудительно задать программирование параметра Станок=1 (например, потому что в приложении реализован только Станок 1), в окне для станка необходимо задать:
- "(1)": значение, заключенное в круглые скобки, или "v,1". При таком задании значение Станка всегда будет 1, а параметр остается видимым при введении/изменении операции setup (или: Паза, Створки), но не может быть изменен (обозначение "v," означает "view,")
- "[1]": значение, заключенное в квадратные скобки, или "h,1". При таком задании значение Станка всегда будет 1, а параметр будет невидимым при вводе/изменении операции setup (или: Паза, Створки) и неизменяемым (обозначение "h," означает "hide,").

Обе настройки меняют программирование или интерпретацию уже запрограммированных операций. В примере поле Станок всегда принудительно задано на 1.

Другие действительные задания:
- "()": никакого значения в круглых скобках или "v". При таком обозначении параметр остается видимым в окне ввода операций, но не может быть изменен и не имеет заданного задания.
- " []": никакого значения в квадратных скобках или "h". При таком обозначении параметр не будет отображаться в окне ввода операций, не может быть изменен и не имеет заданного задания.

Такое принудительное задание параметров должно использоваться для особых заданий технологии установки, например, установки,
- состоящей только из одного **Станка** и/или только одной **Группы** и/или только одного **Фрезеровального** инструмента,
- исключающей программирование **Электрошпинделя**.

ВНИМАНИЕ:
- в любом случае, не меняются параметры, уже определенные как неизменяемые в базе данных операций;
- могут задаваться только параметры: Станка, Группы, Шпинделя, Инструмента, Типа инструмента, Диаметра, Скорости и Вращения инструментов;
- заданное значение может быть выражено в числовом или параметрическом формате, в любом случае, сигнализируется возможная ошибка параметрического программирования
- заданные здесь настройки не интегрируются с настройками предыдущей страницы **Коды по умолчанию**.

13.5 Персонализировать файл "прототип"

Как уже говорилось, при создании программы в качестве исходной точки используется файл прототипа по умолчанию.

Для открытия и правки файла прототипа выбрать команду "Открыть файл прототипа"  из меню
Применение: открывается файл PIECE.TCN в папке TPACADCFG\CUSTOM.

В зависимости от конфигурации TpaCAD можно реализовать различные прототипы для каждого задаваемого типа: программа, подпрограмма или макропрограмма. В этом случае: выбор команды приводит к последующему выбору необходимого файла, как это уже происходит при создании новой программы.

В частности, файлу прототипа можно задать не минимальный уровень доступа и/или записи для предупреждения неуполномоченных изменений. В любом случае, уровень доступа и/или записи, заданный новой программе, приводится к минимальному, требуемому для типа детали, что, в случае программы, соответствует уровню *Оператор*.
Прототип для создания детали типа программы может также задавать другой тип: например, подпрограмму.

14 Создание клиентских операций

"Клиентская" операция - это всегда сложная операция, т. е. совокупность операций, задаваемая с целью скрытия от оператора сложности операции, облегчая выбор и задание параметров и свойств.

База данных операций, инсталлируемая с приложением TpaCAD, обеспечивает доступность многих кодов сложных операций, которые обычно основываются на макропрограмме.
Рассматриваемая здесь функция реализует возможность задания сложных кодов также конечным пользователем, основываясь на подпрограммах, написанных самим пользователем. Можно задать максимум 100 "клиентских" операций.

Команда **Клиентские операции**  по определению сложной операции выбирается из меню  при закрытой программе. Эта команда не доступна в меню, если база данных операций не задает исходную операцию (эталонную операцию), которая необходима для создания клиентских операций.

Для создания сложной операции нажать кнопку  : открывается окно Открыть Деталь для выбора вызываемой подпрограммы.
При доступе к уровню производителя можно выбирать также файл типа макропрограммы.

- **Имя операции:** имя операции. Предлагается имя по умолчанию, например, W + (cop), где cop=рабочий код, заданный автоматически операции (первый свободный код среди имеющихся для этой группы операций). Можно задать число символов минимум 2 и максимум 10, символами могут быть только буквы или цифры, первый символ должен быть буквой. Считаются зарезервированными и поэтому не могут использоваться буквенные имена из 2 символов, начинающиеся с буквы "W" (например, "WC", "WB"..), т. к. они предназначены для внутреннего использования. Выбранное имя не может быть именем, уже определенным для другой операции или для параметра этой операции, включая операции и параметры, заданные для перезадаваемых переменных. Имя операции в формате ASCII начинает программную строку в таблице Формат ASCII. Задание этого поля обязательно.
- **CodOP:** рабочий код операции. Значение этого поля задается автоматически и не может быть изменено
- **Подпрограмма:** содержит подпрограмму, применяемую операцией, которая здесь указывается как имя.расширение. В зоне Текстовой помощи, приведенной внизу, указывается полный маршрут файла. Это поле не может быть изменено
- **Наименование:** описательное имя операции. Это поле начинается с имени подпрограммы (например, W4901), имеет длину максимум 30 знаков и не вводится в файл языка, т. е. непереводимо.
- **Активировать:** при выборе этого поля оно вводит операцию в палитру операций. Даже если не подключено, задание операции должно быть полностью действительным.
- **Кнопка выбора:** отображает имя группы операций, в которую вводится операция. При нажатии кнопок  и  обновляется выбираемое изображение, с прокруткой групп вкладки Операции. Клиентская операция добавляется к выбранной группе операций.

 Эта кнопка позволяет задавать новую группу для ввода клиентской операции. При выборе этой кнопки открывается папка для поиска изображения, присваиваемого новой группе. Имя, определяющее новую группу, задается автоматически.

ВНИМАНИЕ! Кнопки выбора группы операции не отображаются, если палитра графического выбора задана непосредственно развернутой единой группе.

Затем могут конфигурироваться типичные параметры кода вызова подпрограммы (например, Зацепление точки, Относительный, Координаты размещения, Грань для применения, Угол вращения, Зеркальное отображение по горизонтали, Зеркальное отображение по вертикали и Инвертировать) в случаях, в которых:
- если они сконфигурированы производителем станка в эталонной операции;
- не превышаются 30 параметров;
- для каждого параметра приводится имя ASCII, заключенное в квадратные скобки.

Выбрать это поле для подключения отображения и последующего управления параметром в окне операции.
Особым параметром является параметр **Грань** применяемой подпрограммы, для которой задается клетка выбора и поле редактирования:
- выбрать это поле для подключения отображения и последующего управления параметром в окне операции;
- поле редактирования задает значение параметра по умолчанию, которое может меняться напрямую только если параметр управляется напрямую. Выделяются особые случаи функционирования:

- для принудительного применения определенной грани подпрограммы оставить параметр неподключенным и задать номер грани в поле редактирования (например, 1);
- для принудительного применения вложенных вызовов оставить параметр неподключенным и оставить поле редактирования пустым или со значением, заданным на 0 или -1;
- для того, чтобы оставить доступность двух предыдущих случаев, задать подключенный параметр и оставить поле редактирования пустым или с исходным значением по умолчанию, но, в любом случае, изменяемым.

- **Переменные r:** содержит перезадаваемые переменные подпрограммы, которые становятся параметрами сложной операции. Для каждой переменной задаются: имя ASCII, описание, состояние подключения, значение по умолчанию, тип ввода поля.
  Максимум рассматриваются первые 50 перезадаваемых переменных подпрограммы.
  - **Имя:** имя переменной в формате ASCII. Можно задать от 1 до 10 буквенно-цифровых символов. Первый символ должен обязательно быть буквой. Считаются зарезервированными и поэтому не могут использоваться буквенные имена из 2 символов, начинающиеся с буквы "W" (например, "WC", "WB"..), т. к. предназначены для внутреннего использования, а также имена параметров, уже заданные в эталонной операции (например, Зацепление точки, Относительный,..) и в имени ASCII этой операции.
  - **Наименование:** описательное имя параметра (например, “Сдвиг x”). Это поле начинается символическим именем переменной или, если оно не задано, с наименования переменной или, если оно не задано, - как R+(nn), где nn= номер переменной (например, “R0”, “R27”). Это поле имеет максимальную длину 30 символов и не вводится в файл языка, т. е. непереводимо.
  - **Активировать**: при выборе подключает прямое задание поля. Если это поле не подключено, задание поля соответствует полю **Значение**, без возможности изменения.
  - **Значение**: значение по умолчанию, предлагаемое при вводе операции. Это поле задано на значение, заданное в подпрограмме. Если подключена опция **Задать поле управления,** отображается клетка выбора вместо поля редактирования значения. Можно задать параметрическое значение.
  - **Задать поле управления:** отображает ячейку выбора вместо поля редактирования для задания значения.

Если **Имя** переменной r назначено равным имени, используемому для идентификации технологического поля электрошпинделя или инструмента, автоматически переменная будет связана с возможностью запуска открытия технологического окна, с возможностью интерактивного выбора значения.

Если необходимо обновить сложный код, заданный для подпрограммы, нажать кнопку  **.** Тот же результат получается при выборе кнопки  и соответствующую подпрограмму. При этом приложение:
- проверяет, что уже задан код, определенный для подпрограммы;
- отыскивает и проверяет уже заданную информацию;
- предлагает результирующие настройки.

Для завершения клиентской операции необходимо предусмотреть:
- файл изображения для загрузки в состав группы вкладки Операции (задание файла считается необходимым). Файл должен архивироваться в папке TPACADCFG\CUSTOM\DBBMP с именем, состоящим из “W” + (рабочий код) или, в качестве альтернативы, - с тем же именем, заданным операции, и в известных форматах изображений (*.png; *.jpg; *.bmp);
- файл изображения для загрузки в качестве контекстуальной графической справки при задании операции (задание файла может считаться необязательным). Файл должен архивироваться в папке TPACADCFG\CUSTOM\DBBMPHLP с именем и форматом, заданным для предыдущего пункта.

Для удаления клиентской операции из списка переместить выбор на операцию и нажать кнопку  . Для отображения и проверки ввода данных созданной операции нажать кнопку  .

Кнопки  и  перемещают выбранную операцию, устанавливая другой порядок представления в соответствующей группе операций.

15 Программы преобразования

15.1 Из формата DXF в формат TpaCAD

При стандартной инсталляции устанавливается один модуль импорта из формата DXF:
TpaSpa.DxfCad.v2.dll.

- **DxfCad**: доступный начиная с версии TpaCAD 1.4.2.

Документацию см. в руководстве импортера.

15.2 Из формата TpaCAD в формат DXF

При стандартной инсталляции устанавливается один модуль экспорта в формат DXF:
TpaSpa.DxfCad.v2.dll.

Программа в формате DXF задает только операции, в которых были соблюдены логические ограничения, как они заданы в детали, архивированной в среде TpaCAD. Исключаются из преобразования системные логические операции (циклы IF..ELSE.. ENDIF, ERROR, EXIT, задание переменных J). Сложные операции (профиля, применения подпрограммы или макропрограммы) разбиваются и каждое параметрическое задание решается и заменяется числовым заданием.

Параметры

- **Слой детали**: имя слоя, заданное в файле DXF для определения геометрии габарита детали. В файле DXF генерируется полилиния прямоугольной формы (длина на высоту детали):
  ✓ при экспорте в 3D: полилинии присваивается толщина, соответствующая толщине детали.
  Заданное имя назначается слою полилинии.

- **Слои**: эта группа указывает имена слоев, заданных типам операций, в случае, если не применяются конкретные задания (см. ниже: страница настроек **Операции и Слои**). Настройки группы не могут быть изменены

- **Создать отдельные геометрические элементы**: выбрать для создания отдельных геометрических элементов, не в полилинии, каждый высотой, соответствующей конечной координате Z отрезка (линейного или дуги). Соответствует функционированию по умолчанию. Если не выбирается:
  ✓ профиль с постоянной глубиной генерирует одну полилинию
  ✓ профиль с переменной глубиной генерирует полилинию для каждого элемента (линии или дуги) с координатой Z, заданной на конечную глубину этого элемента

- **Разделитель полей**: выбирает знак для интерпретации в качестве разделителя между полями. Возможен выбор между знаками #_ %-+ (никакого знака, решетка, подчеркивание, процент, минус, плюс).

- **Рассчитать грани 3D-вида**: выбирает поле, чтобы включить экспорт 3D-детали. Умолчание активировано. Если поле выбрано:
  - Для операций, заданных на базовых гранях, рассматриваются 6 базовых граней и грань-деталь. Получаемый файл DXF соответствует трехмерному чертежу, с системой координат граней, которая преобразуется в прямоугольную систему координат, а программирование оси Z глубины реализовано как для положительных, так и для отрицательных значений.

  Если поле не выбрано: обработана только грань 1 и файл DXF соответствует двухмерному чертежу.
  - **Слой детали**: выбрать, чтобы экспортировать слой детали в двухмерный чертеж. Если поле не выбрано: экспортирует только объекты, соответствующие операции (в грани 1).

Операции и Слои

Эта страница позволяет ассоциировать уровень элемента DXF с исходной операцией.

На этой странице можно задать до 40 ассоциаций для точечных или подготовительных операций.

В файле DXF слой характеризуется именем с 100 символами:

- первые символы, которые мы назовем префиксом, ассоциируются с операцией, имеющейся в базе данных операций TpaCAD, среди точечных или подготовительных операций;
- остальные символы ассоциируются с параметрами и/или свойствами операции.

Префиксы операций

Префикс:

- включает от 2 до 30 буквенно-цифровых символов (первый символ не может быть числовым);
- не может повторяться в таблице

Каждой строке, заданной в первой таблице, можно задать индикаторы для параметров и/или свойств операции TpaCAD в последующей таблице.

Список рассматриваемой здесь информации только частично соответствует информации, которая напрямую управляется при задании операции в TpaCAD: теперь добавлены поля, производные из *компиляции* операций. Примером для всех является технологическая информация для подготовки фрезы:

- в TpaCAD обычно программируется рабочий инструмент с заданием станка, группы, электрошпинделя и инструмента
- среди доступной теперь информации имеется также диаметр инструмента, как его считывает технология установки.

Префиксы параметров

Индикаторы параметров и/или свойств задаются одним алфавитным символом. Для каждого префикса можно иметь до 30 общих индикаторов параметров и свойств.

В случае параметра, напрямую задаваемого при задании операции в TpaCAD, в таблице приводится также описательное сообщение параметра.

Задание параметра нечислового типа не имеет никакого действия на преобразование.

Запрограммированные операции

Ниже приводится перечень операций, которые рассматриваются модулем экспорта.

Точечные операции (рабочий код в интервале: 1-1000)

Каждой точечной операции соответствует:
✓ один геометрический круг, если диаметр инструмента не равен нулю;
✓ в противном случае - одна геометрическая точка.
Высота геометрического элемента соответствует запрограммированной координате Z.
В случае операции, не сконфигурированной на странице **Операции и Слои**, задается уровень "BOR".

Операции подготовки (рабочий код в интервале: 1-1000)

Каждой неизолированной операции setup соответствует: одна полилиния высотой, соответствующей координате Z подготовки.
Особый случай - профиль, соответствующий прорези (подготовка резца, за которой следует линейный отрезок): возможно и обычно рекомендуется конфигурировать специальный уровень, чтобы можно было дифференцировать прорезь от заданного иначе профиля.

В случае операции, не сконфигурированной на странице **Операции и Слои**, задается слой "ROU".

Если подготовка изолированная, преобразуется:
✓ один геометрический круг, если диаметр инструмента не равен нулю;
✓ в противном случае - одна геометрическая точка.
В случае операции, не сконфигурированной на странице **Операции и Слои**, задается слой "SET".

Операции профиля линейного типа

Каждой линейной операции соответствует одна линия в полилинии.
Заданный слой - тот же, который интерпретируется для операции Setup ("ROU", в случае незаданной ассоциации).

Операция профиля типа дуги (плоскость xy)

Каждой операции типа дуги соответствует одна дуга в полилинии.
Если это круг: если изолированный, генерирует круг; в противном случае преобразует в полилинию из двух полукругов.
Заданный слой - тот же, который интерпретируется для операции Setup ("ROU", в случае незаданной ассоциации).

Операция профиля типа дуги (плоскость не xy)

Дуга, заданная на плоскости, отличной от плоскости (xy) грани, должна подойти к преобразователю разбитой на ломаную линейных отрезков.
Каждый отдельный отрезок ломаной преобразуется в линию полилинии и действительны приведенные выше положения.
Если на преобразователь приходит дуга, заданная на плоскости, отличной от xy: не преобразует отрезок.

15.3 Из формата ISO в формат TpaCAD

Ниже описываются характеристики функционирования модуля импорта из формата ISO, включенного в стандартную инсталляцию.
Процедура преобразования должна быть активизирована на этапе конфигурации производителем станка.

Уставки, предусмотренные для преобразования, задаются в диалоговом окне. Уровень доступа к окну конфигурации решается изготовителем станка.
Перед тем, как подробно рассматривать предусмотренные задания, рассмотрим общие критерии, принятые на этапе преобразования файла в формате ISO.

Преобразуются операции профиля и сверления с заданием только на грань 1.
Ниже приводится фрагмент действительного файла ISO с выделенными полужирным шрифтом интерпретируемыми полями:

(FLAT 20MM 2F EC HSS)
G71
G0 X-627.857**Y**0**Z**312.249 **B**13.135 **A**0 **S**12000 **T**4;…(комментарий) ..
G40
G1 **X**-2.272**Y**0**Z**-9.738 P0.22724Q0R0.97384 F6000 T1
G1 **X**888.346**Y**0**Z**-217.56 P0.22724Q0R0.97384 T1 B13.134
G1 **X**898.083**Y**0**Z**-219.832 P0.22722Q0R0.97384 T1 B13.134
...
M2

Файл признается действительным, если первая строка начинается с одного из следующих символов: **%** (процент), **(** (круглая открывающая скобка), **;** (точка с запятой), **:** (двоеточие), **[** (квадратная открывающая скобка), **/** (символ "/"), **O** (буква "O"), **P** (буква "P"), **G** (буква "G"), **N** (буква "N"), **M** (буква "M"), **T** (буква "T"), **S** (буква "S").

Интерпретация формата ISO не делает различия между прописными и строчными буквами (т. е. не чувствительна к регистру), например, не имеет значения, будет ли "g10" или "G10".

- Первая строка в круглых скобках, считанная до значимой команды, задает комментарий программы (в примере - (FLAT 20MM 2F EC HSS);
- строки, начинающиеся с символа **%** (процент), **(** (круглая открывающая скобка) и **;** (точка с запятой), не интерпретируются
- Символ **;** (точка с запятой), находящийся в строке файла, превращает в комментарий следующую за ним часть строки;

- единица измерения по умолчанию файла ISO - [мм]. Для прямого задания единицы программирования, а также размеров детали, необходимо задать поле **G70/ G71** до первой **G0** (а не в неинтерпретируемых строках):
  "G70X20**Y**12**Z**3.9" задает единицу измерения файла ISO в дюймах [inch] и размеры детали (длина=20 дюймов; высота=12 дюймов, толщина 3.9 дюйма)
  "G71X1300**Y**1300**Z**80" задает единицу измерения файла ISO в [мм] и размеры детали (длина=1300 мм; высота=1300 мм; толщина 80 мм)

В примере мы видим **G71** на второй строке без задания размеров детали. В этом случае они задаются автоматически, включая положительный габарит на всех осях координат.

Интерпретация профиля начинается со строки **G0** (движения в быстром режиме. В нашем примере это третья строка и в ней интерпретируются следующие поля:
- (**X**, **Y**, **Z**) как исходные координаты профиля
- (**B**, **A**) как исходные значения осей поворота (затем они приводятся на подготовке профиля, если он задан с поворотными осями)
- **G90** / **G91** для программирования абсолютных/инкрементных координат
- **T4** выбор инструмента
- **S12000** скорость вращения шпинделя

На одной строке с G0 или на следующей строке может интерпретироваться характеристика коррекции на радиус фрезы:
G40 без коррекции (умолчание)
G41 в левую часть профиля
G42 в правую часть профиля.

Каждая строка профиля, следующая за первой, может задать:
1. линейный отрезок интерполяции **G1** и интерпретируются поля (**X**, **Y**, **Z**) как конечные координаты линейного отрезка и **G90/G91** для программирования абсолютных или инкрементных координат. Незаданная координата распространяется из предыдущего отрезка. Может интерпретировать скорость интерполяции в поле **F** (единица измерения: [мм/мин.] или [дюймов/мин.]), которая преобразуется в единицы программирования, как задано в конфигурации TpaCAD.

2. отрезок круговой интерполяции **G2/3** (соответственно, по/против часовой стрелки) и интерпретируются поля:
   (**X**, **Y**, **Z**) как конечные координаты закругленного отрезка
   G17/G18/G19 для программирования плоскости развертки дуги (соответственно, XY (умолчание), ZX, YZ)
   (**I, J, K**) как координаты центра. Значимы 2 координаты, соответствующие плоскости дуги (при относительном программировании или как вытекает из программирования **G90/G91)**
   G90/G91 для программирования абсолютных (умолчание) или инкрементных координат.

   Может интерпретировать скорость интерполяции в поле **F**.
   В случае дуги в плоскости ZX (G18), если приложение TpaCAD решает дуги в первичной плоскости Xz, направление вращения дуги инвертировано.

   В случае дуги в плоскости XY (G17=умолчание), если обе координаты центра (I, J) не заданы и если дуга не решает круг, решает дугу с программированием радиуса (**R..**). Радиус должен иметь значение, как минимум равное (эпсилон*10.0) и обязательно не ниже расстояния между крайними точками дуги, в противном случае преобразование прерывается вследствие ситуации ошибки.
   Если в плоскости ZX или в плоскости YZ не заданы обе координаты центра, преобразование прерывается, т. к. это является ситуацией ошибки.
   В случае линий профиля (G1, G2, G3), не имеющих соответствия до операции G0, преобразование аннулируется.

3. новое начало профиля **G0**

4. Интерпретация отверстия интерпретирует код **G81**, а на этой строке интерпретирует поля:
   - (**X**, **Y**, **Z**) как исходные координаты профиля
   - **G90** / **G91** для программирования абсолютных/инкрементных координат
   - **T4** выбор инструмента
   - **S12000** скорость вращения шпинделя
   - **F100** скорость входа инструмента

5. возможные строки с другими кодами **G** игнорируются.

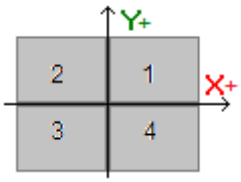
Интерпретация программы завершается в конце файла или если интерпретируется поле **M2**.

Настройки

Теперь рассмотрим настройки, предусмотренные на этапе конфигурации модуля импорта:

- **Код G размеров**: задать код G, которому будет соответствовать интерпретация размеров детали. Как уже говорилось для кодов G70/G71, интерпретация которых, в любом случае, активна для задания единицы измерения и размеров. Здесь можно задать другой код с действительным значением от 100 до 10000
- **Код G сверления**: задать значение в интервале (81-89) для интерпретации как операция сверления

- **Квадрант станка**: задать значение от 1 до 4 для интерпретации координат XY, считанных в файле ISO. Настройка, отличная от 1, соответствует интерпретации файла ISO в координатах станка. Со ссылкой на чертеж:

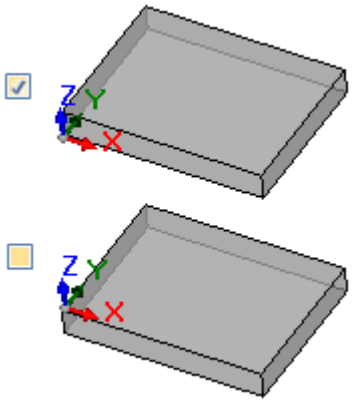


  - **1** соответствует ситуации по умолчанию, для которой координаты не меняются
  - **2** соответствует ситуации координат X в отрицательной зоне станка: импорт изменяет координаты X, приводя их в положительную область
  - **3** соответствует ситуации обеих координат (X,Y) в отрицательной зоне станка: импорт изменяет координаты X и Y, приводя их в положительную область
  - **4** соответствует ситуации координат Y в отрицательной зоне станка: импорт изменяет координаты Y, приводя их в положительную область

  Изменение рабочих координат выполняется в соответствии со считанными (или выведенными) размерами для файла.

- **Вращающиеся оси, которые назначают (B, A):** настройка касается считывания кривых ISO непосредственно из программы и, в частности, интерпретации вращающихся осей. Выбор находится в списке из трех элементов и указывает пару вращающихся осей, которые считаются для назначения осей (B, A):
  - (B, A): задание использует ту же номенклатуру
  - (A, C): ось A кривой ISO назначает B, ось C кривой ISO назначает ось A
  - (B, C): ось B кривой ISO назначает B, ось C кривой ISO назначает ось A.

- **Абсолютная система координат оси Z**: выбрать это поле для интерпретации координат Z в абсолютной системе координат. В противном случае координаты Z интерпретируются непосредственно в системе грани 1.
  При активном выборе импорт изменяет координаты Z, приводя их в систему грани 1.



- **Координаты центров применяют G90/G91**: выбрать это поле для подключения интерпретации координат центров как для кодов g90/g91. В противном случае координаты центров всегда интерпретируются инкрементными относительно точки начала дуги.

- **Удалить изолированные коды G0**: выбрать это поле для удаления из импорта кодов g0, которые не продолжаются движениями на линейной и/или закругленной траектории. Как правило, речь идет о

позиционированиях в быстром режиме, выполняемых в воздухе на максимальной скорости, разрешенной для осей, соответствующих нулевым положениям детали, расцеплению, смене инструмента и не эффективных для интерпретации операций на детали.

- **Подготовка фрезы**: настройки, относящиеся к преобразованию вертикальной Подготовки фрезы. Случай соответствует профилю без назначения вращающихся осей.
  - Первое поле соответствует списку вариантов, соответствующих доступным настройкам.
  - Второе поле позволяет напрямую назначать параметры для выбранной обработки.
    - пример настройки - «TMC = 1 TR = 2», соответствующий назначению технологических параметров Станка (1) и Группы (2)
    - в настройке можно использовать только имена параметров в формате ASCII, а значение должно быть разделено символом '=' и числовым символом
    - в любом случае поля, которые соответствуют параметрам, уже независимо управляемым модулем импорта, игнорируются и исключаются (координаты применения, вращающиеся оси, частота вращения, поправки на радиус фрезы)
  - можно выбрать операцию вертикальной или ориентированной Setup

- **Подготовка фрезы (ориентированной)**: настройки, относящиеся к преобразованию ориентированной Подготовки фрезы. Случай соответствует профилю с назначениями вращающихся осей. Назначения полностью аналогичны предыдущему случаю.

15.4 Из формата TpaCAD в формат ISO

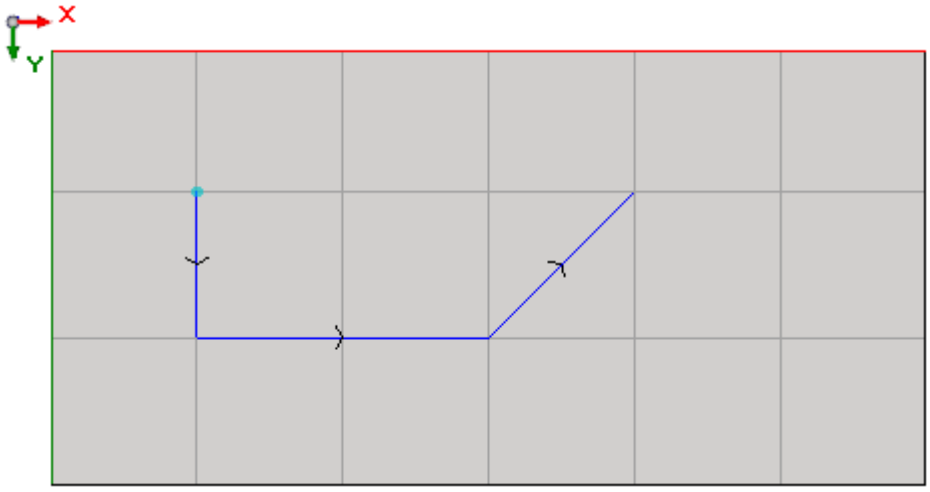
Ниже описываются характеристики функционирования модуля экспорта в формат ISO, включенного в стандартную инсталляцию (**TpaToIso**).
Процедура преобразования должна быть активизирована на этапе конфигурации производителем станка.
Преобразование может применяться только к программам или подпрограммам.

Преобразование применяется только к операциям, запрограммированным в грани 1 (напрямую или с грани-детали) или, если грань 1 не реализована, - в грани 2 (напрямую или с грани-детали) и к сверлениям и профилям.
Программа в формате ISO задает только операции, в которых были соблюдены логические ограничения, как они заданы в детали, архивированной в среде TpaCAD. Кроме того, исключаются из преобразования системные логические операции (циклы IF ..ELSE.. ENDIF, ERROR, EXIT, задание переменных J). Сложные операции (профиля, применения подпрограммы или макропрограммы) разбиваются и каждое параметрическое задание решается и заменяется числовым заданием.
Модуль преобразования загружает технологический параметр, чтобы получить полезную информацию для состава файла, который создается. Уведомление не предоставляется, если действительные технологии не проверены.

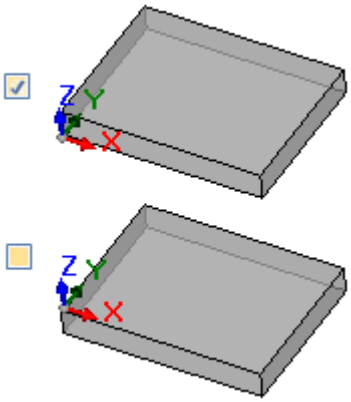
Настройки

- **Абсолютная система координат осей XY**: выберите поле для показания координат XY в абсолютную систему отсчета. Выбор касается только координатных осей XY и применяется в случаях, когда TpaCAD управляет системой XY для грани 1, отличной от стандартной. В этом случае:
  - при активном выборе, координаты XY преобразуются в абсолютную систему с началом осей в правом нижнем углу. В процессе преобразования некоторая информация также может быть изменена, например: направление вращения дуг и корректирующая сторона профиля
  - при неактивном выборе, координаты XY остаются неизменными.

Со ссылкой на рисунок:
- источник программирования находится в верхнем левом углу
- профиль запрограммирован в:
  - SETUP X100 Y100
  - L01 Y200
  - X300
  - X400 Y100
- посмотрим, как профиль преобразуется в систему XY по умолчанию (источник в нижнем левом углу)
  - SETUP X100 Y200
  - L01 Y100
  - X300
  - X400 Y200



- **Абсолютная система координат оси Z**: выберите поле для показания координат Z в абсолютную систему отсчета вместо грани 1. При активном выборе толщина детали добавляется ко всем координатам Z. На чертеже показаны две системы отсчета:
  - **абсолютная** (на вершине)
  - **грани** (в нижней части)



- **Сверления:** операция сверления может быть переведена с помощью кода G, соответствующего фиксированному циклу или с циклом фрезерования
  - **Выполняет цикл фрезерования:** выберите поле для перевода сверлений с фрезерными циклами (с подробными движениями входа и выхода из детали)
  - **Код G сверления:** установите значение в диапазоне (81-89) для использования в качестве цикла сверления

- **Фрезерование: решает вход / выход:** выберите поле для перевода фрезерования, детализируя движения входа и выхода из детали

- **Координаты центра применяют G90/G91:** выберите поле для показания координат центров, соответствующих кодам g90 / g91. В противном случае координаты центров всегда задаются инкрементно относительно начальной точки дуги.

- **Z безопасности:** установите координату Z безопасности для движений над деталью. Единица программирования: [мм], в диапазоне [10,0; 1000,0]. Значение устанавливает положение оси Z в абсолютной системе отсчета.

- **Координата в воздухе:** установите координату Z в воздухе по умолчанию, которая будет использоваться в случае неназначенного технологического значения. Единица программирования: [мм], в диапазоне [2.0; 500.0]. Значение задает положение оси Z в системе граней 1: значение 5,0 соответствует положению оси Z на 5,0 мм над деталью.

- **Строки заголовка**: можно установить до 5 строк, показанных в начале программы, перед линиями, соответствующими операциям. Первая строка созданного файла назначается как «%0».

Пример может соответствовать строки, которая присваивает системный блок и размеры исходной детали « **G7%u X%l Y%h Z%s** », в котором используются некоторые параметрические формы (синтаксис: «%» + буква):

- «%u» заменяется символом, соответствующим единице измерения: «0» если (дюймы [inch]), «1» если (мм)
- «%l» заменяется длиной детали
- «%h» заменяется высотой детали
- «%s» аменяется толщиной детали

Другие управляемые параметрические формы:

- «%o#» (с # = 0..7) заменяется переменной (o0, .. o7) детали
- «%v#» (с # = 0..7) заменяется переменной (v0, .. v7) детали
- «%n» заменяется значением, соответствующим режиму выполнения детали: 0 = нормальное выполнение, 1 = зеркало x, 2 = зеркало y, 3 = зеркало xy
- «%x», «%y», «%z»: заменяются координатой (X, Y, Z) отсчета рабочей области, как назначены в детали.

- **Строки нижнего колонтитула:** можно установить до 5 строк, показанных в конце программы, после строк, соответствующих операциям. В конце созданного файла добавляется строка «M2».

Синтаксис и примеры

Ниже приведен пример файла продукта:

%0
(TpaToIso By TPA Srl)
G71X800.0Y450.0Z80.0
..
G90G40
G0X100.0Y-65Z-12.5 A10B-60 T4M12S12000
G01X250 F4000
G02G17X..Y..I..J..
..
M02

- **%0** : фиксированная строка заголовка
- (TpaToIso By TPA Srl): инвариантная строка комментария
- **G71**X800.0Y450.0Z80.0: единица измерения и размеры детали: G71 единица измерения SI (мм, мм / мин), G70 англосаксонская единица измерения (дюйм [inch], дюйм / мин)
- .. (другие строки заголовка)
- **G0.**..: строки инструкций
- ..
- .. (строки нижнего колонтитула):
- **M02 ..** закрытие программы

Фрезерование

Перевод профиля определяется опцией

- **Фрезерование: решает вход / выход**

Если поле не выбрано, давайте посмотрим, как переводится операция setup:

- **G90G40**
- G90 абсолютное программирование (все переведено на G90)
- G40: спецификация, относящаяся к коррекции радиуса фрезы
  - G40 без коррекции (по умолчанию)
  - G41 слева от профиля
  - G42 справа от профиля

- **G0**X100.0Y-65Z-12.5 A10B-60 T4M12S12000M3F3000
  - операция setup переводится с кодом G0:
  - X..Y ... Z .. координаты осей
  - A .. круговая ось вокруг X
  - B .. круговая ось вокруг Y
  - T .. выбор шпинделя или инструмента (если строго положительный)
  - M .. переводит поле M операции (если строго положительно)
  - S .. скорость вращения шпинделя (если строго положительная)
  - M3 / M4 вращение шпинделя (M3 = по часовой стрелке, M4 = против часовой стрелки)
  - F ... скорость входа ([мм / мин], [дюйм / мин]) - (если строго положительно)

Поле **M** можно использовать для активации вспомогательной функции.
Поля **S** и **F** показывают значение, запрограммированное в оригинале или присвоенное технологическому параметру инструмента.
Информация **M3 / M4** о вращении шпинделя показывает значение, присвоенное технологическому параметру инструмента.

Давайте посмотрим, как переводится операция setup с выбранной опцией **Фрезерование: решает вход / выход**:

- T4 M6 — выбор и активация шпинделя
- S12000 M3 — настройки шпинделя (скорость и вращение: M3 = по часовой стрелки, M4 = против часовой стрелки)
- G90G40 — абсолютное программирование, компенсация на инструмент (G40 / G41 / G42)
- **G0** X100.0Y-65 Zout — быстрая (X, Y) рабочая, Z до безопасной высоты (Zout)
- M12 — вспомогательная функция
- **G0 Z в воздухе** — быстрое движение Z в воздухе
- **G1** Z-12,5 F3000 — интерполированное движение Z на рабочей высоте, скорость F входа

Первые две строки, относящиеся к настройкам шпинделя, отсутствуют, если им предшествует выполнение профиля по той же технологии.
Для полей **S, F** и **M3 / M4** применяются те же соображения, которые описаны выше.
Значение **Zout** показывает значение, назначенное в настройке **Z безопасности**, возможно, связанное с системой грани.
Координата в воздухе **Z в воздухе** показывает значение, присвоенное технологическому параметру инструмента или, если не установлена, настройке **Координата в воздухе**.

Профиль продолжается строками линейной или круговой интерполяции:

- **G01**X250 F4000 M55
  - линейная интерполяция переводится с кодом G01
  - X ... Y..Z .. координаты осей (не показанные оси не перемещаются)
  - F .. скорость интерполяции: показывает значение, запрограммированное в оригинале или присвоенное технологическому параметру инструмента.
  - M .. переводит поле M операции (если строго положительно)

- **G02**G17X..Y..I..J..F.. M55
- **G03**G17X..Y..I..J..F.. M55
  - круговая интерполяция переводится с кодом G02 (вращение по часовой стрелке) или G03 (вращение против часовой стрелки)
  - X ... Y..Z .. координаты осей (не показанные оси не перемещаются)
  - G17 плоскость круговой интерполяции: G17, если плоскость XY (по умолчанию), G18, если плоскость ZX, G19, если плоскость YZ. Если он не назначен, он распространяет значение последнего назначения
  - I..J..K .. координаты центра соответственно по осям X, Y, Z. Две координаты, соответствующие указанной плоскости, являются значимыми (в абсолютном или относительном режиме, как установлены **Координаты центра применяют G90/G91**)
  - F .. скорость интерполяции: показывает значение, запрограммированное в оригинале или присвоенное технологическому параметру инструмента.
  - M .. переводит поле M операции (если строго положительно)

Поле **F** отображается в первом элементе профиля (G1 / G2 / G3) и в последующих элементах, только если запрограммировано изменение.
Поле **M** отображается только в том случае, если вдоль профиля имеется изменение.

С выбранной опцией **Фрезерование: решает вход / выход**, профиль заканчивается линиями:

- **G0** Zout — быстрое движение Z до уровня безопасности (Zout)
- M5 — остановка шпинделя

Функция **M5** остановки шпинделя отсутствует, если сопровождает выполнение профиля по той же технологии.

Опрация сверления

Перевод операции сверления определяется опциями

- **Сверление: выполняет цикл фрезерования**
- **Код G сверления**

Если поле **Сверление: выполняет цикл фрезерования** не выбрано, обработка переводится с фиксированным циклом (пример: G81)

- **G81**G90 X100.0Y-65Z-12.5 T4M12S12000F3000
  - пунктуальная обработка переводится с кодом **G81** (соответствует настройке **Код G сверления**)
  - G90 абсолютное программирование (все переведено на G90)
  - X ... Y..Z .. координаты осей
  - T .. выбор шпинделя или инструмента (если строго положительно)
  - M .. переводит поле M операции (если строго положительно)
  - S .. скорость вращения шпинделя (Част. вращения= об./мин.) - (если строго положительно)
  - F ... скорость входа ([мм / мин], [дюйм / мин]) - (если строго положительно)
  - D.. запрограммированный диаметр, если нет выбора шпинделя или инструмента.

В особых случаях поле **T** может содержать множественный выбор несколько инструментов с синтаксисом «Tv1 / v2, v3, .., vn»:

- ✓ v1 = первый инструмент (ссылка на позицию)
- ✓ v2 = второй инструмент
- ✓ ..
- ✓ vn = последний инструмент.

Поля **S** и **F** показывают значение, запрограммированное в оригинале или присвоенное технологическому параметру инструмента.

Если поле **Сверление: выполняет цикл фрезерования** выбрано, операция переводится с циклом фрезерования.
Давайте посмотрим, как та же обработка, что и выше, переводится с кодом G81:

- T4 M6 выбор и активация шпинделя
- S12000 M3 настройки шпинделя (скорость и вращение: M2 = по часовой стрелки, M4 = против часовой стрелки)
- G90G40 абсолютное программирование, отключает компенсацию на инструмент
- G0 X100.0Y-65 Zout быстрая (X, Y) рабочая, Z до безопасной высоты (Zout)
- M12 вспомогательная функция
- **G0 Z в воздухе** быстрое движение Z на уровне воздуха
- **G1** Z-12,5 F3000 интерполированное движение Z на рабочей высоте
- **G0** Zout быстрое движение Z до уровня безопасности (Zout)
- M5 остановка шпинделя

Для полей **T**, **S** и **F** применяются те же соображения, которые описаны выше.
Информация **M3 / M4** о вращении шпинделя показывает значение, присвоенное технологическому параметру инструмента.
Функция остановки шпинделя **M5** отсутствует, если сопровождается выполнением профиля по той же технологии.

15.5 Из формата TpaCAD в формат Edicad

Процедура преобразования должна быть активизирована на этапе конфигурации производителем станка. Преобразование может применяться только к деталям типа программы или подпрограммы.

Режим перекодировки

Общая информация о детали

Общие задания детали преобразуются следующим образом:
- **Единицы измерения и размеры:** берутся в формате Edicad. Для получения программы в дюймах [дюймы] в конфигурации TpaCAD необходимо дезактивировать преобразование единиц измерения на этапе создания матрицы детали. В противном случае программа в формате Edicad будет преобразована в [мм].
- **Комментарий:** берется в формате Edicad, максимум 250 символов.
- **Переменные "o"**: первые три переменные "o" приводятся в сдвигах детали в среде Edicad в числовом формате. Каждая параметрическая форма решается.
- **Переменные "v"**: первые восемь переменных "v" приводятся в переменных системы детали в среде Edicad в числовом формате. Каждая параметрическая форма решается.
- **Переменные "r"**: задания, относящиеся к переменным "r", теряются.
- **Изменяемые геометрии**: заданные фиктивные грани восстанавливаются. Задание сводится на три ребра, в числовом формате. Каждая параметрическая форма решается. Напрямую решаются также задание исходной грани или определение фиктивной грани на геометрии, отличной от задания трех ребер в прямоугольной системе: в детали в формате Edicad имеются координаты трех ребер грани, в любом случае, сводимые к абсолютной системе детали. Восстанавливаются также фиктивные грани, заданные пустыми или как вспомогательные грани построения. В переменных геометриях восстанавливаются также автоматические грани, заданные программой на грани-детали. В этом случае нумерация грани приводится к нумерации фиктивных граней с занятием первых свободных номеров. Теряются задания, относящиеся к толщине фиктивных граней и направлению оси z.
- **Клиентские разделы**: все задания утрачиваются.
- **Последовательности:** все задания утрачиваются.

Запрограммированные операции

На каждой грани преобразование осуществляется для максимум 32500 операций, заданных в матрице; лишние операции будут потеряны.
Это преобразование касается всех граней, заданных в TpaCAD, включая грань-деталь. В этом случае:
- операции грани-детали размещаются на соответствующих гранях задания до операций, которые могли быть заданы непосредственно на гранях;
- автоматические грани преобразуются в фиктивные грани.

Программа в формате Edicad задает только операции, в которых были соблюдены логические ограничения, как они заданы в детали, архивированной в среде TpaCAD.
Кроме того, исключаются из преобразования системные логические операции (циклы IF ..ELSEIF..ELSE.. ENDIF, ERROR, EXIT, задание переменных J).
Сложные операции (профиля, применения подпрограммы или макропрограммы) разбиваются и каждое параметрическое задание решается и заменяется числовым заданием.

Точечные

Точечные операции имеют рабочий код от 1 до 1000.
Рабочий код [81] в среде TpaCAD решает программирование отверстия для инструмента и для диаметра; в среде Edicad код [81] решает программирование отверстия для диаметра, а код [82] решает программирование отверстия для инструмента.
При таком допущении код [81] переводится в:
- код [81]: если нет заданного инструмента (поле инструмента со значением: 0);
- код [82]: если в поле инструмента задано значение, отличное от нуля.

Для всех остальных случаев точечной операции преобразование дает рабочий код как он задан в матрице.
Для всех точечных операций действуют следующие правила преобразования:

формат TpaCAD	**формат** Edicad
поле L	Задает в поле "Слой" (только значение до 8)
поле O	Задает в поле "Начало координат" (только значение до 3)
поле M	Задает в поле M
координата X точки применения	Задает в поле Qx
координата Y точки применения	Задает в поле Qy
координата Z точки применения	Задает в поле Zp
Станок	Задает в соответствующем поле, только если значение отлично от 0
Группа	Задает в соответствующем поле, только если значение отлично от 0
Инструмент	Задает в соответствующем поле, только если значение отлично от 0

Тип инструмента	Задает в соответствующем поле, только если значение отлично от 0
Диаметр инструмента	Задает в соответствующем поле, только если значение отлично от 0
Скорость вращения	Задает в соответствующем поле, только если значение отлично от 0
Рабочая скорость	Задает в соответствующем поле, только если значение отлично от 0
Координата замедления на входе	Задает в соответствующем поле, только если значение отлично от 0
Координата замедления на выходе	Задает в соответствующем поле, только если значение отлично от 0
Клиентские параметры	Задает тот же клиентский параметр, только если значение отлично от 0

Клиентские логики

Логические операции имеют рабочий код от 1 до 1000.

Преобразование содержит рабочий код в соответствии с заданным в матрице.
Для всех логических клиентских операций действуют следующие правила преобразования

формат TpaCAD	**формат Edicad**
поле L	Задает в поле "Слой" (только значение до 8)
поле O	Задает в поле "Начало координат" (только значение до 3)
поле M	Задает в поле M
Станок	Задает в соответствующем поле, только если значение отлично от 0
Группа	Задает в соответствующем поле, только если значение отлично от 0
Инструмент	Задает в соответствующем поле, только если значение отлично от 0
Тип инструмента	Задает в соответствующем поле, только если значение отлично от 0
Диаметр инструмента	Задает в соответствующем поле, только если значение отлично от 0
Скорость вращения	Задает в соответствующем поле, только если значение отлично от 0
Рабочая скорость	Задает в соответствующем поле, только если значение отлично от 0
Координата замедления на входе	Задает в соответствующем поле, только если значение отлично от 0
Координата замедления на выходе	Задает в соответствующем поле, только если значение отлично от 0
Клиентские параметры	Задает тот же клиентский параметр, только если значение отлично от 0

Подготовка

Операции подготовки имеют рабочий код от 1 до 1000.
Преобразование содержит рабочий код в соответствии с заданным в матрице. Для всех операций подготовки действуют следующие правила преобразования:

формат TpaCAD	**формат Edicad**
поле L	Задает в поле "Слой" (только значение до 8)
поле O	Задает в поле "Начало координат" (только значение до 3)

поле M	Задает в поле M
координата X точки применения	Задает в поле Qx
координата Y точки применения	Задает в поле Qy
координата Z точки применения	Задает в поле Zp
Станок	Задает в соответствующем поле, только если значение отлично от 0
Группа	Задает в соответствующем поле, только если значение отлично от 0
Инструмент	Задает в соответствующем поле, только если значение отлично от 0
Тип инструмента	Задает в соответствующем поле, только если значение отлично от 0
Диаметр инструмента	Задает в соответствующем поле, только если значение отлично от 0
Скорость вращения	Задает в соответствующем поле, только если значение отлично от 0
Рабочая скорость	Задает в соответствующем поле, только если значение отлично от 0
Координата замедления на входе	Задает в соответствующем поле, только если значение отлично от 0
Координата замедления на выходе	Задает в соответствующем поле, только если значение отлично от 0
Координата оси C (вращение)	Задает в соответствующем поле
Координата оси B (горизонтальное вращение)	Задает в соответствующем поле
Клиентские параметры	Задает тот же клиентский параметр, только если значение отлично от 0

Если матрица генерирована с применением коррекции на инструмент, задания, соответствующие коррекции подготовки фрезы, преобразованием не задаются.
При подготовке резца параметры коррекции всегда задаются преобразованием, т. к. не применимы в матрице.

Действуют следующие правила преобразования:

формат TpaCAD	формат **Edicad**
Выбор Коррекции на инструмент (Выкл/Левый/Правый)	Задает в соответствующем поле
Радиус коррекции	Задает в соответствующем поле, только если значение отлично от 0

Результат коррекции на инструмент в TpaCAD и в Edicad может отличаться: TpaCAD улучшает восстановление ситуаций, реализованных иначе или нереализованных в Edicad, и добавляет новые характеристики, которые при переходе на Edicad были бы утрачены (сокращение профилей, изменения коррекции в профиле).
Генерирование матрицы с применением коррекции на инструмент устраняет эти различия: программы, импортированные в Edicad, уже скорректированы.

Профиль

Линейный тип
Преобразование содержит рабочий код L01 [2201].
Действуют следующие правила преобразования:

формат TpaCAD	**формат Edicad**
координата X точки применения	Задает в поле Xf, только если отлична от предыдущего отрезка
координата Y точки применения	Задает в поле Yf, только если отлична от предыдущего отрезка
координата Z точки применения	Задает в поле Zf, только если отлична от предыдущего отрезка
Скорость интерполяции	Задает в соответствующем поле, только если значение отлично от 0
Клиентские параметры	Задает тот же клиентский параметр, только если значение отлично от 0

Тип дуги (плоскость xy)
Преобразование содержит рабочий код A01 [2101].
Действуют следующие правила преобразования:

формат TpaCAD	**формат Edicad**
координата X точки применения	Задает в поле Xf, только если отлична от предыдущего отрезка
координата Y точки применения	Задает в поле Yf, только если отлична от предыдущего отрезка
координата Z точки применения	Задает в поле Zf, только если отлична от предыдущего отрезка
координата X центра	Задает в поле Cx
координата Y центра	Задает в поле Cy
направление вращения	Задает в поле вращения (0=по часовой стрелке, 1=против часовой стрелки)
Скорость интерполяции	Задает в соответствующем поле, только если значение отлично от 0
Клиентские параметры	Задает тот же клиентский параметр, только если значение отлично от 0

Тип дуги (плоскость xz)
Преобразование содержит рабочий код A05 [2105].
Действуют следующие правила преобразования:

формат TpaCAD	**формат Edicad**
координата X точки применения	Задает в поле Xf, только если отлична от предыдущего отрезка
координата Y точки применения	Задает в поле Yf, только если отлична от предыдущего отрезка
координата Z точки применения	Задает в поле Zf, только если отлична от предыдущего отрезка
координата X центра	Задает в поле Cx
координата Z центра	Задает в поле Cz
направление вращения	Задает в поле вращения (0=по часовой стрелке, 1=против часовой стрелки)
Скорость интерполяции	Задает в соответствующем поле, только если значение отлично от 0
Клиентские параметры	Задает тот же клиентский параметр, только если значение отлично от 0

В конфигурации TpaCAD можно выбрать записать в матрице детали вместо дуги в плоскости xz ломаную линию из линейных отрезков. В этом случае каждый отдельный отрезок ломаной преобразуется с рабочим кодом L01 [2201] и действуют положения, приведенные для операции профиля линейного типа.

Тип дуги (плоскость yz)
Преобразование содержит рабочий код A06 [2106].
Действуют следующие правила преобразования:

формат TpaCAD	**формат Edicad**
координата X точки применения	Задает в поле Xf, только если отлична от предыдущего отрезка
координата Y точки применения	Задает в поле Yf, только если отлична от предыдущего отрезка
координата Z точки применения	Задает в поле Zf, только если отлична от предыдущего отрезка
координата Y центра	Задает в поле Cy
координата Z центра	Задает в поле Cz
направление вращения	Задает в поле вращения (0=по часовой стрелке, 1=против часовой стрелки)
Скорость интерполяции	Задает в соответствующем поле, только если значение отлично от 0
Клиентские параметры	Задает тот же клиентский параметр, только если значение отлично от 0

В конфигурации TpaCAD можно выбрать записать в матрице детали вместо дуги в плоскости yz ломаную линию из линейных отрезков.

В этом случае каждый отдельный отрезок ломаной преобразуется с рабочим кодом L01 [2201] и действуют положения, приведенные для операции профиля линейного типа.

Тип дуги (плоскость xyz)
Дуга, заданная на общей плоскости (xyz), преобразуется в матрице детали в ломаную линию из линейных отрезков.
Критерии, определяющие режим создания ломаной, заданы в конфигурации TpaCAD.
Каждый отдельный отрезок ломаной преобразуется с рабочим кодом L01 [2201] и действуют положения, приведенные для операции профиля линейного типа.

15.6 Программа TpaCAD

Программа, которая может напрямую открываться в TpaCAD, представляет собой текстовый файл с особым синтаксисом.
Расширение по умолчанию, которое использует TpaCAD, - (.tcn).
Далее приводится описание формата для реализованных элементарных операций для обеспечения удобного сопряжения в случае, если требуется внешнее генерирование программы.

Рассмотрим структуру программы:

```
TPA\ALBATROS\EDICAD\02.00
$=проверка интерфейса TpaCAD
::UNm DL=1000 DH=800 DS=40
SIDE#1{
$=base
W#81{ ::WTp
#1002=10 #1=101 #2=102 #3=-15 #8015=0 #2005=1.5 #2002=3300 #9012=-5 #9013=-10 #1001=1 }W
}SIDE
SIDE#2{
$=низ
}SIDE
SIDE#3{
$=перед
}SIDE
SIDE#4{
$=конец
}SIDE
SIDE#5{
$=зад
}SIDE
SIDE#6{
$=начало
}SIDE
```

Предлагаемая здесь структура соответствует детали в виде параллелепипеда с минимальными общими заданиями (размеры и комментарий).

Строки заголовка

Теперь рассмотрим блоки, определяющие структуру.

```
TPA\ALBATROS\EDICAD\02.00
$=проверка интерфейса TpaCAD
::UNm DL=1000 DH=800 DS=40
```

Первая строка является обязательной для предварительного запуска при открытии программы.
Вторая строка, начинающаяся с "$=", задает комментарий к программе и не является обязательной. При наличии она должна иметь начало "$=", за которым должно следовать описание.

Третья строка является обязательной и задает единицу измерения и размеры:
- "::" заголовок
- "UNm" единица измерения в [мм] (умолчание); "UNi" единица измерения в [дюймах]
- "DL=1000 DH=800 DS=40" размеры: DL= длина, DH= высота, DS= толщина.

Поля разделены пробелом.

Раздел программы грани

```
SIDE#1{
$=base
W#81{ ::WTp
#1002=10 #1=101 #2=102 #3=-15 #8015=0 #2005=1.5 #2002=3300 #9012=-5 #9013=-10 #1001=1 }W
}SIDE
```

Первая строка обязательна для открытия раздела программы грани: "SIDE#1{" открывает раздел грани 1, … , "SIDE#6{" открывает раздел грани 6.
Вторая строка, начинающаяся с "$=", задает имя грани и не является обязательной. При наличии она должна иметь начало "$=", за которым должно следовать имя.
Следуют блоки, которые начинаются с "W#nn{ ::" и закрываются "}W", для определения операций грани.
Последняя строка ("}SIDE") является обязательной для закрытия раздела грани.

Не обязательно задавать разделы граней, не имеющих запрограммированных операций.

Раздел операции, заданной в программе грани

```
W#81{ ::WTp
#1002=10 #1=101 #2=102 #3=-15 #8015=0 #2005=1.5 #2002=3300 #9012=-5 #9013=-10 #1001=1 }W
```

Операция может задаваться на одной или нескольких строках текста. Формы, эквивалентные предлагаемой здесь операции, могут быть, например:

```
W#81{ ::WTp #1002=10 #1=101 #2=102 #3=-15 #8015=0 #2005=1.5 #2002=3300 #9012=-5 #9013=-10
#1001=1 }W
```

```
W#81{ ::WTp
#1002=10 #1=101 #2=102 #3=-15 #8015=0 #2005=1.5 #2002=3300 #9012=-5 #9013=-10 #1001=1
}W
```

```
W#81{ ::WTp #1002=10 #1=101 #2=102 #3=-15
#8015=0 #2005=1.5 #2002=3300 #9012=-5 #9013=-10 #1001=1
}W
```

```
W#81{ ::WTp
#1002=10 #1=101 #2=102 #3=-15
#8015=0 #2005=1.5 #2002=3300 #9012=-5 #9013=-10 #1001=1
}W
```

в которых различные части были помещены на одну строку или разбиты на несколько строк, но с сохранением без изменений некоторых синтаксических правил:
- поля в одной строке разделены пробелом;
- заголовок раздела имеет фиксированную структуру "W#*nn*{ ::WT*c*" (например: "W#81{ ::WTp"), где:
- nn= рабочий код (числовой) операции,
- c= символ, задающий тип операции ('p'=точечная, 's'=подготовка, 'l'=линия, 'a'=дуга);
- остальные поля имеют фиксированную структуру "*#nn=st*" (например: "#1002=10"), где:
- nn= числовой идентификатор параметра,
- st= значение, заданное параметру;
- закрытие раздела имеет фиксированную структуру "}W".

Операция: Отверстие

Заголовок получается: "W#81{ ::WTp".

Геометрические параметры:

#1=	Координата X применения
#2=	Координата Y применения
#3=	Координата Z глубины
#8015=	Координаты применения, выраженные в Абсолютных значениях = 0 (умолчание)/ Относительных значениях = 1

Технологические параметры при запрограммированном сверлении для диаметра:

#1002=	Диаметр инструмента
#201=	Станок
#203=	Группа
#1001=	Тип инструмента

Технологические параметры при запрограммированном сверлении для инструмента:

#201=	Станок
#203=	Группа
#205=	Инструмент
#1001=	Тип инструмента

Общие технологические параметры:

#2005=	Скорость врезки инструмента (м/мин.)
#2002=	Частота вращения шпинделя (об./мин.)
#9012=	Координата замедления на входе
#9013=	Координата замедления на выходе

Операция: Подготовка фрезы

Заголовок получается: "W#89{ ::WTs".

Геометрические параметры:

#1=	Координата X применения
#2=	Координата Y применения
#3=	Координата Z глубины
#8015=	Координаты применения, выраженные в Абсолютных значениях = 0 (умолчание)/ Относительных значениях = 1

Технологические параметры для выбора инструмента:

#201=	Станок
#203=	Группа
#205=	Инструмент
#1001=	Тип инструмента

Общие технологические параметры:

#2005=	Скорость врезки инструмента (м/мин.)
#2002=	Частота вращения шпинделя (об./мин.)
#40=	Применение коррекции на инструмент: 1 для коррекции слева, 2 для коррекции справа, 0 для нетребуемой коррекции (умолчание)

Операция: Линия

Заголовок получается: "W#2201{ ::WTl".

Геометрические параметры:

#1=	Координата X применения (конечная точка линейного отрезка)
#2=	Координата Y применения
#3=	Координата Z глубины
#8015=	Координаты применения, выраженные в Абсолютных значениях = 0 (умолчание)/ Относительных значениях = 1

Общие технологические параметры:

#2008=	Скорость интерполяции (м/мин.)

Операция: Дуга на плоскости для грани

Заголовок получается: "W#2101{ ::WTa".

Геометрические параметры:

#1=	Координата X применения (конечная точка дуги)
#2=	Координата Y применения
#3=	Координата Z глубины
#8015=	Координаты применения, выраженные в Абсолютных значениях = 0 (умолчание)/ Относительных значениях = 1
#31=	Координата X центра относительно начальной координаты X дуги
#32=	Координата Y центра относительно начальной координаты Y дуги
#34=	Направление вращения дуги: 0 - по часовой стрелке (умолчание), 1 - против часовой стрелки

Общие технологические параметры:

#2008=	Скорость интерполяции (м/мин.)

Tecnologie e Prodotti per l'Automazione S.r.l.

Via Carducci 221
I - 20099 Sesto S.Giovanni (MI)
Ph. +393666507029

www.tpaspa.com

info@tpaspa.it

26.01.2026