

PTP 400

User's Manual

T.P.R.

Information in this document is subject to change without notice on the part of T.P.A. Spa - Tecnologie Prodotti per l'Automazione -. No part of this manual may be reproduced without the permission of T.P.A. SpA.

This document is related to release SW 8.0.

Tecnologie Prodotti per l'Automazione SpA

20099 Sesto San Giovanni (MI) - Via Carducci, 221 -
Tel. (02) 26223139 - Fax 2486201

TABLE OF CONTENTS

1. GENERAL FEATURES AND STARTUP

IDENTIFICATION	1-1
PTP400'S HARDWARE STRUCTURE	1-1
CENTRAL UNIT'S HARDWARE STRUCTURE	1-1
PTP200N	1-2
ESPAS	1-2
INOUTR	1-2
POWER FEEDER	1-2
RACKS	1-2
CONFIGURATION OF THE CENTRAL UNIT	1-3
SUPPLY OF THE SYSTEM PTP400	1-5
INSTALLING THE SOFTWARE	1-5
PTP400'S CONFIGURATION SETTINGS	1-6
Selecting the language	1-6
Selecting the measure unit	1-7
Selecting the type of system XY	1-7
CONFIGURATION OF MACHINE CYCLES	1-8
CONFIGURATION OF LOCAL NETWORK	1-9
STARTING SYSTEM	1-10
TURNING THE SYSTEM OFF	1-11
NOTES ABOUT USING DISKS	1-11
NUMBERING THE INPUT/OUTPUT PORTS	1-12
PTP400'S MAIN FEATURES	1-17
MAIN MENU	1-19
Programs' edit	1-20
Functions' edit	1-20
Machine control mode	1-20
Parameters (settings)	1-20
Formatting program disks	1-20
Make directories	1-20
Programs' backup	1-20
System's auxiliary	1-21
Debug's auxiliary	1-21
Operator identification	1-21
Enter data and time	1-21
Make labels (barcode)	1-21
Version of PTP400	1-21
User's auxiliary	1-24
MS-DOS video file editor	1-24
Run MS-DOS commands	1-24
Restart PTP400	1-24
Exit on MS-DOS commands	1-25

2. MACHINE CONTROL MODE

AUTOMATIC (Machine control mode)	2-1
DETAILED DESCRIPTION OF MACHINE PERFORMANCES	2-3
F1 - AUTO	2-3
Repetitions	2-5
Article	2-5
Execute	2-7
Exit	2-9
Displays during programs execution	2-10
F3 - END	2-11
F2 - START	2-11
F4 - SET POINT	2-11
F5 - MANUAL	2-11
F6 - EXIT	2-11
HANDLING THE EXECUTION COUNTERS	2-12
SYSTEM OF REFERENCE XY	2-13
DEFINITION OF WORK AREAS	2-14
EXECUTION MODES IN WORK AREAS	2-15
Execution modes N and N/	2-16
Execution modes S and S/	2-16
Execution modes T and T/	2-16
Execution modes M and M/	2-17
Execution modes R and R/	2-17
Execution modes A and A/	2-17
EXECUTION MODALITIES IN OPTIONAL MODE "D"	2-23
MANUAL MODE AND DIAGNOSTIC	2-24
F1 - JOG-STEP	2-26
F3 - AXIS	2-27
F4 - SPEED	2-27
F5 - PITCH	2-27
F6 - IO CODE	2-28
F7 - SET	2-28
F8 - RES	2-28
F9 - INPUT	2-28
F2 - DacU	2-29
F10- EXIT	2-29

3. MACHINE PARAMETERS

PARAMETERS MENU	3-1
KEYBOARD AND FUNCTION KEYS	3-3
MEASURE UNIT	3-6
REMARKS WHEN DEALING WITH PARAMETERS	3-7
PARAMETERS DESCRIPTION	3-10
Number of tools	3-10
Head configuration	3-10
Mills radii	3-13
Tools correctors on axes X , Y , Z	3-13
Parameters of fitting tools and an axis W	3-16
Work fields' reference points	3-17
Air values and offsets	3-18
Depth of horizontal holes	3-21
Processing speed	3-22
Mandrels' rotation velocity	3-24
Selected axes	3-25
Axes' parameters	3-25
Input test for manual	3-26
Head's features	3-27
Parameters of horizontal tools' raising	3-28
Axes limits	3-28
Parameters of linearity correctors	3-29
Linearity correctors	3-29
Interpolation parameters	3-31
Machine emergencies	3-32
ERRORS DISPLAYED WHEN HANDLING PARAMETERS	3-34

4. PROGRAMS CREATION AND RUNNING

PROGRAMS ORGANIZATION	4-1
FORMAT PROGRAM DISK	4-2
MAKE PROGRAMS' DIRECTORIES	4-3
PROGRAMS' BACKUP	4-4
Menu 1° (selection modes)	4-4
Menu 2° (selection connections)	4-5
Menu 3° (run selected operations)	4-7
Messages when handling recordings	4-9
PROGRAMS' EDIT	4-11
Working procedures	4-12
Special keys	4-13
Mode "wait command"	4-15
Commands programmed on function keys	4-17
Insert	4-17
Revise	4-17
Delete	4-17
Merge	4-18
Line	4-18
List	4-18
Dimensions and measure units	4-19
Graphic	4-20
Kill	4-23
Store	4-23
Directory	4-24
Rename	4-24
New	4-24
End	4-24
Ctrl + T (head's description)	4-25
Errors displayed when handling the edit	4-26
 PTP400'S MILLING AND BORING COMMANDS	 4.1-1
Description of the functions	4.1-1
Preparatory functions (Gnn)	4.1-1
Axes co-ordinates (X, Y, Z)	4.1-3
Co-ordinate of the circumference center (I, J, K) ...	4.1-4
Circumference radius (I)	4.1-4
Circumference point of intermediate passage (I, J) .	4.1-4
Function of plane selection when milling (P)	4.1-5
Function of tool selection (T)	4.1-5
User's functions (F)	4.1-5
Velocity (VX VY VZ)	4.1-6
Mandrel's rotation velocity (VM)	4.1-8
Function "mill radius" (R)	4.1-9
Function (A)	4.1-9
Function "origin" (O)	4.1-10
Function "sub-program" (S)	4.1-10
Function "rotation" (D)	4.1-10

Description of commands display	4.1-11
Vertical boring command	4.1-11
Command "load Sub-program"	4.1-12
Horizontal boring commands	4.1-13
Milling commands	4.1-15
Blade grouoving commands	4.1-19
Fitting commands	4.1-20
Programmed Stop commands	4.1-21
Inserting user's functions	4.1-22
Inserting the special function G93	4.1-22
GENERAL EXAMPLES OF PROGRAMS' LAYOUTS	4.1-23
SPREADINGS AND AUTOMATIC SELECTIONS.....	4.1-24
Automatic spreading of the values	4.1-24
Automatic spreading of the milling plane	4.1-24
Automatic spreading of origins	4.1-25
Automatic selection of the tool (special tools)	4.1-25
Automatic selection of velocities	4.1-26
Automatic preservation of a programmed velocity	4.1-26
<u>5. USE OF DRILLING COMMANDS</u>	
FUNCTION "VERTICAL BORING"	5-1
FUNCTION "FITTING"	5-2
fitting with a fixed single head	5-2
fitting with independent mandrels	5-3
fitting with two parallel heads	5-5
automatic calculation of the fitting ref.points	5-6
ERRORS DISPLAYED WHEN EDITING THE FITTING	5-7
FUNCTIONS OF HORIZONTAL BORINGS	5-8
EXAMPLES OF BORING PROGRAMS	5-9
<u>6. USE OF MILLING COMMANDS</u>	
LINEAR MILLING FUNCTIONS	6-1
CIRCULAR MILLING FUNCTIONS	6-2
HELICOIDAL MILLING FUNCTIONS.....	6-4
PROGRAMMING OVALS	6-5
REMARKS ABOUT THE USE OF MILLING COMMANDS	6-15
ERRORS DISPLAYED WHEN EDITING MILLINGS	6-18
EXAMPLES OF MILLING PROGRAMS	6-20

7. SIMPLIFIED PROGRAMS ON ARCS

PROGRAMMING AN ARC ON THE RADIUS	7-2
PROGRAMMING AN ARC PASSING THROUGH THREE POINTS	7-7
PROGRAMMING AN ARC ON THE INITIAL TANGENT	7-9
PROGRAMMING CONNECTIONS.....	7-12
errors of geometry	7-18
ERRORS DISPLAYED WHEN EDITING.....	7-21
EXAMPLES OF PROGRAMS.....	7-23

8. PROGRAMMING ORIGINS

ORIGIN'S SELECTION	8-1
ORIGIN'S PROGRAMMING.....	8-5
EXAMPLES	8-10
REMARKS CONCERNING THE EXECUTION IN OPTION "D"	8-14

9. MILL RADIUS CORRECTION

PROGRAMMING MODES	9-1
MILL RADIUS CORRECTION ON CLOSED FIGURES	9-6
CONDITIONS OF GEOMETRIC ERROR.....	9-15
ERRORS DISPLAYED WHEN EDITING	9-19

10. CONTOURING

PERFORMING CONTOURING	10-1
AUTOMATIC RUNNING MODE	10-2
PROGRAMMED RUNNING MODE (function C)	10-4
EXAMPLES OF AUTOMATIC RUNNING	10-5
EXAMPLES OF USING FUNCTION "C"	10-14

11. PROGRAMMING PARAMETERS

PROGRAMMING MODES	11-1
EXAMPLES	11-3

12. SUBPROGRAMS

PROGRAMMING MODES	12-1
LAYOUTS OF SPECIAL FUNCTIONS	12-7
EXAMPLES	12-11
ERRORS DISPLAYED WHEN EDITING	12-20

IDENTIFICATION

The PTP400 is a modular system of numeric control with calculator. Specific utilization foreviewed for the system is in shop floors specialized in wood panels treatment for furniture industry.

The system is based on a multiprocessor structure, that means, on the use of two calculators working simultaneously : one "master" calculator, that allows the introduction of processing programs and their management, and a "slave" calculator running the processing sequences and controlling completely the machine.

Such a system guarantees flexibility and remarkable performances, obtained by exploiting the different features of both calculators: the former specialized in data treatment and processing, and the latter fast in execution and machine control.

PTP400'S HARDWARE STRUCTURE

Personal Computer Olivetti (model M250), IBM (or IBM-AT compatible) version :

- 640K Ram memory
- High resolution graphic screen (colour or monocromatic)
- Central unit with a 20 M capacity of program memory on hard disk.
- n. 1 floppy-disk unit of 720K or 1.44M
- Serial interface RS232 to communicate toward the machine control unit.
- Centronics interface to communicate with the printer.
- Optional boards and application program to communicate with another Personal Computer trough the local network.

CENTRAL UNIT'S HARDWARE STRUCTURE

19" rack suited for armoire.

The Hardware modules composing it are ;

- CPU "PTP200N"
- Axes expansion board "ESPAS"
- Input/Output expansion board "INOATR"
- Power feeder.

General configuration features

PTP200N

CPU board based on uP 8085AH1 at 10MHz with :

- 32K RAM memory
- 32K ROM memory
- Interface circuits with 2 motors DC
- 24 opto-coupled inputs
- 24 opto-coupled outputs with transistors 24Vcc/1A
- Interface RS232 for asynchronous communication toward calculator
- Interface for plug-in connection with expansion board.
- Power consumption : 1A su1 +5V , 100mA su ±12V.

ESPAS

Axes expansion board with plug-in connection with PTP200N.
with :

- Interface circuits with 3 motors DC
- Analogic output ±10V
- 4 opto-coupled inputs.
- Power consumption : 0.5A on the +5V , 100mA on ±12V.

INOCTR

Input/Output expansion board with :

- 24 opto-coupled inputs
- 24 RELAIS outputs 24Vcc/2A, 220Vac/10A
- Power consumption : 3A on the +5V (with every relais on)

POWER FEEDER

Drawer version insertable in a 19" rack

- + 5 10A
- ± 12 2A

Plug-in version .

- + 5 20A
- ± 12 2A

RACKS

19" racks drawer type, with power feeder, available in two types :

- Code 1004 with 4 positions free
- Code 1010 with 10 positions free

CONFIGURATION OF THE CENTRAL UNIT

The PTP400 central unit has at least the following configuration:

- n. 1 PTP200N board, with control on 2 axes (X and Y)

The maximum configuration is here listed :

- n.1 PTP200N board, with control on two axes (X and Y)
 - n.1 ESPAS board, with control on two axes (Z and W) and one analogic output
 - n.3 INOUTR boards
- in total 4 axes and 96 + 96 Inputs/Outputs.

The interface circuit with motors DC is constituted by the following elements :

- Analogic output $\pm 10V$ with offset and gain regulation trimmer , realized through a converter D/A at 12 bit.
- Opto-coupled inputs for encoder phases for frequencies till 50KHz per phase. (200 KHz processing frequency with multiplication *4) suited for encoder with square wave output feeded with 5 or 12 VDC (the encoder feeding power is supplied by the board : it can be selected by means of bridges at 5 or 12 Vdc /100 mA).
- Position counter with direction distinguisher. (Sampling frequency 5 Mhz).

On the PTP200N board are the interface circuits for two motors DC. The driving specifications are :

- Processing frequency max 150 KHz (corresponding to 90 mt/1' with resolution $\emptyset.01$ mm)
- Selectable acceleration from 1 to 9999 ms.
- Continuous control of speed and position.

On the PTP200N board can be added an axes expansion board with plug-in connection with the circuits for 3 more motors DC.

General configuration features

SUPPLY OF THE SYSTEM PTP400

The programs of the P.C. work in ambient MS-DOS, version 3.3 Olivetti or IBM.

The PTP400 software is supplied on disks of 3.5", suitable to work on a P.C with hard disk.

If the P.C. is not supplied with the software already installed by T.P.A., it is necessary to proceed to its installation.

The first operation, to be performed before beginning the installation of the PTP400 programs, consists in making a copy of the disks themselves, using the MS-DOS command "DISKCOPY", and store the original disks for possible recovering or further installations. The command "DISKCOPY" is available also on the PTP400 software disks.

INSTALLING THE SOFTWARE

The software is installed on the hard disk unit through automatic procedure. The hard disk should be prepared formatted, with the operative system version corresponding to the version of the installation disk.

About this procedure :

- If one of the following programs exists on the Root :

COMMAND.COM
CONFIG.SYS
AUTOEXEC.BAT

it will be renamed with the extension BAK.

- On the disk Root are copied the files :

COMMAND.COM interpreates the commands of the operative system MS-DOS on version 3.3 .
AUTOEXEC.BAT program on settings and startup.
CONFIG.SYS program on operative system settings .

-The software is installed on three directories called 400VB0, COMP1000 and UTILI.

These directories are created by the installation procedure.

The installation procedure is as follow :

- Switch on the PC with the INSTALLATION disk inserted in drive A
- Follow the "guide and request" messages being displayed during the installation.
- When the operation has ended, a message informs that the installation has been completed.

Once completed the installation, remove the disk from drive A and restart the computer (with the key RESET or switching it off and on) to start the PTP400.

PTP400'S CONFIGURATION SETTINGS

Program PTP400 is "configurable" on certain performances. This is realized by using a few general parameters, set down at operative system level of MS-DOS within the PTP400 startup programs (Programs with the "BAT" extension). We will define these particular parameters as " system variables".

The system variables concern :

1. Selection of the language (variable : LINGUA);
2. Selection of the measure unit (variable: UNIT);
3. Selection of the type of system XY (variable : TIPY).

To modify these variables is possible by exploiting the MS-DOS command "EDIT" , on file CUSTOM.BAT (read "file" as "program").

Set at MS-DOS command level, on directory PTP400, and digit :

```
EDIT CUSTOM.BAT <--'
```

The Video File Editor mode of the operative system will start, with the reading of the file CUSTOM.BAT.

SELECTING THE LANGUAGE

The PTP400 disk are supplied with the languages ITALIAN or ENGLISH as settings (per default). To change language, proceed as follows : Within the program CUSTOM.BAT, change the line

```
SET LINGUA=xxx
```

where xxx can assume the following values :

ITA (for ITALIAN)
ING (for ENGLISH)
TED (for GERMAN)
FRA (for FRENCH)
ESP (for SPANISH)
FLM (for FLEMISH)
DAN (for DANISH)

Example : if the desired language is German :

SET LINGUA=TED

To quit the Editor push at the same time the keys SHIFT and F5.
(equivalent to the Video File Editor command : EXIT AND SAVE).
Exit to DOS.

SELECTING THE MEASURE UNIT

Within the program CUSTOM.BAT change the line :

SET UNIT=x

where x can assume the following values :
M (to run the unit in [mm])
I (to run the unit in [inches])

Example : SET UNIT=I

Quit the Editor with SHIFT and F5.

SELECTING THE TYPE OF SYSTEM XY

In the program CUSTOM.BAT change the line :

SET TIPY=x

where x can assume the following values :
Ø (To run on system XY type Ø)
1 (To run on system XY type 1)

Example : SET TIPY=Ø

The systems XY are discussed in detail in the specific chapters

Quit the Editor with SHIFT and F5.

CONFIGURATION OF MACHINE CYCLES

The program FTP400 should be besides configured on the selection regarding the machine cycle.

It is defined "machine cycle" the whole of the execution modalities on a working program ; this concerns sides as :

- controlled axes (X and Y are obliged ; Axis Z is anyway free and axis W is free only if the machine has been mechanically preset on a movable fitting head)
- Definition on the cycle connected to each processing mode.
(single boring, fitting, horizontal, milling, grooving with blade)

The definition of the machine features - necessary in order to select the machine cycle among the ones configurables on the control - has to concern the preliminary stage of evaluation and definition of the complete system specifications, to be done in collaboration with the designers of the T.F.A.

CONFIGURATION OF LOCAL NETWORK

System PTP400 allows also the installation on local network (LAN: LOCAL AREA NETWORK), making this way possible direct program transfer, by means of the local network.

This allows to develop program and work plans in a office with a PC and then transfer it to the machine PC without need to resort to traditional transferring such as copy on diskette or communication trough serial channels.

For the LAN specifications of choice, installation, startup and utilization, please refer to direct contacts with T.P.A. and to the instruction manual supplied upon request.

STARTING THE SYSTEM

The system should be started according with the following sequence:

- Switch on the central unit
- Switch on the Personal Computer

The computer performs first a serie of inspections of the functions belonging to the Personal Computer (Internal tests.)

The calculator start thereafter automatically the inicialization procedure of the program PTP400.

The date is displayed, as known to the system, followed by an inquiry of introduction of a new date.

Current date is : 1/1/88
Please enter new date (day - month - year):

If the date as displayed is correct, confirm with <--'

To define a new date, enter it in the required form :

- Day - month - year ex:2-10-87 <--'
- Day - month / year ex:2/10/87 <--'

The time is displayed, as known to the system, followed by the inquiry of introduction of a new time .

Current time is 9:54:27,60
Enter new time :

If the time as displayed is correct, confirm with <--'

To define a new time, enter it in the required form :

- Hour:minutes ex:9:07 <--'

Wait for the display of the initial grafic page,then press any key to access the MAIN MENU.

TURNING THE SYSTEM OFF

To turn off the system, follow this sequence :

- Switch off the Personal Computer
- Switch off the power to the central unit .

It is very important to avoid turning the system off when the PC is performing operations such as writing on and/or reading from a disk. (signalled through the drives or the hard disk light)

NOTES ABOUT USING DISKS

Despite disks are generally safe, the possibilities of damage will be even less if a few simple advices are followed :

- NEVER FOLD THE DISKS
- AVOID TOUCHING THEIR MAGNETIC SURFACE
- KEEP THE DISKS IN THEIR ENVELOPE AND STORE THEM IN THEIR BOX WHEN NOT USED.
- NEVER BRING THE DISKETTES NEAR TO WARM OR MAGNETIC SOURCES.
- KEEP DUST AWAY FROM THE DRIVE BY KEEPING THE DRIVES CLOSED.

DISKETTE LABELLING

Each box of diskettes contains a number of self-adhesive labels :
It is good practice to write upon them the required informations before sticking them to the diskette.

NUMBERING THE INPUT/OUTPUT PORTS

All the I/O devices are divided in groups of 8 and each group is named PORT.

Therefore, every port includes 8 Outputs and 8 Inputs and each of them refers to a BIT.

The number of BIT goes from 0 to 7 and shows the position occupied in the port by the Input or the Output.

In the programs, in fact, the I/O should be indicated as BIT + PORT. The numbering (addressing) of the ports goes from 000 to 063, with a partition of the ports in two groups, one for the Input and the other for the Output :

- 32 ports of Input (000÷031)
- 32 ports of Output (032÷063)

The following pages report the documentation concerning the I/O. The word FLAGS means that if there are no INOUTR expansion boards installed, the ports involved can be employed as flags within the programs.

NOTE : A flag can be addressed both as Input and Output, independently from its position (address)

General configuration features

CUSTOMER				MACHINE						
INPUT CONFIGURATION				SHEET						
PTP 200N BOARD										
7	6	5	4	3	2	1	Ø	CODE		
								Ø	Ø	Ø
								Ø	Ø	1
								Ø	Ø	2
ESPAS BOARD										
7	6	5	4					CODE		
								Ø	Ø	3
INOUTR BOARD OR INTERNAL FLAGS										
7	6	5	4					CODE		
								Ø	Ø	4
								Ø	Ø	5
								Ø	Ø	6
INOUTR BOARD OR INTERNAL FLAGS										
7	6	5	4					CODE		
								Ø	Ø	7
								Ø	Ø	8
								Ø	Ø	9
INOUTR BOARD OR INTERNAL FLAGS										
7	6	5	4					CODE		
								Ø	1	Ø
								Ø	1	1
								Ø	1	2

General configuration features

CUSTOMER								MACHINE			
INPUT CONFIGURATION						SHEET					
INTERNAL FLAGS											
7	6	5	4	3	2	1	0	CODE			
								0	1	3	
								0	1	4	
								0	1	5	
								0	1	6	
								0	1	7	
								0	1	8	
								0	1	9	
								0	2	0	
								0	2	1	
								0	2	2	
								0	2	3	
								0	2	4	
								0	2	5	
								0	2	6	
								0	2	7	
								0	2	8	
								0	2	9	
								0	3	0	
								0	3	1	

General configuration features

CUSTOMER								MACHINE			
OUTPUT CONFIGURATION						SHEET					
FTP 200N BOARD											
7	6	5	4	3	2	1	Ø	CODE			
								Ø	3	2	
								Ø	3	3	
								Ø	3	4	
INOUTR BOARD OR INTERNAL FLAGS											
7	6	5	4	3	2	1	Ø	CODE			
								Ø	3	5	
								Ø	3	6	
								Ø	3	7	
INOUTR BOARD OR INTERNAL FLAGS											
7	6	5	4	3	2	1	Ø	CODE			
								Ø	3	8	
								Ø	3	9	
								Ø	4	Ø	
INOUTR BOARD OR INTERNAL FLAGS											
7	6	5	4	3	2	1	Ø	CODE			
								Ø	4	1	
								Ø	4	2	
								Ø	4	3	

General configuration features

CUSTOMER								MACHINE			
OUTPUT CONFIGURATION						SHEET					
INTERNAL FLAGS											
7	6	5	4	3	2	1	Ø	CODE			
								Ø	4	4	
								Ø	4	5	
								Ø	4	6	
								Ø	4	7	
								Ø	4	8	
								Ø	4	9	
								Ø	5	Ø	
								Ø	5	1	
								Ø	5	2	
								Ø	5	3	
								Ø	5	4	
								Ø	5	5	
								Ø	5	6	
								Ø	5	7	
								Ø	5	8	
								Ø	5	9	
								Ø	6	Ø	
								Ø	6	1	
								Ø	6	2	
								Ø	6	3	

PTP400'S MAIN FEATURES

It is here listed a resume of the FTP 400's main features: clearly this list is intended only as a first introduction.

- Continuous control and positioning on three Cartesian axes (X,Y,Z) with logic point to point.
- Optional control over a fourth axis (W) to handle double fitting heads.
- Milling functions on linear and circular interpolation on planes XY, XZ, YZ ; linear and helicoidal on space XYZ.
- Automatic acceleration/ deceleration in the positioning point-to-point and start/end interpolations.
- Programmable tangential velocity of interpolation .
- Automatic contouring in mill processings.
- Able to run up to 99 tools with possibility of using more than a tool at the same time.
- All machine functions are settable.
(tool correctors, work planes, linearity correctors, mills radii correctors
- Automatic running of the fitting on three different modes : with fixed head (variable pitch),with independent mandrels (variable pitch) and double reference point fitting (on axis W).
- Possibility to run grooving blades both along axis X and along axis Y.
- Length of the programs until 400 executive blocks.
- Possibility of entering a program list in sequential execution (processing list or article) with setting of the number of repetitions on the whole list and on each program.
- Possibility of normal or specular processing, on 4 independent processing areas.
- Configuration of the control with measure units in [mm] or in [inches], with possibility of entering programs simultaneously in both measure units.

General configuration features

- Programmable quotes :
from ± 0.0001 mm to ± 9999.9999 mm
from ± 0.00001 inch to ± 999.99999 inches.
- Processing speed programmable in (m/min) or (inches/sec).
- Unlimited number of processing programs, identified through a alphanumerical code of 8 characters.
- Unlimited number of processing lists (Articles), identified through a alphanumerical code of 8 characters.
- Possibility of EDIT while running programs execution.
- Print and grafic display of the programs.
- Simplified programming to execute ovals, connections, circumferenc arcs.
- Possibility of display in seven different languages.
- Possibility of introduction in a local network, to communicate with another/s Personal Computer.
- Possibility of installation of program CAD- T.P.A., to ease introduction of unquoted processing contours.
- Possibility of customization by integrating an applicative program of control user definition.
- Possibility of composition and print of labels defined on the bar code system.
- Possibility of composition, on the automatic processing list, of programs through the bar code reader.

MAIN MENU

After the initialization, the main menu is displayed.
It concerns the operations allowed by the control.
The list displayed is shown below :

```
MAIN MENU
E Programs editing
Z Functions editing
G Machine control mode
P Parameter mode
F Format program disk
R Build-up directory
B Program backup
S System's auxiliary
D Debug's auxiliary
O User identification
T Data and time setting
L Barcode mode
V Set-up PTP400
U User's auxiliary\CAD TPA
M MS-DOS video file editor
H Run DOS command
X Restart PTP400
Q Exit to MS-DOS
```

On the right side of the screen, in its lower part, are displayed the date and the time as known to the system.

The highlight word in the first line of the menu indicates the selection of the " Program edit " mode.

To select within the machine menu :

- The keys "cursor up" / "cursor down" (in the numerical keyboard) allows to select the desired mode ;
- Other way, it is possible to select by typing the letter that refers to the required mode.

(For instance, to select "Parameter mode" press the key "P".)

Having selected, start working by pressing "ENTER".

NOTE : Often the key "ENTER" will be referred to with the graphic symbol <--'.

NOTE : The modes' selection letters, displayed in a different colour, remains the same independently from the language selected.

NOTE : If selecting with the keyboard, and the key CAPS LOCK has not been enabled, it will be displayed a message together with an acoustic signal (BEEP)

PROGRAMS' EDIT

Video Editor Program concerning the work programs in language PTP400, with possibility of grafical representation. The possibilities of EDITOR in the programs are examined in chapter 4 (VIDE EDITOR,DESCRIPTION OF PROGRAMMING LANGUAGE).

FUNCTIONS EDIT

Allows editing the function program named "FUNCT", and further indicated also with names of non-resident functions. The possibilities of EDITOR on non-resident functions are examined in chapter thirteen (EDIT OF NON-RESIDENT FUNCTIONS)

MACHINE CONTROL MODE

Control program concerning the execution of processing programs,of the Set-Point procedure, of the manual and diagnostic modes, and of the monitor functions .

The possibilities of "automatic" (handling of processing and Set-Point programs) and "manual", are discussed in the second chapter (AUTOMATIC,DIAGNOSIS AND MANUAL DISPLACEMENT)

The monitor's functions are discussed in appendix D (MONITOR'S FUNCTIONS).

PARAMETERS

Program of formulation of machine settings : configuration of machine tools,parameters of fixed values, parameters related to the axes, machine emergencies. Its possibilities are examined in chapter three (MACHINE PARAMETERS).

FORMATTING PROGRAM DISK

Formatting mode acting on a disk in drive A, used to backup a installation disk or a processing program/articles. This mode is examined in chapter four (FORMAT PROGRAM DISKS)

BUILD-UP DIRECTORIES

Inizialization mode of the programs/articles directories in order to use them to store programs and articles. This mode will be examined in chapter four (MAKE PROGRAMS DIRECTORIES)

PROGRAMS' BACKUP

Mode to file and/or recover programs and/or processing articles. This mode is discussed in chapter four (PROGRAMS' BACKUP).

SISTEM'S AUXILIARY

Mode to insert resident functions in the program.

This possibility is examined in chapter thirteen (SYSTEM'S AUXILIARY MENU).

DEBUG'S AUXILIARIES

Auxiliary modes available to check the machine's execution modes.

This possibility is examined in chapter fourteen (DEBUG'S AUXILIARY PROGRAMS).

USER IDENTIFICATION

Memorizes the machine operator identification.

When selected, it requires the insertion of the operator's name (identifier) with further memorization. This information can be useful when used to plan the work with the machine.

ENTER DATE AND TIME

Allows modification of date and time directly from the PTP400 menu. The entering of date and time is completely guided with simplification of the data insertion by distinguishing between day, month and year (in the date) and between hours and minutes (in the hour).

Wrong insertions are filtered.

BARCODE MODE

Allows definition of labels defined with the bar code method, with possibility of printing the labels (on the corresponding codes).

This possibility will be examined in chapter fifteen (MAKE LABELS)

SET-UP OF PTP400

When this mode is selected, it proposes a list of the main settings employed by the System PTP400, similar to the one shown in the next page.

General configuration features

version of PTP400

Date of last recording	<mm-dd-yyyy><hh:mm:ss>
Version S/W	8.0 (dd/mm/yy)
Version F/W	PTPV3_0 (dd/mm/yy)
Language	ING
System XY	0
Measure unit	M
Customer	customer's name
System driver	C:
Virtual memory driver	
Program driver	C:
Articles driver	C:
System's path	C:\400V80\
Program's path	PROGRAM\
Articles' path	ARTICOLI\
System's auxiliary path	C:\COMP1000\
Call user's auxiliary (Y/N)	Y
User auxiliary Path	C:\TABLET\
User auxiliary Name	M400

M modify , <--' exit

The informations above listed are taken in part from system variables, and in part from a few PTP400's configuration programs. It is opportune to set in evidence that most informations reported in the list, and they will be fully described later, refer mainly to features belonging to the Personal Computer's system + PTP400's application programs and, as such, they constitute subject of primary interest to the personnel busy with system installation, updating and intervention.

The system end user can ignore those extremely specialized sides, getting to know them only to pursue personal interest or in the eventuality of an express demand of specialization.

The last insertion date refers to the last variation demanded to the configuration programs.

The usefulness of that parameter is essentially of listing the machine supply.

Versions S/W and F/W indicate respectively the name and the date connected to program version PTP400 on Personal Computer (S/W) and to the board PTP200 (F/W).

Language, system XY and Measure Unit mean respectively :

- The language set for all displays.
- The selection choosed for the system of axes XY
- The linear measure unit selected.

For these three settings, the selections are run on the same number of machine variables. (File : CUSTOM.BAT)

General configuration features

"Customer" indicates a machine's identification field, customerizable by the end user.

For instance, it can indicate the end user's name.

To modify the parameter (setting) "customer", digit :

M <--' , as displayed on the screen.

Then enter the parameter (setting) as required and quit this mode by inquiring to "STORE" :

S <--' , as displayed on the screen.

System driver means the disk unit employed as System PTP400 unit : it is indicated the hard disk, addressed to as C .

Virtual memory driver displayed without any assignement means that this unit is not actually being used (other way it is displayed as virtual disk or RAM disk) :

an assignment to RAM disk could, on the other hand, prevent the correct running of some PTP400's programs, due to unsufficient room on the RAM disk.

Programs' and articles' driver means respectively the disk unit employed to run the programs and the user articles ; in both cases is displayed the hard disk, addressed to as C.

Follow the paths identification on hard disk, respectively of :

- PTP400's programs (sistem path), displayed as complete path :
C:\400V80\ ;
- Working programs (programs' path), displayed related to the system path : PROGRAM\ . The full path shall be :
C:\400V80\PROGRAM\ ;
- Working articles (Articles' path), displayed related to the system path : ATICLES\ . The full path shall be :
C:400V80\ARTICOLI\ ;

System's auxiliary path indicates the path relative to the mode called "System's Auxiliary" (see : MAIN MENU)

Follow three items related to:

- A generical application of user configuration (see USER'S AUXILIARY) ;
- An application supplied by T.F.A. to run CAD (same as above)

This identification take place when the program (path) is located on hard disk, when the program itself is named, and when the mode itself is enabled to operate. (Mode "User's auxiliary")

USER'S AUXILIARY

This function, if enabled, allows the execution of :

- A generical program to configure the user of the PTP400 programs. For a description of the interface modalities please refer to the specifications supplied upon request.
- An application supplied by T.P.A. to run CAD with grafic board or with universal drafting devices, auxiliary to define generical processing contours.

For a complete description of the program CAD-T.P.A., please refer to the user's manual.

MS-DOS VIDEO FILE EDITOR

This function allows to start the DOS operative system in the program Video File Editor. The operator is required to type the name of the program to be read by the program Editor.

The program Video File Editor of the operative system can be runned from several PTP400's menus, and it is supplied together with the PTP400's program as a facility to modify the configuration data available to the control user.

Please refer to the operative system manual for a full explanation about the use of EDIT.

RUN MS-DOS COMMANDS

When this mode is selected, the program PTP400 is interrupted, and it is possible to run a command belonging to the operative system. However, we wish to inform the operator that it is just an interruption of the program PTP400, and that both the program PTP400 and the command DOS exist in memory.

This does not assure the possibility of running any command DOS, as only a part of the PC memory results available.

For commands of general utility this is anyway assured.

Some PTP400's updating programs, runned in DOS, are available.

Refer to chapter thirteen for a detailed explanation. (System's auxiliary programs)

RESTART PTP400

Allows resetting the PTP400, with re-display of the first page.

This reset does not imply re-execution of the operative system variables' setting programs (example : program CUSTOM.BAT), but it concerns solely the interface procedures with the machine control's board PTP200 (re-executes the full transmission of machine parameters (settings) and resident function)

_____ General configuration features _____

EXIT TO MS-DOS

Return to operative system level MS-DOS.
To run again the program FTP400, digit :

WORK: <--'

MACHINE CONTROL MODE

On the upper part of the screen, are displayed all the machine's operative functions.

1:AUTO

2:START

3:END

4:SETP.

5:MAN.

6:EXIT

On the lower part of the screen, on the left, is displayed the flashing message "PROCESSING IN PROGRESS".

At the first selection of the MACHINE CONTROL MODE (machine handling), by the system startup or by resetting the PTP400, this flashing message indicates the begin of the PTP200 board's initialization procedure, that imply passing through the following stages :

- 1- Operative check of the communication on serial line RS232 ;
- 2- Transmission of the required machine settings, as selected in Parameters :
 - identification of qualified axes
 - parameters of all qualified axes
 - list of performable emergencies
 - Linearity correctors ;
- 3- Transmission of "resident" functions (if existing)

If stage 1 imply a wrong result, it will appear the message

ERROR IN BOARD INIZIALIZATION RETRY (Y/N) ?

- With N (no) the message "BOARD NOT CONNECTED " indicates the impossibility of running the Set-Point operativity, the execution of programs and of performing manual displacements.
- With Y (Yes) the connection with the board is retried.
If a state of error is remarked in the connection with the board, check the serial connection with the board PTP200N.

Only the positive result of stage 1 allows to step to the next stages 2 and 3.

When stage three is reached the flashing message "PROCESSING IN PROGRESS" disappears, and a serie of messages is displayed on screen requiring confirmations upon the functions transmitted to the board (Files from which they are read, identification number of the transmitted functions, total number of transmitted functions).

To define writing modalities and criterias for functions identification and organization of the board PTP200N, please refer to paragraph " AUXILIARY PROGRAMS " in chapter thirteen.

Once the message "PROCESSING IN PROGRESS" has disappeared, it is possible to select among the modes belonging to "MACHINE CONTROL MODE" by pressing the corresponding function key.

Example : To access AUTOMATIC, press key F1 .

F1 - AUTO Allows compilation and starting of the program list to be run.

F2 - START Start on SET POINT procedure

F3 - END Allows an interruption of the SET-POINT procedure or programs' automatic execution.
An execution interrupted this way cannot be started again from the point of interruption, but it has to be started all over again.

F4 -SETP Pre-sets the machine to the execution of the initialization procedure (named Set-point procedure). This execution is started by pressing the function key "START"

F5 - MAN Selects operativity "Diagnostic and manual movements"

F6 - EXIT Goes back to the machine main menu.

NOTE

Upon the first selection of the mode "MACHINE HANDLING" starts, as seen, the initialization of the board PTP200N with transmission of parameters and resident functions.

Among the resident functions is the function "parallel task" (see chapter "AUXILIARY PROGRAMS") to define an integrated P.L.C. program (Programmable Logic Control).

When the stage of functions transmission to the board is completed, the execution on function "parallel task" starts automatically.

DETAILED DESCRIPTION OF MACHINE PERFORMANCES

1:AUTO

AUTOMATIC

By pressing key F1 it is possible to type down the program list to be executed.

The display on the screen is as follow :

1:AUTO	2:START	3:END	4:SETP.	5:MAN.	6:EXIT
--------	---------	-------	---------	--------	--------

REPETITIONS > 999 < ARTICLE > < EXECUTE EXIT
--

The cursor, indicated with the symbol "█", is located at the beginning of the area for entering the programs list, defined upon 17 editable lines .

The area to enter the programs list is not, usually, displayed empty but showing the last list entered in the FTP400.

The machine is pre-set to run the program list in two different ways : the right mode is enabled automatically, according with the length of the list itself.

Up to a number of 4 programs (list defined with 4 lines) the control allows the execution of CONTEMPORARY EDIT, contemporary to program execution; this allows interventions such as writing or correction of a program, or of a list of programs (articles), while the machine runs.

In case of a list with more than 4 programs, the control turns off automatically the CONTEMPORARY EDIT facility.

Each line of the execution list should be defined with the syntax :

nn * (name) (type of execution) (option of execution)

where nn is the number of repetitions connected to the program (not obliged). The maximum number is 255.

* divides nn from the program's name (obliged if the field "nn" appears)

name program's name (obliged)

execution mode defines the processing field on plane XY and the execution mode (obliged).
It can be defined with one or more characters :
with one character it defines a "long" area
with 2 or more characters it defines a "short" area and it has the second character defined with "/" .
The types of executions allowed are defined later.

option if missing, the type of execution is entirely defined upon the previous setting.
If present, it assume a meaning when defined on character "D" and it allows contemporary processing on two references. (Processing in option D).

Example of composing a list :

```
3*P0001N
2*P0002N/
P0003SD
```

After having typed the first line of the list, go to the next line by pushing '<--'.
To reach a point in the list, use the cursor keys "up" and "down" of the numerical keyboard.



Aside is shown the key (TABULATOR), which allows moving along the 5 positions of ENTER activated upon selection of AUTOMATIC.

The active selection is set in evidence in reverse. The fields selectionables are :

- list of programs
- list's repetition factor.
- Article
- Run (the inserted list)
- Quit (AUTOMATIC)

"ESC"

By pressing the key ESC, the editable area on which the cursor is positioned is set on zero : it can delete a line from the list of programs, re-write an article's code, and so on.

The fields affected by this key are :

- List of programs
- List's repetition factor.
- Article
- Search (articles or programs directory)
- Comment upon articles

"HOME"

When pressed, this key deletes the whole list of programs typed.

This key is active when entering a list (with the cursor located within the central frame on the screen) .

"END"

When selected, with the cursor in any position, it relocate the cursor on the area QUIT and then, by pushing '<--', reset the previous screen.

REPETITIONS > <

Defines the number of repetitions required to the program list typed. It is initialized at 999.

ARTICLE > <

Allows to store on disk a list of programs named ARTICLE, to read one previously stored or to display the directory of the programs or articles available.

To store the program list displayed within the frame in the center of the screen, type the name to be connected to the list itself followed by " / ", and then push <--'.

Example:

ARTICLE >ART001/ < Stores the list called ART001

If the article to be recorded exists already on disk, the question

ARTICLE ALREADY EXISTING ON DISK, ERASE OLD ARTICLE (Y/N) ?

If the answer is Y, the new list take the place of the old one.
If the answer is N, the new list will not be recorded .

When an article has to be stored, the entering of a comment upon the article is required. Max. lenght of the comment 30 characters.

To load an article from disk, type the name connected to it when it has been stored, without " / " , and press <--'.

example :

ARTICLE >ART002 < Reads and displays the list stored as
ART002

To display the directory, type " / " and press <--' : a page with the headline "SEARCH" will appear .

The possibilities of "SEARCH" are here listed :

- List all programs or articles beginning with certain specified characters ;
- List the entire directory of the programs and articles.

Steps of "SEARCH" :

SEARCH : Without any parameter , displays the program directory (press key <--').

SEARCH : /P The letter P indicates the program directory.
The result is the same as the previous example.

SEARCH : PR Displays all programs existing in the directory
 (or PR/P) which name begins with the specified string, in
 this case the string "PR".
 (ex. PR001 PROG1 PRG006 etc.)

SEARCH : /A The letter A indicates the Article directory :
 it is displayed the whole article directory.

SEARCH : AR/A All the articles existing on disk which name
 begins with the specified string are displayed.
 (ex. ART001 ART005 ART1 etc.)

The directory displays the programs/article 's names, the comment
and the date of last intervention.

The directory is displayed on a scrolling list, with a maximum
of 17 names per page.

To get back to the previous menu, press the key END in the
numerical keyboard.

RUN (execute)

After having typed or recalled the required program list, to start
its execution it is necessary to set the cursor on the area RUN
with the key TABULATOR, and then press <--'.

The computer checks first the syntax and that the program exists
on disk and displays, on the right side of every line, the
description, the size and measure unit connected to the program
required on the line.

When a confirmation of the selected performance is required, press
again <--' or, in case of accidental confirmation, use the
TABULATOR key to get back on the program list for an eventual
modification or to change operative field (QUIT, ARTICLE ,....).

After confirmation, every program of the list is read from disk,
checked, translated into a language intelligible to the control,
(board PTP200), transmitted through the serial line and performed.
The stage of program translation into a language intelligible to
the control is also called compilation stage .

As already anticipated, the machine is able to work in two different ways :

1. PROGRAMS' LIST EXECUTION MODE "OFF LINE "

This mode runs lists up to 4 programs.

Example : 3*P01N
 2*P02N
 P03S

With ENTER, all programs of the list are in sequence read from disk, checked, listed, and sent to the board .

When the list is exhausted, on the entire list is run the execution.

The disappearing of the flashing words "ELABORATION IN COURSE " means the end of the elaboration and transmission stage.

The operator could, for instance, quit the mode " MACHINE HANDLING" (with QUIT : function key F6) to access the program's EDIT, or type a new program list (in MACHINE HANDLING) while the machine keeps on running till the end of the programs' list already in execution.

A new programs' list cannot, anyway, be run if the control is already on execution.

When such a request is entered, appears the message :

"PROGRAM IN EXECUTION"

and the command " RUN " on the new list is ignored.

To end earlier the execution in course, press the key "END" (function key F3)

There exists anyway an exception to the "OFF LINE" execution, with a list defined upon less than five programs:

it is the case in which the programs set in execution require an amount of memory exceeding the memory available on the board PTP200N. In such a case, the execution "IN LINE" is automatically started, as normally started with a list of more than four programs.

In this case, besides, the flashing words "ELABORATION IN COURSE " do not disappear.

2. PROGRAMS LIST' EXECUTION MODE "IN LINE"

This mode is selected with a list of more than 4 programs, or less than 5 programs in the cases above considered.

When entered, the programs' list is read, checked, compiled, sent to the board and the programs are performed one by one.

Example :

```
3*P01N
2*P02N
P03S
5*P04T
10*P05N
```

Each program of the list is read, compiled, transmitted and run independently and in sequence, upon the list entered.

The line in a different color tells which program is actually running.

A character "█" on the right side of the outer frame indicates the line of the list in elaboration : on the lower part of the screen is updated the number of program line in current elaboration.

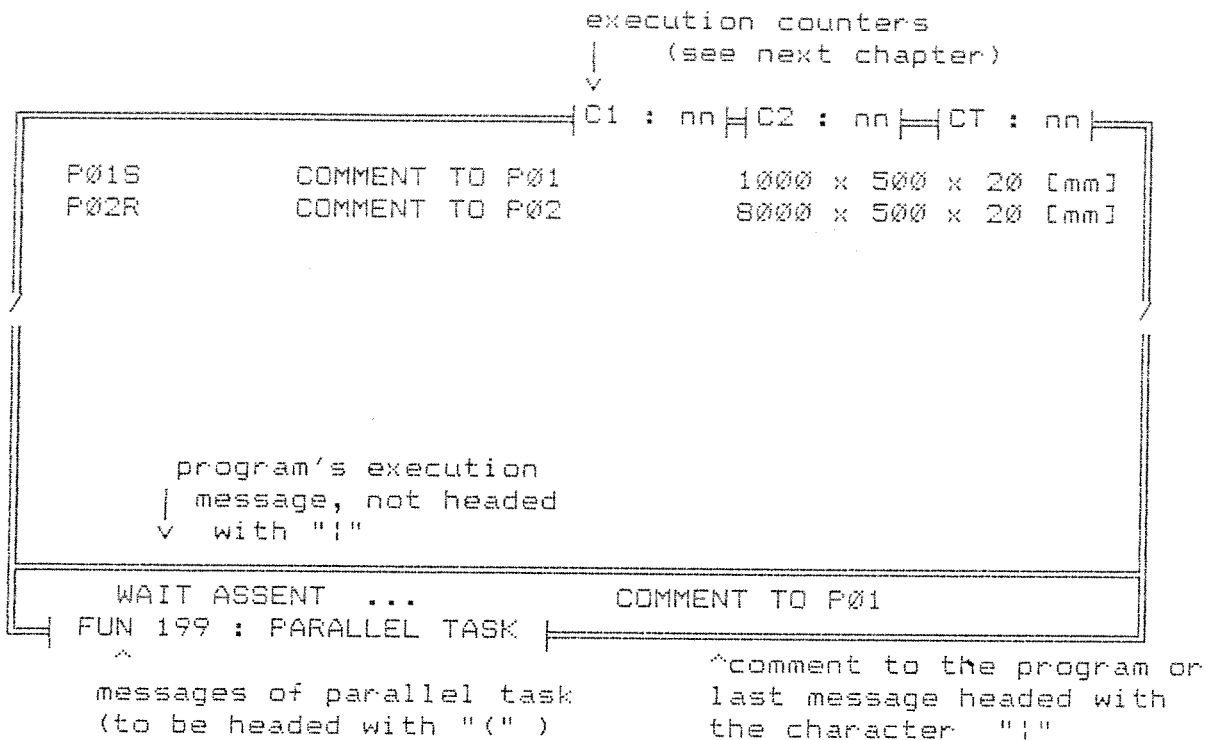
The execution stage can be ended with the key END (function key F3). During programs' list execution no other program can be run.

EXIT

Move to "EXIT" with the tabulator key or with the key END of the numerical keyboard, then press <--' to run again the commands displayed in the upper part of the screen (Function keys F1 to F6)

DISPLAYS DURING PROGRAM EXECUTION

During program execution, few places of the screen are employed to display messages :



Upon program's automatic execution (execution of processing cycle or of set-point) every message heading the character "!" is displayed in the second last line of the screen, on the right side. When performing a processing cycle, the comment to the program is automatically run upon a message, heading the character "!" and displayed in the program's first instruction. Usually, the message heading the character "!" is inscribed in the start-cycle functions (see chapter "AUXILIARY PROGRAMS") so that the kind of processing in course is kept displayed.

(Example of message : PRG (N) running)

The messages not headed with "!" are displayed on the same line, on the left side.

On the last line of the screen is displayed a "parallel task" message :

- The "parallel task" is anyway on execution, independently from running a program, therefore this field is always active ;
- All messages on the parallel task MUST head the character "(", other way they will be displayed on the second last line of the screen, deleting possible program signals .

3:END END

Interrupts the execution of the program list entered or the SET-POINT procedure.

4:SETP. SET-POINT

2:START START

Set-point defines the machine initialization procedure; in the modalities required by the electromechanic typology of the machine itself.

Generally speaking, the initial setting of the machine has to include :

- Setting upon the output to the machine.
- Checking upon the state of the inputs.
- Reset on zero of the position of every controlled axes, through automatic research of the position zero point.

The Set-point execution is necessary to start the machine (before every program execution) and after intervention of an emergency related to axes control (servo error, uncorrect connection with the encoder) ,or after a machine emergency to reset the machine mode.

To run the Set-point,press key F4 (SETP.) ,in order to select the function, followed by key F2 (START).

Once Set-point has been selected, press key F3 (END) if the selection made has to be deleted.

5:MAN. MANUAL AND DIAGNOSTIC OPERATIVITIES

Selection on function key F5 (MAN.) starts the manual mode. Please, refer to related chapter.

6:EXIT QUIT

Goes back to the main menu page.

HANDLING THE EXECUTION COUNTERS

When running programs/lists, counters of the number of executions performed are in function. There can be two different cases, according with the number of programs in execution.

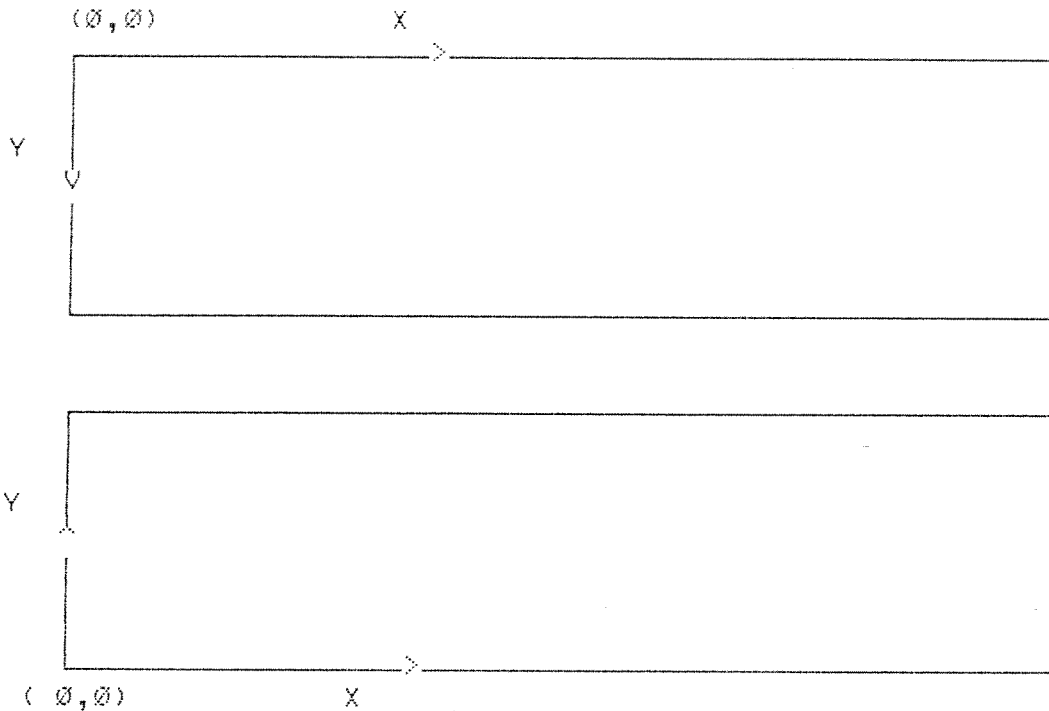
Up to a number of 4 programs (list defined up to 4 lines) and only if execution "off line" results enabled : every single program has a counter (C1, C2, C3, C4), and one is for the entire programs' list specified. (CT)
All counters run scaling down :

- a. C1 (C2, C3, C4) is initialized to the repetition factor connected to program number 1 (2, 3, 4) of the list and it decreases of one unit when each execution is started.
- b. CT is initialized to the repetition factor connected to a program list (field : REPETITIONS) and it decreases at each execution on the list itself.

With a number of programs above 4 (list defined on more than 4 lines) :
only the counter CT is in function, with the same modalities above described.

SYSTEM OF REFERENCE XY

The processing plane is defined upon two axes : X and Y.
It is possible to configure the FTP400 on one of the two systems of reference, as represented :



The two systems differ for the orientation of the axis Y and, therefore, for the starting point position of the axes X and Y. In the first case, the starting point is set on the rear of the machine. It is the most used configuration. In the second case, the starting point is set on the front of the machine.

On the chapters of this manual dedicated to the definition of the processing programs no reference is made about the type of system XY qualified : it has to be assumed as good the first system defined (with starting point on the rear of the machine)

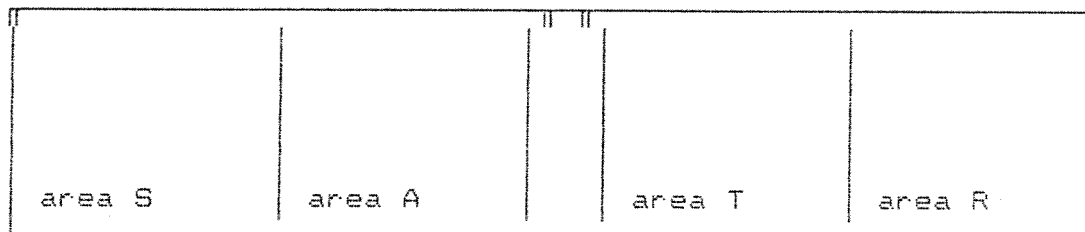
For the differences between the two configurations of the system XY, please refer to Appendix C.

DEFINITION OF WORK AREAS

The modalities of the programs' automatic execution allow to run different processing areas (or fields), with the double benefit of optimizing the processing time and of executing mirrored panels without having to rewrite the program.

The most complete utilization of the machine foreviews four different processing areas, without limitations in the possibilities of combination on contiguous areas.

The four work areas are defined in the following graphic diagram :



The outer frame represent the work plane XY.

The four symbols " π " represent the supporting points managed (called in this manual reference points or ref.points), to verify the panel's points of support. Beginning with the first reference point on the left, they are identified as :

- Ref.point of processings N and S
- Ref.point of processing A
- Ref.point of processing T
- Ref.points of processings M and R.

The meaning of these processings is reported later, on this same section.

Each area has been assigned one fourth of the work plane XY.

Areas S and T are areas of "normal" processing :

- The execution of a program is the same as the program's pattern, with the axes' origins on the left side of the machine, and values X of processing increasing toward the right.

Areas R and A are areas of "mirrored" processing :

- The execution of a program reflects the program's pattern mirrored on axis X. The axes' origins, under execution, is reversed on the right side of the machine, with the values of X increasing toward the left.

EXECUTION MODES N and N/

Execution N (N long) is employed to process panels as long as the whole processing plane : Also panel fastening takes place on the plane XY.

It is no possible any fastening contemporaneous on the plane.

Execution N/ (N short) is employed to process panels with lenght within the three areas S, A, T.

Panel fastening take place in these three areas : it is therefore possible contemporaneous fastening on the sole area R.

In both types of execution, the panel is leaned against reference point N and the processing is normal.

EXECUTION MODE S and S/

Execution S (S long) is employed to process panels with lenght not beyond the left side half-plane (areas S and A).

Panel fastening takes place on the entire left side half-plane.

A contemporaneous fastening on the right side half-plane is possible, on a specified area or on the whole half-plane (that means, $R+T=R$ long ; other way $T+R=T$ long ; or : single area R or T specified as short areas)

Execution S/ (S short) is employed to process panels with lenght within the area S.

Panel fastening takes place only on the area S :

therefore it is possible a fastening on the three remaining areas A, T, or R, area by area or on a possible combination of contiguous areas (that means, $R+T+A=M$ short ; $R+T=R$ long ; $T+R=T$ long ; or single area A, T or R specified as short areas.)

In both types of execution, the panel is leaned against the ref.point N and the processing is normal.

EXECUTION MODES T and T/

Execution T (T long) is employed to process panels with lenght not beyond the right side half-plane (areas T and R).

Panel fastening takes place on the whole right side half-plane.

It is possible a contemporaneous fastening on the left side half-plane on a specified area or on the whole half-plane (that means : $S+A$ long ; $A+S$ long ; or single area A or S specified as short area) .

Execution mode T/ (T short) is employed to process panels with lenght within area T. Panel fastening takes place on the sole area T : it is therefore possible a fastening on the three remaining areas (S, A, R), on a specified area or on a possible combination of contiguous areas (that means : $S+A=S$ long ; $A+S=A$ long ; or : single area S, A, R specified as short area) .

In both executions the panel is leaned against ref.point T and the processing is normal.

EXECUTION MODE M and M/

Execution M (M long) is employed to process panels long as the whole work plane : also panel fastening takes place on the whole plane XY.

Contemporaneous fastening on the plane is not possible.

Execution M/ (M short) is employed to process panels with length within the three areas R, T, A.

Panel fastening takes place on these three areas : It is possible contemporaneous fastening on the sole area S.

In both executions, the panel is leaned against ref.point M and the processing is mirrored.

EXECUTION MODES R and R/

Execution R (R long) is employed to process panels with length not above the right side half-plane (areas T and R).

Panel fastening takes place on the whole right side half-plane.

It is possible a contemporaneous fastening on the left side half-plane, on the whole half-plane or on a specified area (that means, S+A=S long or single areas S or A specified as short areas)

Execution R (R short) is employed to process panels with length within area R.

Panel fastening takes place on the sole area R : It is possible therefore a contemporaneous fastening on the three remaining areas (T, A, S) on a specified area or on a possible combination of contiguous areas (that is : S+A+T=N short ; S+A= S long ; or : single area T, A, S specified as short areas).

In both cases the panel is leaned against ref.point M and the processing is mirrored.

EXECUTION MODES A and A/

Execution A (A long) is employed to process panels with length not above the left side half-plane (areas A and S).

Panel fastening takes place on the whole left side half-plane.

It is possible a contemporaneous fastening on the right side half-plane (that is : R+T=R long ; T+R=T long ; or : single area T or R specified as short areas) .

Execution A/ (A short) is employed to process panels with length within area A.

Panel fastening takes place on the sole area A and, therefore, it is possible a contemporaneous fastening on the three remaining areas (S, T, R), on a specified area or on a possible combination of contiguous areas (that means: R+T=R long ; T+R= T long ; single area S, T, R specified as short areas)

IN both cases the panel is leaned against ref.point A and the processing is mirrored.

REMARK :

To exploit the plane and the related work areas as just depicted, imply a deep acknowledgment of the FTP400's processing features. Moreover it should be added to that, a skilled utilization of programs' control and of the informations supplied concerning it. Basilar, among the most important sides, are :

- The definition modalities of the machine's cables layout ...
(To an effective handling of 4 independent work areas should correspond cables' layouts of the control signals to the machine equally independent, for each work area);
- The layout of the functions performing panel loading and unloading;
- The compilation of the processing lists.

It remains possible anyway to employ the machine for lighter duties, such as handling only two independent areas (defined upon two work half-planes).

For a more complete explanation of the definition modalities of the control's handling specifications, please refer to chapters "System's auxiliary programs" and "Description of machine's cycles".

Here are examined few examples of processings defined on the same number of processing lists :

Example 1 :

```
P01N  
2*P02M
```

The list is defined on two programs, with execution modes :

- N long , on program P01
- M long , on program P02.

Besides, on program P02 has been specified a multiplying factor of 2.

First, the fastening on area N long is enabled. Therefore :

- The panel is leaned against ref.point N
- The fastenings are enabled on the whole work plane.

It is no possible any contemporary fastening on the work plane.

Once fastening on area N long took place, program P01 in normal processing starts running.

When the panel is set free in area N long, fastening on area M long is enabled, and therefore :

- The panel is leaned against ref.point M
- The fastenings are enabled on the whole work plane.

It is no possible any contemporary fastening on the work plane.

Once the fastening on area M long took place, program P02 of mirrored processing starts running.

When program execution has ended and the panel is set free on area M long, the fastening on area M long is enabled again : Infact the multiplication factor specified in program P02 is 2.

When the panel is set free again after the second run of program P02 in area M long :

- If the list's repetition factor is 1, the execution ends.
- If the list's repetition factor is more than 1, the execution starts again from the beginning of the list, till exhaustion of the repetitions required when the list was compiled.

Example 2 :

```
P01S  
P01T  
P01A  
P01R
```

The list is defined upon 4 programs, with the following execution modes :

- S long
- T long
- A long
- R long

The program to be run is P01 on every line of the list ;
The list's repetition factor is 10.

Program P01 is runned alternatively 10 times, on each of the four processings specified : The sequence of execution reflects the list's compilation sequence.

About the areas' occupation sequence on the work plane:

It is always possible, during processing of a panel (on a half-plane), to fasten the panel next in processing, as that panel is always positioned on the free half-plane .

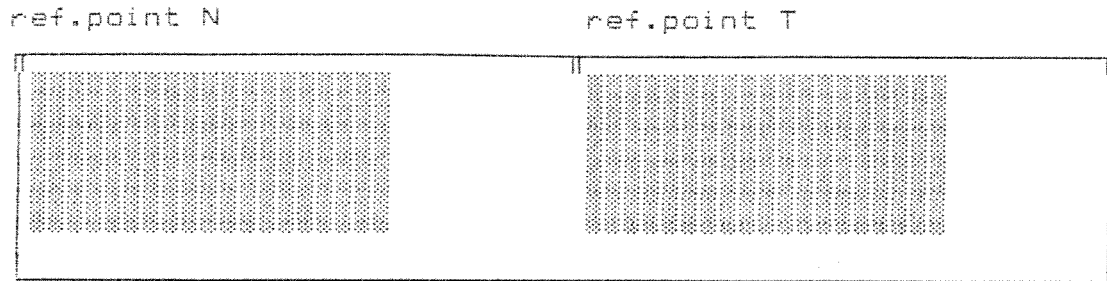
Starting the execution of the list, the fastenings of each of the two half-planes are enabled in execution modes S (ref.point N) and T (ref.point T).

The execution starts first with program P01 mode S, when the fastening has been carried out.

The operator can fasten the panel on the right side half-plane, while the machine process the panel in area S.

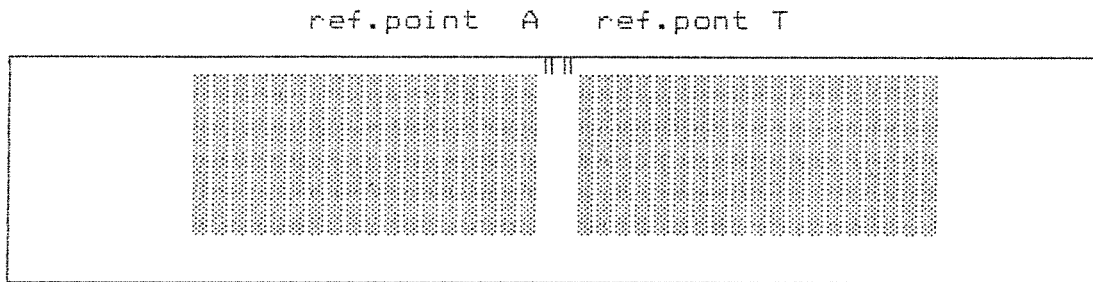
When the panel is set free in area N long, execution of the program in field T can be immediately started (If the fastening has been carried out) : while processing the panel in field T, the operator can fasten the next panel in field A (left side half-plane, ref.point A), and so on till exhaustion of the list, upon the 10 preset repetitions.

The possible dynamic situation of machine employment can be represented this way :



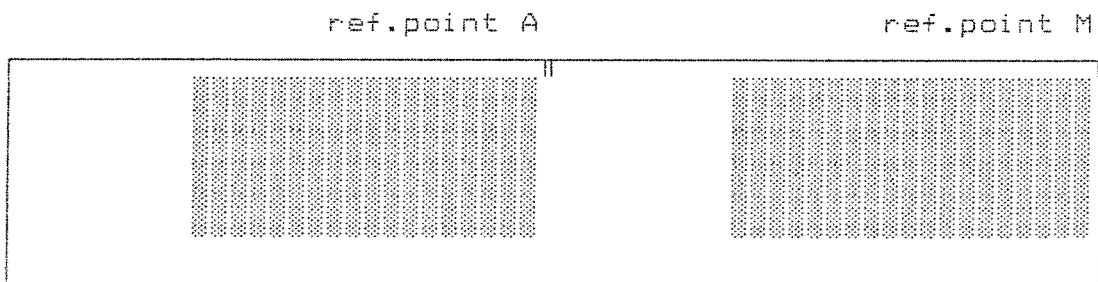
↓
panel in processing stage
in mode "S".

↳ panel fastened, ready for
processing in mode "T".



↓
panel fastened, ready for
processing in mode "A" .

↳ panel in processing stage
in mode "T".

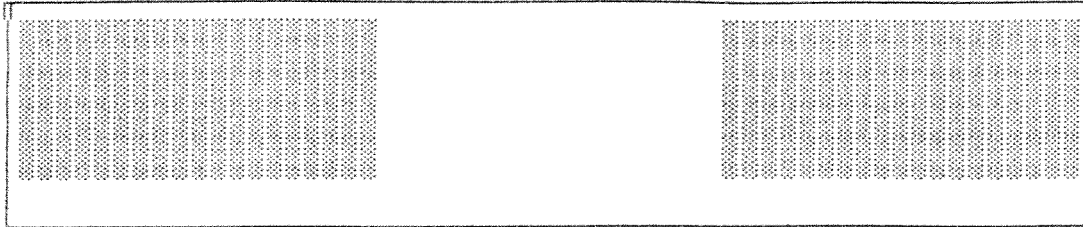


↓
panel in processing
stage, in mode "A" .

↳ panel fastened, ready
for processing in mode "R".

ref.point N

ref.point M



↓
panel fastened, ready
for processing in mode "S"

└> panel in processing
stage, in mode "R".

EXECUTION MODALITIES IN OPTIONAL MODE "D"

As anticipated when introducing AUTO (Automatic), every panel can be called again in execution with the optional mode "D" concerning the type of processing.

This option allows to process a panel on the two references left and right at the same time. This execution is possible if the program specified has been written using the Origin's functions. (function : 0 ; see related chapter)

Lets suppose to have to process the panel PROG1 with execution mode "N" having specified option "D".

When typing the automatic running list, insert :

PROG1ND
with repetition factor opportunely suited.

Program PROG1 runs in area N for every operation programmed on starting points 0 and 1, in area M for every operation programmed on starting points 2 and 3.

The change of sector, and therefore of reference point, means the manual displacement of the panel to the required area and ref.point.

The order programmed concerning panel processing remains the same, with the eventuality that more changes of sector can be necessary if the operations were not divided in two groups when programmed.

When option "D" is selected to process a panel, the control sets off some fastening procedure in processing the next panel, or better, with the panel processed contemporarily to the one in option "D".

This happens independently from the execution mode specified.

MANUAL MODE AND DIAGNOSTIC

The mode to run manually the machine is accessed by selecting the function "FS-MAN" in the menu " Machine control mode".

The following page is displayed :

DEDUG AND MANUAL MODE

AXES

```

X
Y
Z
W
screw's interval [mm] :
speed [mt/1'] : 3
pitch [mm] : 0.1

```

INPUT / OUTPUT

```

output :
input :

status :

```

AXIS DISPLACEMENT

```

-                +
      AXIS X

```

ANALOGIC OUTPUT

```

value : 0

status : OFF

```

MESSAGES :

```

keys : X/Y/Z/W select axis
dacU : INS enable DEL disable

JOG
1      2      3      4      5      6      7      8      9      10
JOG/ST dacU  AXIS  VEL   PITCH  IOCODE  SET    RES    INP    EXIT

```

The page proposed is divided in squares, each having its specific function.

In the square "AXES" are displayed :

- The names of the qualified axes
- The values related to the axes
- The values of: manual displacements velocity, set pitch, manual displacement (mode STEP) and the interval of axis sampling to run the linearity correctors. These last parameters are reported for the sole selected axis : pointed out by the words in reverse on the axis' name (on the first page it is always selected axis X, as indicated) .

In the square "AXES" the operator can set :

- Axis' selection
- velocity
- pitch

The square "AXES DISPLACEMENT " displays :

- The name of the selected axis (therefore qualified to manual displacement.
- When performing the selected axes' manual displacement, an horizontal arrow points out the direction of axis' displacement : in positive direction when the pointer is located next to the character "+", in negative direction when the pointer is located next to the character "-" .

The square "INPUT/OUTPUT" displays the fields :

- output, to select an output signal
- input , to select an input signal
- status, to display the selected input signal mode.

The square "ANALOGIC OUTPUT" displays the fields :

- value , for the setting of the ESPAS board's analogic output.
- mode , to display the analogic output mode

The FUNCTION KEYS are programmed to run the functions :

F1 - JOG-ST	F6 - IOCODE
F2 - dacU	F7 - SET
F3 - AXIS	F8 - RES
F4 - VEL	F9 - INF
F5 - PITCH	F10 - EXIT

FUNCTION KEYS TO HANDLE AXES' DISPLACEMENT

F 1

JOG/STEP

Sets the selected axis' type of displacement.

The choice of the type of displacement remains single, for all axes run by the control.

The words "JOG" or "STEP" immediately above the F1 function key wording shows the choice made.

Press F1 to select among the two possible options.

The displacement with JOG starts when the cursor key is pressed and ends when the key is released.

The displacement velocity is the one specified in the square "AXES" and, except for modification entered by the operator, it is assumed to be equal to the manual velocity specified in "PARAMETERS" (for each axis) .

With STEP, the axis displacement is equal to the pitch set in the square "AXES".

By keeping pressed the cursor key, the axis moves more STEPS consecutively.

The mode active per default is JOG (as shown by the word above area F1).

Upon axis' displacement with JOG, if the control is in mode " Set-point performed " with inquiry of axis' displacement in "-", it is set as maximum reachable value the axis' negative limit (set in PARAMETERS); in "+" the maximum reachable value is the axis' positive limit .

If the control is in mode "Set-point not performed" no fixed limit runs automatically.

The selected axis' displacement takes place by pressing the keys "right arrow" or "left arrow" on the numerical keyboard :

- Pressing the key "right arrow" the axis moves in positive direction.
- Pressing the key "left arrow" the axis moves in negative direction.

F 3

AXIS

This function is employed to select one of the qualified axes. The modality of selection /de-selection of a axis is entirely led through messages.

Upon selection of F3 : a pointer (arrow) (->) indicates the axis selected (within the square "AXES") :

- With the keys "pointer up" and "pointer down" of the numerical keyboard, the pointer moves to the previous axis or to the next;
- <--' (ENTER) exit the axes selection function and the square "AXES" is updated with the parameters related to the selected axis (velocity, pitch, interval of linearity correction) .

The selection of an axis can be performed also from the keyboard without selecting F3, as indicated with the message :

"Keys : X/Y/Z/W/ to select axis"

For instance, press the key Y to select axis Y ; the square "AXES" is immediately updated, on the parameters related to the axis selected.

F 4

VEL

It is employed to set the selected axis' displacement velocity, in relation with the displacements handled with JOG or STEP.

It is not allowed to set a velocity above the max. allowed velocity (parametric).

The velocity is set in :

- [mt/min] in case of measure unit control in [mm]
- [inch/sec] in case of measure unit control in [inches].

F 5

PITCH

Allows to set axis' displacement for displacements in STEP.

On every qualified axis, the pitch is initialized at 0.1 mm.

The pitch runs in :

- [mm] in case of measure unit control in [mm]
- [inches] in case of measure unit control in [inch].

INPUT/OUTPUT FUNCTION KEYS

F 6

IOCODE

It is employed to select a OUTPUT and/or an INPUT (square "INPUT/OUTPUT") on which to run the functions of SET, RES, INP. The input/output selection mode foreviews setting of :

- number of the BIT (a number : from 0 to 7)
- number of the PORT (three numbers : from 000 to 031, on input : from 032 to 063, on output).

Example : 3032 <--'
 7000 <--'

F 7

SET

It qualifies the last selected output line (on F6).

F 8

RES

It disables the last selected output line (on F6).

F 9

INP

It displays the last selected input mode (on F6). The input mode is inscribed in square "INPUT/OUTPUT", next to the word "mode".

- ON means that the input is ON
- OFF means that the input is OFF

F 2

DACU

It allows to enter the setting value of the ESPAS board's analogic output (indicated as : output U) .

The value is entered in the square "ANALOGIC OUTPUT", by the word "value" ; the setting concern tenths of milliVolts, with values ranging from -1000 to +1000.

In correspondance with the setting values, the analogic voltage U is run upon values ranging from -10V to +10V :

- Upon selection of key "INS" (on the numerical keyboard), the value inscribed is entered as analogic output U setting.

The word "ON" in field "status" of the correspondent square means this mode.

- Upon selection of key "DEL" (on the numerical keyboard), value 0 is entered as analogic output U setting.

The word OFF in field "status" of the correspondent square means this mode.

F10

QUIT

Return to the menu "Machine control mode".

PARAMETERS

To access the machine parameters select "PARAMETERS" in the main menu.

"PARAMETERS" displays, in shape of lists (cycles), all numerical informations needed to run the machine.

PARAMETERS MENU

The accessible cycles are listed in the two following lists :

- └─> Number of head's tools
- └─> Head's configuration
- └─> Mills radii
- └─> Axis X tools correctors
- └─> Axis Y tools correctors
- └─> Axis Z tools correctors
- └─> Parameters of fitting tools and on axis W
- └─> Ref.points on work fields (N M T A)
- └─> Air values and offsets
- └─> Depth of horizontal holes
- └─> Processing speed [mt/1']
- └─> Mandrels' rotation speed

- └─> Selected axes
- └─> Parameters of axis X
- └─> Parameters of axis Y
- └─> Parameters of axis Z
- └─> Parameters of axis W
- └─> Input test for manual
- └─> Head's features
- └─> Definition of horizontals
- └─> Axes' limits
- └─> Parameters of linearity correctors
- └─> Linearity correctors on axis X
- └─> Linearity correctors on axis Y
- └─> Linearity correctors on axis W
- └─> Contouring parameters of axis X
- └─> Contouring parameters of axis Y
- └─> Contouring parameters of axis Z
- └─> Contouring parameters of axis W
- └─> Machine emergencies

The distinction in two different lists of parameters cycles reflects in two different modalities of access :

- The first menu is directly accessible.
- The second menu can be accessed only by entering a password.

In the upper corner of the screen, on the right side, is displayed the current page number and the total number of the pages availables on the menu running. (menu of cycle selection or of parameters setting.)

KEYBOARD AND FUNCTION KEYS

Here are examined the modalities of selection, modification and introduction of the cycles and/or parameters related. Moreover, here are examined specific functionalities managed with function keys.

keys (on numerical keyboard)

recall the previous page (PGUP) or display the next page (PGDN) of the menu running (menu of cycle selection or parameters setting).

keys (arrow up or down on numerical keyboard)

move the selection cursor one position up or down in the current menu.

keys (arrow left or right, on numerical keyboard)

display the first or last page of the current menu.

key (on numerical keyboard)

Sets the cursor on field "QUIT" of the current page. At this point, if '←' is pressed, it returns to the previous menu: if the first menu is running, it returns to the machine menu. In case of return to machine menu, the parameters are stored (with access to disk) only if at least one parameter has been entered (with key "INS").

key (on numerical keyboard)

allows to enter a parameter to replace the parameter displayed. Once entered it, confirm it with '←'.

key (function key)

modifies the selected parameter with the last value entered, only within a cycle (spreading). This function is very useful to set different parameters with the same value (typical of linearity correctors).

key F2 (function key)

prints the machine parameters :

- If the print is required in the menu of cycles selection, it involves every machine parameter (to print the parameters accessible through password, it must be entered) .
- If a cycle menu (parameters insertion) is to be printed, it involves only the cycle parameters (example: AXIS X TOOLS CORRECTORS) .

key F3 (function key)

allows to "convert" certain parameters ,limited to dimensional parameters of quotes and velocities, so to recover some written machine parameters when needed, like for instance, measure unit in [mm] for a work to be configured with measure unit in [inches]. To convert different measure units, perform the following steps :

- enter MI : qualifies the conversion function from [mm] to [inches]
- enter IM : qualifies the conversion function from [inches] to [mm]
- enter upon 0 (zero) : disables the conversion function (disables running of function keys F4 and F5)
- enter upon <--' : goes back to parameters menu.

On every parameters page, from machine menu, the conversion function is always disabled, in order to avoid unintentional data variation. When this function is qualified by the operator, on the right side of the screen is displayed the symbol of function key F3 followed by the qualified selection (MI, IM) .

However, qualification of key F# results possible only in the cycles selection menu.

keys F4 F5 (function keys)

These two function keys are qualified only if the conversion function (key F3) results qualified, and only in the parameters setting menu :

- Selection of F4 converts the required parameters according to the conversion rule :

(parameter) / 25.4 if "conversion" runs on MI
(parameter) * 25.4 if "conversion" runs on IM

- Selection of F5 converts the required parameter according to the conversion rule :

$$\frac{(\text{parameter}) * 1000}{60 * 25.4}$$
 if "conversion" runs on MI

$$\frac{(\text{parameter}) * 25.4 * 60}{1000}$$
 if "conversion" runs on IM

Key F4 should be selected on quotes (dimension) parameters ([mm] or [inches]).

Key F5 should be selected on velocity parameters : ([m/min] or [inch/sec]).

If a cycle of parameters is entirely run on homogeneous parameters (like all parameters of: velocity, quotes or numerical but without dimension, or not numerical parameters) the validity of the conversion is checked automatically. Example :

- If conversion of dimensions is selected (F4) to handle the cycle "machine emergencies" or "interpolation parameters", the conversion will not be performed (and the machine will BEEP).

If a cycle of parameters is instead run on not homogeneous parameters, like for instance "Axes' parameters", before running the required conversion the machine asks a confirmation to the operator.

key F 10 (function key)

Selection of function key F10 allows paging on a help menu. This menu displays a short guide of the selections availables in Parameters.

MEASURE UNIT

All possible parameters of the parameter cycles can be divided in three classes :

- 1- Non-dimensional parameters
- 2- Dimensional parameters with measure unit independent from control configuration.
- 3- Dimensional parameters with measure unit depending from control configuration.

For instance, the parameters belonging to class 1 are :

- number of tools
- parameters of head configuration
- parameters related to machine emergencies

Parameters belonging to the second class are :

- acceleration time (measure unit :[ms]).

Parameters belonging to class three are, for instance :

- axis X (Y/Z) tools correctors
 - mill radius
 - axes' limits
 - ref.point fields....
- processing speed
 - minimum and maximum speed per each axis .

The measure unit of "dimensional" parameters is in :

- [mm] if the control runs in [mm]
- [inches] if the control runs in [inches];

The measure unit of "velocity" parameters is in :

- [m/min] if the control runs in [mm]
- [inches/sec] if the control runs in [inches].

In the dimensional parameters are always specified the qualified measure units.

REMARKS WHEN DEALING WITH PARAMETERS

A first important side concerns the correctness of the parameters as entered.

To each cycle of parameters, it is connected :

- A qualification of the pattern, that can be "numerical" or "alphanumerical".
- A maximum dimension, given as max. number of characters on which the parameter can be defined.

Based on that, the Parameters operativity performs, on a value entered, the following check-ups :

- If the parameter is of the "numeric" type, is accepted only to enter the pattern $\pm xxx.xxx$, where :
 - . The sign + must be omitted
 - . The character "." (dot) should be employed to divide the whole numbers from the decimals.
 - . "X" is intended as a number from 0 to 9.
- If the parameter is of the "alphanumerical" type, are accepted both patterns with numbers and letters :
- Anyway the parameter cannot exceed the max.number of characters assigned to it.

No check-up is instead performed concerning the validity of the parameter as entered, under the functional point of view.

For instance, a velocity entered with negative sign keeps recorded that way.

Therefore, it is recommended to the operator to follow the instructions given in the present manual when dealing with parameters.

Only when the mode "Machine handling" starts, controls are performed concerning the correctness of some parameters of basic importance for the system, such as :

- qualified axes
- resolutions on the qualified axes
- max.speed on the qualified axes
- acceleration on the qualified axes

with consequent diagnostic signals if some parameter is not correct.

Here are listed the basic rules to be adopted while setting the machine parameters :

- Check always the correctness of every parameter, when starting a new machine.
- Axes X and Y have always to be qualified.
- The resolution on the qualified axes has to be entered with a positive value, not void.
- Every parameter concerning velocity of axes displacement has to be of positive value, not void.
- Every parameter of quote related to non-qualified axes (tools correctors, air quotes, offsets of processing etc.) has to be entered with void value. (= Ø).
- Acceleration values of the qualified axes have to be positive, not void.

A second important side concerns the definition of the stages of utilization of the machine parameters. It is useful to divide the parameters in different groups :

- GROUP 1
 - └─> Axes selected
 - └─> Parameters of axes X
 - └─> Parameters of axes Y
 - └─> Parameters of axes Z
 - └─> Parameters of axes W
 - └─> Parameters of linearity correctors
 - └─> Contouring parameters of axis X
 - └─> Contouring parameters of axis Y
 - └─> Contouring parameters of axis Z
 - └─> Contouring parameters of axis W
 - └─> Machine emergencies

- GROUP 2
 - └─> Linearity correctors on axes X
 - └─> Linearity correctors on axes Y
 - └─> Linearity correctors on axes W

- GROUP 3
 - └─> Number of head tools
 - └─> Head's configuration
 - └─> Mills radii
 - └─> Tools correctors on axes X, Y, Z
 - └─> Parameters of fitting tools and on axis W
 - └─> Work field's ref. points (N M T A)
 - └─> Air values and offsets
 - └─> Depth of horizontal holes
 - └─> Processing speed [mt/1']
 - └─> Mandrel's rotation velocity
 - └─> Input test for manual
 - └─> Head's features
 - └─> Definition of horizontals
 - └─> Axes' limits

The parameters belonging to group 1 should be entered when starting the system, before gaining access to the Machine handling mode : Infact, these parameters are transmitted to the PTP200N board after having been properly elaborated and the transmission takes place only once, at startup, at the first verification of the communication with the board.

All the parameters belonging to group 1 relate to data pages accessible upon introduction of password.

The parameters belonging to group 2 (linearity correctors) are, instead, updated on the PTP200 board about any change entered. Updating on board takes place always in "Machine handling "mode .

The parameters belonging to group 3 relate instead to the drilling section features and configuration, and as such, they interfere in the programs layout and running, but do not interfere with the PTP200 board functionalities.

NUMBER OF HEAD'S TOOLS

Enter the tools' maximum number, with values ranging from 1 to 99.

HEAD'S CONFIGURATION

Per every tool identify the head type.
The kinds recognized are :

TYPE A TOOL single bore

TYPE B TOOL double bore in X (pitch 32 mm)

TYPE C TOOL triple bore in X (pitch 32 mm)

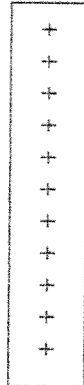
TYPE D TOOL fitting head with variable pitch
(enter in cycle "fitting tools" all
related parameters).

TYPE E TOOL 3 holes in Y (pitch 32 mm)

TYPE F TOOL 2 holes in Y (pitch 32 mm)

TYPE G TOOL 5 holes in Y (pitch 32 mm)

TYPE I TOOL 11 holes in Y (pitch 32 mm)



TYPE P TOOL Horizontal head in X
 on function G06



TYPE Q TOOL Horizontal head in X
 on function G07



TYPE R TOOL Horizontal head in Y
 on function G08



TYPE S TOOL Horizontal head in Y
 on function G09



TYPE J TOOL mill

TYPE L TOOL blade in X



TYPE M TOOL blade in Y



TYPE H TOOL measuring head (related to automatic
 learning procedure of panel thickness)

All unused tools should be classified as type Z tools.
Different kinds of tools can be classified as type Z tools

NOTE

Tools numbering in horizontal processing should follow a precise logic : type Q tool mirroring a P type tool is supposed to occupy the position next to type P tool. The same criteria should be adopted between tools of the types R and S.

About the borers processing the fourth axis (axis W) : they have to run on positions with a difference of 1Ø (in positive) in relation to the tools on axis Y.

If the fitting tool related to axis Y is of type D, the fitting tool related to axis W (fixed head) has to correspond to a number 1Ø positions different (in positive) from the position occupied by type D tool and it has to be entered as type A tool.

If the fitting tool related to axis Y runs with independent mandrels (type A tools), to each tool called on axis Y has to correspond a type A tool with 1Ø position difference (in positive) from the position occupied by type A tool specified for axis Y.

According with the utilization requirements of every machine, it is always possible to configure "feigned" tools, so to exploit at the best the configuration features of the head-tools.

A typical example goes over configuration of "feigned" tools to perform operations of fitting. If, for instance, type A tools can be used to perform fitting boring (besides, of course, single or anyhow combined borings) it is possible to configure one or two "feigned" type D tools so defined :

on the number of type A tools that are going to be used ;

on the pitch given to the different bits, which should result constant.

MILLS RADII

Specify the mill radius value, per each tool defined as type J tool in head's configuration.

Concerning the horizontal and vertical borings tools, it is possible to specify the diameter of the bit mounted, for a visual consultation in stage of EDIT. (see command "Ctrl +T" in the "Programs EDIT" mode).

The measure unit is in [mm] ([inches]).

TOOLS CORRECTORS ON AXES X, Y, Z

Per each tool define the corrector in X,Y,Z .

For every unused tool, (Z type tool) enter correctors value 0.

Correctors x and y related to a tool define the tool central axis distances (calculated along the related axes) from positions $x=0$; $y=0$, calculated in Set-point machine mode (value on axis $x=0$ and value on axis $y=0$).

Similarly, corrector z related to a tool defines the distance (calculated along z) of the tool tip from position $z=0$, calculated with the machine in Set-point condition on axis Z (value on axis $Z=0$).

Concerning the assignment of tools correctors'sign, please refer to the example about head representation described further.

Tools defined as types B, C, D (multiple holes heads in direction X) : The corrector in x should be reported to the head central point, along x.

Tools defined as types E, F, G, I (multiple boring heads in direction Y) : the corrector in y should be reported to the head central point, along y.

NB. Concerning the borers running the fourth axis (axis W) : the tools correctors on the mobile axis W should be run with the mobile head at the value of max. opening .

It is here proposed a tools configuration to help the definition of the tools correctors. The layout refers to the definition of tools correctors on axes X and Y.

The tools represented are 9, numbered from T1 to T9, and set in such a way to cover every possible case.

The character "*" means the tool's central axis.

It is represented the system XY, with starting point indicated by the point (*).

Consider the tool indicated as T1 :

- (a) indicates the corrector in x
- (b) indicates the corrector in y

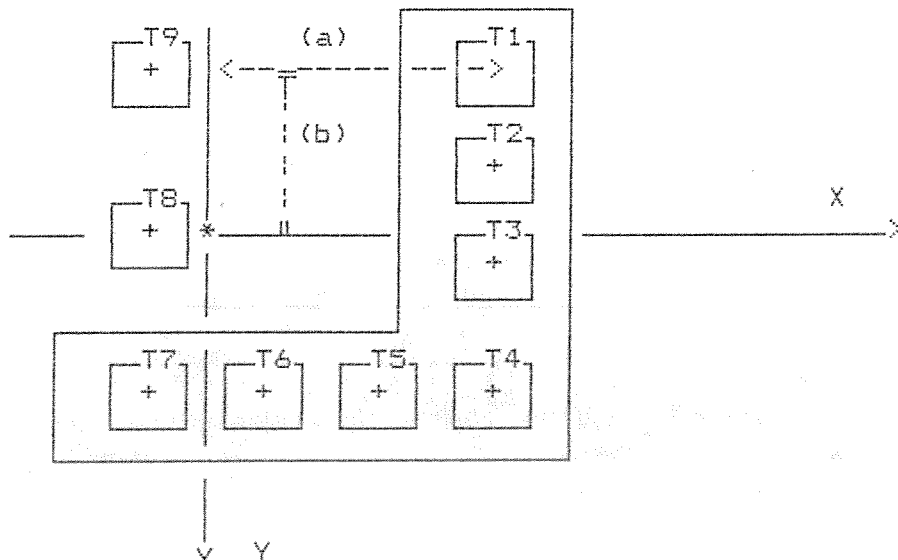
The sign on (a) is considered negative

The sign on (b) is considered positive

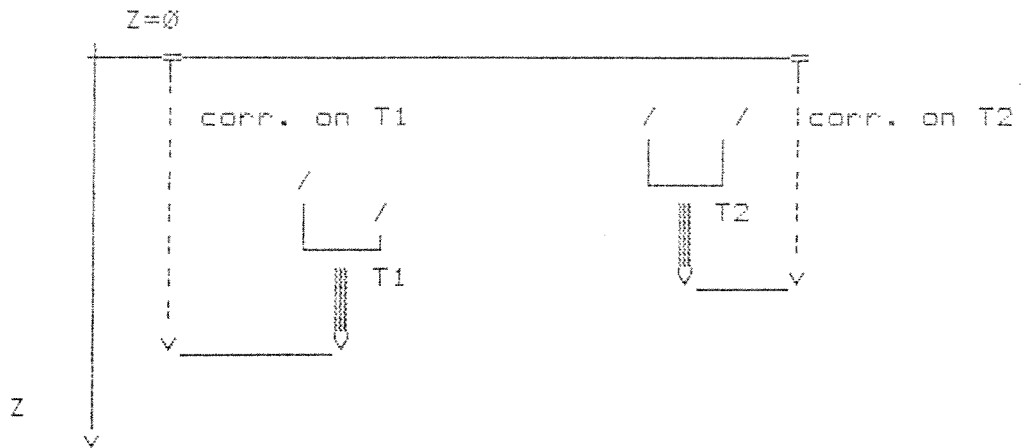
To determine the sign of a tool corrector, follow this rule : if the tool is on the positive values field (relatively to the axis qualified), the tool corrector assumes negative sign (as in case of T1 on axis X); the sign will result positive if the tool is on the negative values field (case of T1 on axis Y).

On the tools representation layout here reported :

- T1, T2 have corrector x negative, corrector y positive ;
- T3, T4, T5, T6 have correctors x and y both negative ;
- T7 has corrector x positive, corrector y negative ;
- T8 has corrector x positive, corrector y void ;
- T9 has correctors x and y both positive.



The following picture shows the definition of the tools correctors on axis z :



In the picture are represented two tools (T1 and T2) and on both of them the corrector in z results negative.

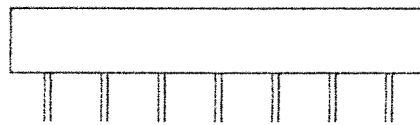
PARAMETERS OF FITTING TOOLS AND ON AXIS W

They specify for a number of 2 fitting tools (type D) the following informations :

- Tool's number : Tool's identification number ;
- Fitting pitch : Pitch ([mm] or [inches]) between the fitting points.(tops)
- Number of bits : number of the fitting head's bits .

Example of type D head :

number of bits : 7



pitch > <

To run independ mandrels fitting these layouts are not to be used: define in "heads configuration" the mandrels on type A. As alternative, define feigned fitting tools.

Moreover, two parameters are defined, significant for the machines arranged to run on axis W :

AXIS W TOOLS' HIGHEST EXTREME
AXIS W TOOLS' LOWEST EXTREME

respectively, these two parameters define the identification limits of the tools mounted on axis W. The identification limits are to be considered included.

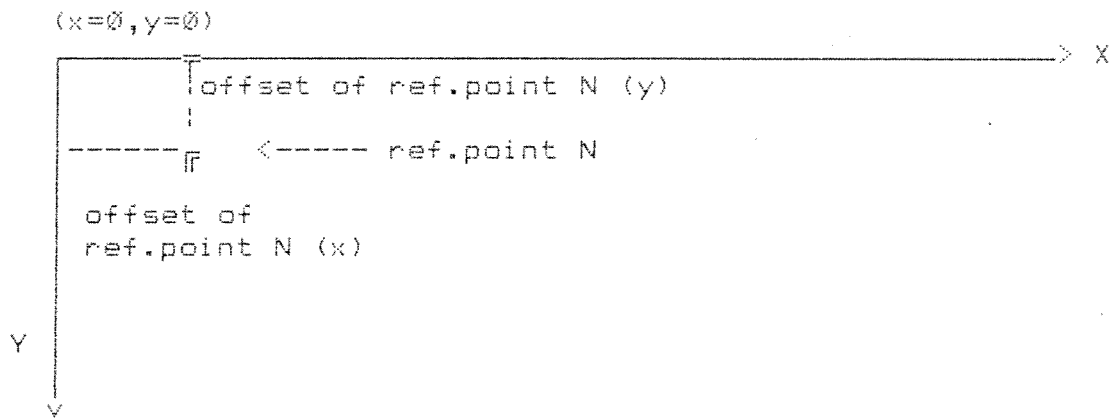
This identification allows the correct application of the correctors on the tools involved, minimizing the shiftings on axis W.

Configurations of tools installed on axis W not according with the mentioned limits do not produce a correct execution of the boring cycles.

WORK FIELDS' REFERENCE POINTS

Here are defined the position of the panels' four ref.points (N, M, T, A) on X and Y.

The picture shows an example of positioning on ref.point N, with indication of the two parameters of offset, in x and in y :

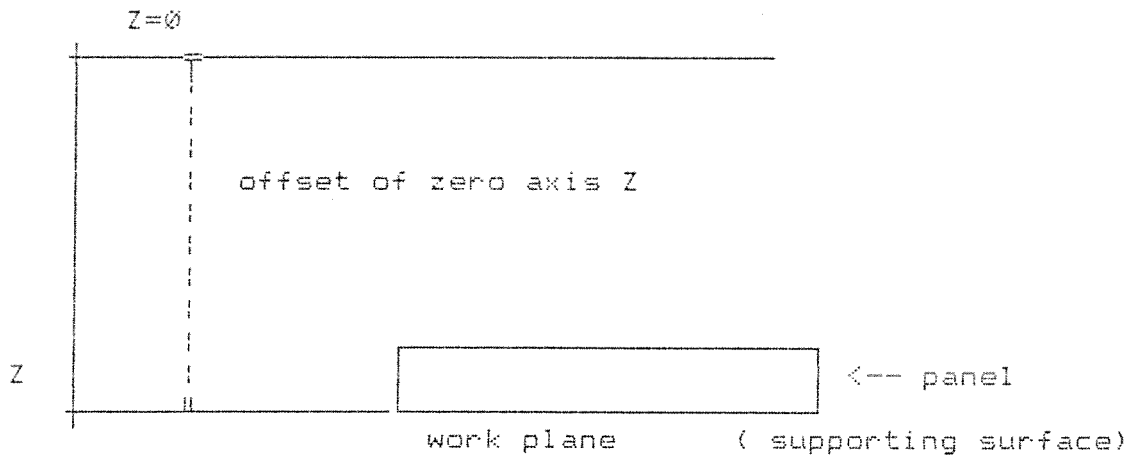


AIR VALUES AND OFFSETS

Here are defined the following parameters :

- OFFSET OF ZERO ON AXIS Z

enter the distance, along axis Z, from the Set-point position to the work plane :



- OFFSET Y ON AXIS W

(parameter significant when working on axis W)

enter the distance between the two rows of tools (one on axis Y and the other on axis W), with axis W on value zero (Set-point position). This is an offset in direction Y.

Below it is represented a mobile fitting head on axis W, and the correspondent fixed fitting head, in line with axis Y.

On axis W are marked the significant positions :

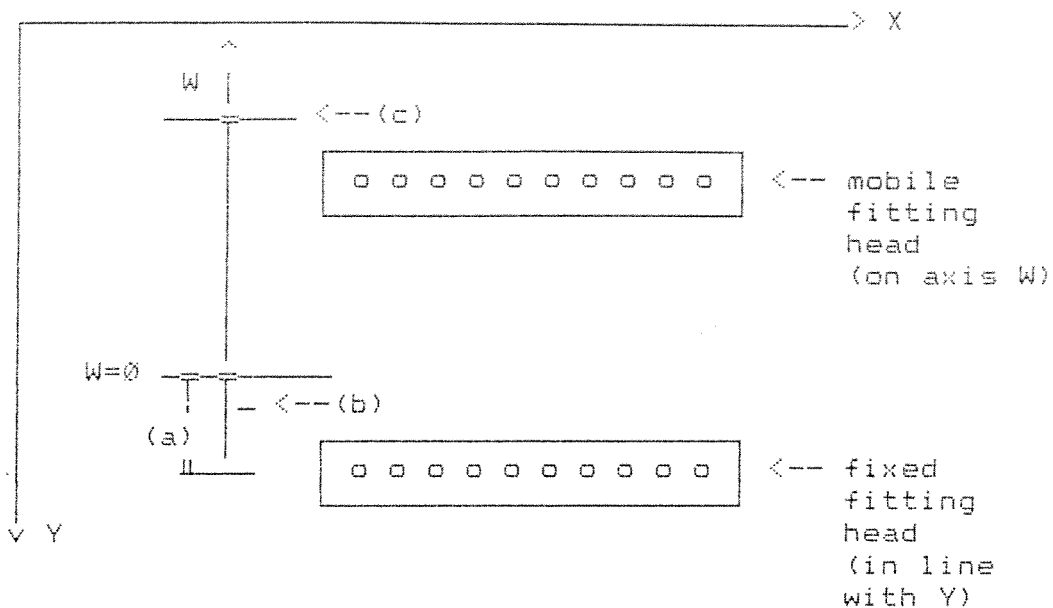
- Position of Set-point ($W=0$)

- Position of max. positive opening : expressed in (c).

The value corresponding to the position in (c) should be defined as limit of axis $W+$ (see further for the parameter definition)

- position of the limit of axis $W-$ (see further for the parameter definition) on the position in (b)

- parameter of "offset Y on axis W" : in (a)



- OFFSET Z ON "G93"

(parameter significant when programming function G93)
 enter the max. tolerated difference between the stated panel thickness and the real one.

- OFFSET Z ON VERTICAL AIRS

Parameter used in cases where the position of the ref.points bigger encumbrement (see parameter REF.POINTS' MAX. HEIGHT and the value considered as example in "Z air of ref. points encumbrement) results higher than the air position (see as example the value obtained as "Z air vr").

In this case :

- If the distance between the two positions exceed the parameter here entered, the two middle positions of the displacement of Z are managed. (air value and ref.points value)
- Other way, the sole position of ref.points' encumbrement will be managed.

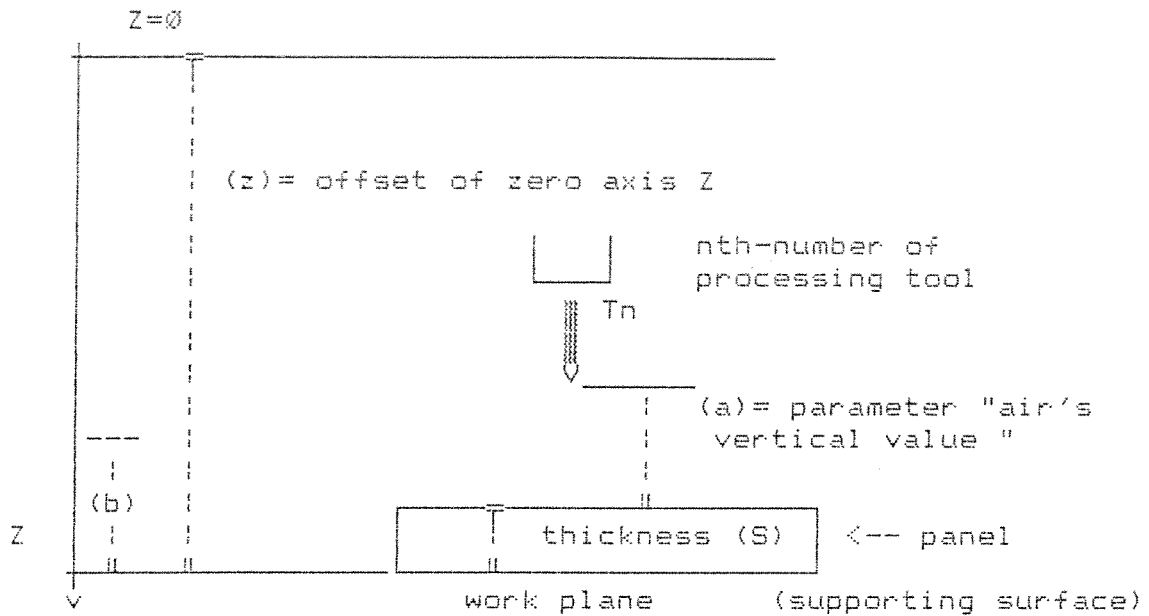
- AIR VERTICAL TOOLS

It defines the distance (along axis Z) at which the panel top is kept, upon consecutive fast displacements.

- REF.POINTS' MAX. HEIGHT

enter the distance from the work plane and the ref. point of bigger encumbrement (along axis Z).

Here are proposed the calculation criteria concerning the air parameters significant values :



- (a) = parameter "air's vertical value"
- (b) = parameter "ref.points' max. height"
- (z) = offset of zero axis Z
- (S) = panel's thickness
- (Tn) = tool working (with more than a tool working contemporaneously it is considered the lower tool)
- (cZn) = tool corrector Tn in z

The absolute processing values in Z (corresponding to positions in Z from the set-point position) are determined as from the following relations :

Z air vr = position with tool Tn in "air vertical value" from panel = $(z) - (S) + (cZn) - (a)$
 (this position depends from panel thickness and tools at work).

Z air on ref.points' encumbrment = air position reckoned on the encumbrment entered on the ref.points, = $(z) - (b) - (a)$.
 (this position depends from panel's thickness and the tools at work).

- AIR TOOL G06
- AIR TOOL G07
- AIR TOOL G08
- AIR TOOL G09

enter the distance between the boring bit and the panel, before performing the corresponding horizontal boring.

- MAX. PROCESSING VALUE OFF PANEL

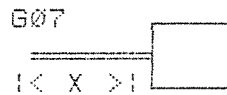
define the allowed processing extremes, out of the panel's area. The axes involved in this parameter are X and Y. The processings : milling and blade grooving.

- MAX. BLADE DEPTH

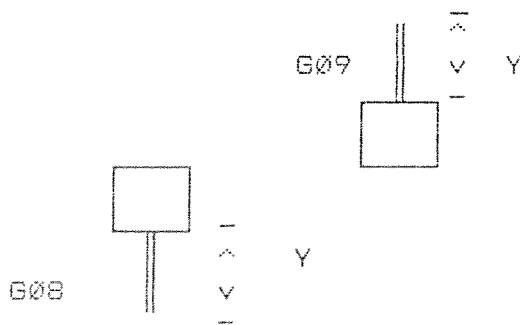
enter the max. processing depth admitted to cutters. (measure along axis Z)

DEPTH HORIZONTAL HOLES

- MAX. DEPTH OF THE HOLES IN X (processings G06, G07)
- MAX. DEPTH OF THE HOLES IN Y (processings G08, G09)



max. depth of the horizontal hole in X



max. depth of the horizontal hole in Y

PROCESSING SPEED EXPRESSED IN MT/1'

- INTERPOLATION SPEED [max 9 mt/1']

Express the max. interpolation speed to use in panels milling.
If in a program the interpolation speed is not expressed, it is assumed valid the one entered in "Parameters".
The interpolation speed express a value of tangential speed to the required trayectory, so it relates with the interpolating axes.

- BLADES PROCESSING SPEED (velocity on axes X ed Y)

Express the max. speed of blade movement in panels grooving.
If in a program this data is not entered, it is assumed as valid the value here entered.

- HORIZONTAL HOLES ENTRY VELOCITY (velocity on axes X,Y)

Express the max. entry velocity in the panel in horizontal borings. The stroke involved goes from the air value (as specified on utilized function Ggg) to the processing value.
If in a program this data is missing, is assumed as valid the value here entered.

- HORIZONTAL HOLES EXIT VELOCITY (velocity on axes X,Y)

Express the exit velocity from the panel in horizontal borings. The stroke involved goes from the processing value to the air value, as specified on the utilized function Ggg.
This value is not programmable.

- VERTICAL HOLES ENTRY VELOCITY (velocity on axis Z)

Express the max. entry velocity in the panel of : vertical borings (functions G00), fitting borings (function G20 and further G00) mill descent (function G10 of start mill profile) or bldes descent (functions G16 and G17).
If a program miss this data, it is assumed as valid the data here entered.

- THROUGH HOLES EXIT VELOCITY (velocity on axis Z)

Express the exit velocity from the panel when performing vertical boring (functions G00), fitting borings (function G20 and further G00).
The stroke involved goes from [work plane - 2 mm] to the final depth value. This velocity value does not have to be programmed.

- AGGREGATES' LOWERING VELOCITY (velocity on axis Z)

Express the lowering velocity from value zero to air value of the tools : Mill (type J), blades (type L and M). This value is not subjected to programming.

- AGGREGATES' RAISING VELOCITY (velocity on axis Z)

Express the raising velocity, from the value of end processing, of the tools : Mill (type J), blades (type L and M). This value is not subjected to programming.

- VELOCITY OF REFERENCE INTERPOLATION

Express the velocity of reference interpolation to determine the velocity of the connection inserted to correct a mill radius.

It is a velocity assumed from interpolation radii (read mills radii) of a 100 mm.

MANDRELS' ROTATION VELOCITY

(For the machines suited)

Enter the min./ max. velocity values on particular processings. For every operation are defined a value of minimum velocity and a value of max. velocity, independently from the number of mandrels utilized. The values on the parameters should be whole numbers, from 1 to 1000.

-MINIMUM VELOCITY OF VERTICAL BORING

Enter the min.velocity to employ in vertical boring operations.

-MAXIMUM VELOCITY OF VERTICAL BORING

Enter the max. velocity to employ in vertical boring operations. If missing, the data is taken from the one entered in "Parameters".

In the same way are defined the min./max. velocities of the followings :

- HORIZONTAL BORING
- GROOVING WITH BLADE
- MILLINGS

SELECTED AXES

Specify the axes handled on the borer.

The parameter should be assigned on 5 numbers, of value :
1 (qualified axis) or Ø (non-qualified axis) :

- 1' number from left : axis X
- 2' number from left : axis Y
- 3' number from left : axis Z
- 4' number from left : axis W
- 5' number from left : to enter always as Ø.

Example : 1110Ø select the 3 axes X , Y , Z
 11Ø1Ø select the 3 axes X , Y , W

Qualification on the two axes X, Y is obliged.

AXES' PARAMETERS

On each axis are run 6 parameters :

- MAX. SPEED : indicates the speed limit imposed by the system.
When a velocity is entered (Programs edit mode), it is checked that this velocity does not exceed the limit imposed by the manufacturer.
On axes X, Y, W the max. speed here specified is related to the displacements in air (In panels' automatic execution).
On axis Z, the max. axis velocity is related to the displacements in air on vertical / horizontal borings operations (in panels' automatic execution)
- ACCELERATION : expressed in milliseconds, indicates the time span in which the axis has to reach the appointed speed.
This parameter can be used also in axis deceleration.
- RESOLUTION : it is expressed in encoder pulses per linear measure unit ([mm] or [inches]), to define the axis resolution.

Example : with resolution Ø.05 mm, enter 20
 with resolution Ø.001 mm, enter 100.

NOTE : The interpolating axes should be defined all on the same resolution value.

- MANUAL SPEED : enter the velocity value for displacements in manual operativity. Enter a value at most equal to the max. axis velocity.
- MINIMUM SPEED: enter the velocity value below which the PTP200N board does not run any deceleration ramp.
- WINDOW : represent the max. offset value on the analogic signal accepted by the system. It is run in encoder pulses, where : 1 pulse = 5 mV.
In case the control realizes an offset above the one entered, will be signalled an error of "NOT ENDED MOVEMENT" after reaching the given quote.

INPUT TEST FOR MANUAL

Defines an entered line, through the number of bits (on a number) and the number of ports (on three numbers).
This entered line is tested in manual operativity mode : if it results qualified, the axes' manual displacement mode is turned off. To disable running on this signal, enter the code on a unexisting bit. (9000, as bit number 9 is not significant).

HEAD'S FEATURES

Three selection parameters are assigned to the type of head employed, for the machines suited :

- KIND OF MACHINE

Values to enter : 0 and 1.

By entering the value 0 the head runs only a position of Z concerning tools change.

This position is assigned to the set-point value.

Value 1 makes possible to run the head on two positions Z of tools exchange :

- A first position is assigned on the set-point quote, specific for the aggregates (mills, blades) and horizontal boring tools.
- A second position is assigned on the parameter following, specific for the vertical boring tools.

- VALUE Z OF TOOLS EXCHANGE

Enter the second position of tools exchange, as above.

This parameter is significant only when the parameter KIND OF MACHINE has value 1.

- PANEL'S THICKNESS

Enter the panel's max.thickness, on which is possible to run the tools exchange distinguished in two positions. This parameter is significant only when the parameter KIND OF MACHINE has value 1.

DEFINITION OF HORIZONTAL TOOLS' RAISING

It defines, for the four horizontal boring functions, if the tool ascent (raise) must take place by two Gnn equals and consecutive. [enter (Y/N)].

Thanks to these parameters is possible to avoid kicking the bit of a horizontal tool against ref.points, unloading devices and other fixed mechanical items.

AXES' LIMITS

On each of the four configurable axes is required :

- AXIS' NEGATIVE LIMIT (-) :
Max.negative displacement from the Set-point position before intervention of the axis' extra-stroke sensor.

- AXIS' POSITIVE LIMIT (+) :
Max. positive displacement from the Set-point position before intervention of the axis' extra-stroke sensor.

The parameter "axis' positive limit" for axis W :
it defines also the mobile head's max. opening quote.
The parameters "Axes' limits" are used in program elaboration, to check execution acceptability of a position programmed upon the item in process.

PARAMETERS OF THE LINEARITY CORRECTORS

It is defined a parameter for each of the three axes X, Y, W employed to assign the linearity correctors interval of sampling, in an independent way on each axis.

Meaning with N the entered parameter, the interval of sampling is assigned on the basis of the relation :

$$\text{interval} = \frac{N}{2} * (\text{axis resolution})$$

expressed in [mm] (or [inches])

N should be entered as a whole number, of such a value to cover (including the linearity correctors available) the entire work area.

Example : with resolution = 0.05 mm and N =10 ,
It is imposed an interval of 51.2 mm.

LINEARITY CORRECTORS

They are expressed in [mm] (or [inches]) and indicate the correction values to relate to the entered value, according with the real position on physical axis.

The interval of sampling of the inserted data is determined by the parameter examined in the previous cycle, and the max. correction value enterable is ± 6 .

The correctors are defined, on each axis, in the following way :

AXIS X

- 10 negative values, corresponding to a max. negative displacement (from the Set-point position) equal to 10 times the value of the entered parameter.

If the interval entered is 51.2 mm : result will be 512 mm.

- 70 positive values, corresponding to a max. positive displacement (from the Set-point position) equal to 70 times the value of the entered parameter.

If the interval entered is 51.2 mm : result will be 3584 mm.

AXIS Y

- 10 negative values.

- 30 positive values.

If the interval entered is 51.2 mm, will result a max. positive displacement of 1536 mm.

AXIS Z

- No linearity corrector runs on this axis. (the displacement on this axis is usually very small).

AXIS W

- 5 negative values, corresponding to a max. negative displacement (from the Set-point position) of 256 mm (keeping valid the hypothesis of 51.2 mm as interval).
- 15 positive values.
With interval entered of 51.2 mm, will result a max. positive displacement of 768 mm.

Except for the displacements covered by the linearity correctors, on each axis, no other correction upon the values as entered is performed.

INTERPOLATION PARAMETERS

On each axis are required three parameters :

- NUMBER OF STEPS PER FEED FORWARD : expressed in resolution pulses. It determines (multiplied by the parameter " FEED FORWARD MULTIPLICATOR") the contribution to the speed command given at the start in respect to the feedback error. This parameter is used when performing interpolations. Valid values : from 0 to 32 (value 0 disables feed forward running)
- FEEDBACK ERROR MULTIPLICATION : determines the gain of the position feedback. Acts as a multiplier of the feedback error and should be entered with a range of values going from 1 to 15. Enter 2 means to double the axis' gain resistor, 3 equals to three times as much, and so on. The gain resistor normally fixed is 20K : a gain = 5 means therefore to fix a resistor of 100K.
- FEED FORWARD MULTIPLICATION
It has to be entered on a value among 1 and 15.

MACHINE EMERGENCIES

It is proposed a list up to 16 emergency signals, some with pre-defined meaning, others completely definable by the system user. Every emergency signal is defined upon three parameters :

- parameter of emergency qualification. Enter :
 - 1 to qualify
 - Ø to disable
- assignement on the signal : b p p p. Enter :
 - b = number of bit, from Ø to 7
 - p p p = number of port, from ØØØ to Ø31
- Normal mode on the emergency signal (emergency mode non active)
 - 1 if normally open
 - Ø if normally closed

Each emergency input is related to a procedure run at every switch from normal mode to emergency mode (It is necessary to return in normal mode to qualify again the control on the same input).

This allows, in case of an axis end stroke, to set again the normal working modalities, through the manual operativity.

To each configurable emergency is related a message of error, displayed when the emergency takes place (displayed in the modes : "MACHINE HANDLING" or "MANUAL OPERATIVITIES").

There are eight emergencies availables to run the end stroke on the qualified axes (two end stroke per each axis, one on positive field and one on negative field).

Upon signalation of an axis'end stroke emergency the shiftment on the involved axis stops (without handling the decelerating ramp) and an eventual program execution is resetted.

An axis'end stroke emergency does not require a second execution of Set-point.

An emergency is used upon signal of "STOP" from automatic execution field (program execution, Set-point execution).

It allows a temporary stop of the machine running, with arrest on the axes in motion. (handling the deceleration ramp).

The execution suspended this way can restart upon assent of field "Start" (Configurable in the emergency list).

A typical utilization of the field signals Stop-Start concerns the signals of photo electric cells of machine encubrement, arranged for the safety of the personnel involved in running the processing machinery.

The remaining signals on the emergency list, referred to with the word "EMERGENCY" followed by an identification number, are available. Upon intervention of one of these emergencies every automatic execution running is resetted, the shiftment on the axes is arrested (without handling the deceleration ramp) and an emergency procedure for user's definition is executed (defined on a machine function).

Below, are listed all emergencies configurables on the control, with indication of the message regarding the reason of the emergency itself :

- Emergency 1 : General emergency
(message : 50. EMERGENCY)
- Emergency 2 : Limit X+
(message : 51. LIMIT AXIS X+)
- Emergency 3 : Limit X-
(message : 52. LIMIT AXIS X-)
- Emergency 4 : Limit Y+
(message : 53. LIMIT AXIS Y+)
- Emergency 5 : Limit Y-
(message : 54. LIMIT AXIS Y-)
- Emergency 6 : Limit Z+
(message : 55. LIMIT AXIS Z+)
- Emergency 7 : Limit Z-
(message : 56. LIMIT AXIS Z-)
- Emergency 8 : Limit W+
(message : 57. LIMIT AXIS W+)
- Emergency 9 : Limit W-
(message : 58. LIMIT AXIS W-)
- Emergency 10 : General emergency
(message : 59. EMERGENCY 59)
- Emergency 11 : General emergency
(message : 60. EMERGENCY 60)
- Emergency 12 : General emergency
(message : 61. Emergency 61)
- Emergency 13 : General emergency
(message : 62. EMERGENCY 62)
- Emergency 14 : Field stop
(message : 63. FIELD STOP)
- Emergency 15 : General emergency
(message : 64. EMERGENCY 64)
- Emergency 16 : Field start
(message : 65. FIELD START)

ERRORS DISPLAYED WHEN HANDLING PARAMETERS

Message and description

FILES PAR.LNG E PAR.DAT NOT CORRESPONDING

If the file of parameters storage PAR.DAT results not in line with the corresponding file defining parameters' messages and sizes.

The fault is displayed when accessing the machine parameters and indicate that the parameters read on file PAR.DAT are less than the number required : this means that the installed version of PTP400 do not correspond with the file of parameters used. To the display correspond the prosecution in "Parameters" with entering of 0 for all missing parameters No recording on the parameter file is performed, unless the operator does not perform operations of insertion on parameters

FILE PAR.LNG NOT CORRECT

Displayed when accessing the machine parameters, it indicates that the file of messages definition has not been entered properly. To the display correspond the exit on the machine menu, as it is impossible to enter any other machine parameter.

DATA AS ENTERED CONTAINS NON-NUMERICAL CHARACTERS

Displayed upon entering of a numerical parameter typed with characters not acceptables. A numerical parameter (sizes , speed ,....) can be entered with numbers (from 0 to 9) and with eventual sign and decimals (separator of the decimal part is ".")

PASSWORD NOT CORRECT

Upon entering a wrong password.

PRINTER NOT READY RETRY? (Y/N)

Upon request to print machine parameters with printer not in line. Answer Yes or Not to the question.

PROGRAMS ORGANIZATION

The purpose of this chapter is to describe all available operativities to create and run the machine cycles (programs, or user's programs).

The programs are recorded on Hard-disk, on the "programs'path", as defined in menu "Version of PTP400" (selected from main menu). From here following, with the words "programs ambient" or "programs directory" is meant the programs' path.

The number of programs recordable on Hard-disk depends only from the room available on it.

In program ambient is handled an index file (directory) of programs. This index is in alphabetical order. To each program correspond two recordings on disk : 1 file for the source program (text edited in Video Editor) and 1 file for the related "tabulated" program.

The "tabulated" program correspond to an elaboration on the source text : it is organized on a scheduler-numerical structure, directly accessible from program "MACHINE HANDLING" after inquiry of program compilation.

The programs index file is : INDEX.IND
The source file of a program is named : (program name). SRG
The tabulated file of a program is : (program name). TAB

To store the programs, are run in program ambient more sub-directories : these sub-directories are identified with a letter, from letter A to letter Z.

The program placement in these sub-directories is performed according with the first letter of the program's name :

- Every program beginning with A is recorded in directory \A\ , and so on till letter Z.

FORMAT PROGRAM DISK

This operativity can be selected in Main Menu.

Format a disk means to initialize a disk so that it can be read and written upon by the system.

When a disk is formatted all programs existing on that disk will be erased.

With the installation of system PTP400 on Hard-disk, formatting concerns the preparation of diskettes for :

- Storage of system version PTP400
- Programs' and/or articles' backup.

Infact, programs recording takes place on Hard disk.

The display is :

FORMATTING DISK IN DRIVE (A)

S confirm , N cancel

answer Y <--' or N <--' :

- with N, return to the selection in Main Menu
- with Y, the format is performed

Then, the following is displayed :

INSERT NEW DISKETTE IN UNIT (A)
AND PRESS ANY KEY WHEN READY

insert in drive A : the new diskette and confirm by pressing any key.

When the formatting has ended, the message will appear :

Format tracks ## Format completed
total bytes on disk
available bytes on disk

(The values meant with the symbols "##" depend from the room on disk)

Format another (Y/N)?

Press Y if another disk is to be formatted, N if the formatting has been completed.

In case should appear the message at the end of the formatting :

"nnnn bytes in sectors damaged"

means that the diskette has some sector damaged and it cannot be used.

MAKE PROGRAMS' DIRECTORIES

This operativity is selected in Main Menu.

It is here described the preparation of the ambients of programs and articles.

As already anticipated (see also : Main Menu, selection on Version of PTF400) :

- The user's programs are recorded in a sub-directory structure within the programs' ambient ;
- The user's articles (execution lists defined in "Machine Handling") are recorded in the articles' ambient.

Creation of articles and programs ambient is not performed automatically but must be required by the operator in Main Menu.

In operativity Make directory programs and articles :

1. concerning the program directory :

- If not configurated, it is created
- the same for the index file INDEX.IND (created empty) (on number of program 0) ,
- and for the sub-directories from "A" to "Z".

The test device for creation or existance displays a message such as :

```
Directory ### already existing
Directory ### created
Program index created
```

2. concerning the articles' directory :

- If not existing, it is created
- the same for the index file INDEX.IND (created empty) (on number of articles)

The test device for creation or existance displays a message, similar to the one of the programs.

The initial procedure regarding the preparation of programs and articles' directory it is necessary :

- Infact, the Video Editor procedures concerning programs' run check the preparation of the programs' directory : if the verification fails, no command of programs'handling (directory, programs'creation or deletion) runs.
- The same in "Machine handling" : no program execution runs when programs and articles' directories are not properly installed.

PROGRAMS' BACKUP

This operativity is selected in Main Menu.

The Backup procedure allows to run user's articles and programs, with possibility of handling :

- Creation of copies to be stored (backup) of programs or articles on a different disk, that anyway should be selected among the ones readable by the system
- recover on hard disk of programs/articles previously saved (with backup) on a different disk .
- erasure of program/articles from a selected disk.

Here is proposed the window [MENU 1']:

FILES MANAGEMENT	
* PROGRAMS	COPY
ARTICLES	DELETE
TO POSITION TO SELECT./DESELECT. TO NEXT PAGE TO END	<- -> <- -' END
	END

The biggest square, on the lower part of the screen, propose a quick guide to the user about using the active keys :

- The four arrows on the numerical keyboard allow to move the cursor on the four selectable squares and to produce the required selections
- press <--' (Return) to step further to the next window menu.
- press END (on numerical keyboard) to quit the operativity and to step back to the machine Menu.

On this window, the operator should select :

- If to work with programs or articles (selections on the left side of the screen)
- if delete or copy (selection on the right side of the screen)

When this window is displayed, are active the selections :

- programs
- copy

The labels in a different colour indicates the active selections ; the character "*" indicates the cursor position.

Once selected, press <-- to step to the next window [MENU 2']:

OPERATIVE DRIVES SELECTION

<p>< CONNECTIONS ></p> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"><p>» LOCAL HARD DISK LOCAL FLOPPY A</p></div>	<p>< CORRESPONDING PATH ></p> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"><p>C:\400V80\</p></div> <p><SOURCE CONNECTION></p> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"><p><> LOCAL HARD DISK</p></div> <p><DESTINATION CONNECTION></p> <div style="border: 1px solid black; height: 20px; margin: 5px 0;"></div>
---	--

< || > POSITION <RETURN> SELECT <ESC> EXIT

This window is to select the working drive(s) :

- In case of copy, two drivers should be selected :
 - source (reading)
 - destination (writing)
- In case of deletion, only one driver should be selected .

All possible selection are displayed in the left side square, for a max. of 15 assignments. These define the connections "seen" by the system (diskette of possible access) and are configured when the system is installed : the drives listed are local if the system is not connected to local network, are in part local and in part remote if the system is installed on a local network allowing communication with another PC.

The labels displayed on the left side square are proposed in a symbolic way, like for instance : "local hard disk" instead of "C:\400V80\".

The flashing symbol "»" indicates the cursor position on the left side square : the keys "up" and "down" on the numerical keyboard allow to move the cursor along the possible connection.

Press return <--' to select the connection.

A transversal rhomb symbol (indicated as "<>") flashing on the right side squares indicates the selection to perform : on source connection or on destination connection.

On the square "corresponding path" is indicated the path corresponding to the connection pointed out in the left side square (example : C:\400V80\).

For instance, if the intention is to copy a program from the Hard disk to the diskette in driver A, with reference to the proposed windows, the steps are :

- with the cursor "»" on the first label (local hard disk) press <--' ; on the label "source connection" confirm the connection "local hard disk" and the selection shifted on the label "destination connection" .
- Cursor "»" is already shifted on the second available connection (local floppy A) : press <--'.
- On the label "destination connection" is confirmed the connection "local floppy A".

To the question "DATA CONFIRMED ? " answer Y.

To the next question

"SINGLE CONFIRMATION ON BACK-UP OPERATIONS ? " answer :

- Yes, if the operator follows the next operation of program copy;
- No, if the operator is not following, as for instance, in case of system installed on a local network : in that case, the operator does not have to confirm upon signalation of program / article not copied, because it is already existing on the destination drive.

The next step is the execution menu of the selected operativity (copy or cancel) [MENU 3']:

PROGRAMS	COPY	N. 1
----------	------	------

P001	P002	
PR004	PD03	
ADM001		
ADM002		
>ADM003		
FRES1		
FRES2		
LAMA		
F1		
ADM003	TEST FITTING G20	05-04-1987

TO POSITION	press	: END HOME ETC.
TO SELECT / DE-SELECT		: SPACE
TO NEXT PAGE		: PGDN
TO RUN		: F1
TO END		: F2

The upper label indicates :

- The work environment : programs or articles
- The kind of operation selected : copy or delete
- The number of selection performed : on programs or articles

The central label (rectangle) displays the list of the firsts 64 programs or articles existing on the "source path" : to display the next ones press key PGDN on numerical keyboard.

By shifting the cursors it is possible to select or de-select (disable) programs or articles to perform the operation required (copy or delete). The selection of a program/article is made by pressing the key "SPACE" and it is signalled by a double pointer displayed on the left side of the name.

It is possible to run up to 100 copies or deletions per time.

If more program/article pages are involved, the page selection is possible when it concerns the pages next (in programs/articles directory) but it is not possible on the previous pages.

In case of wrong selection entered on previous pages, is necessary to repeat the operation from the beginning.

The label immediately under the central rectangle (label), displays:

- name
 - description
 - date of last updating
- of the program/article qualified in reverse in the central rectangle (correspondent to cursor position).

In the lower part of the screen it is proposed a help about the usable keys :

- keys CURSORS, END, HOME (on num.keyboard) :
set on program name (name in reverse)
- SPACE (press the space bar) :
select or disable the name of the program displayed in reverse.
- key PGDN :
unfold the next programs/articles page
It does not have any consequence if the displayed page is the last one.
- Function key F1 :
starts the procedure of copy or delete. The execution run with F1 is not resettable, and this is to avoid accidental loss of data.
- Function key F2 :
Goes back to the menu on the firsts operative selections.

MESSAGES WHEN HANDLING RECORDINGS

- NETWORK CONFIGURATION FILE UNCORRECT OR UNEXISTING
On menu 2, indicates the impossibility of make assignments in the label headed "connections" due to reading wrong or uncompleted data when calling the backup procedure.
- CAREFUL : EQUAL PATHS
On menu 2, indicates that have been entered equals selections of source connection and destination connection to copy programs/articles.
- MAX. 100 PROGRAMS/ARTICLES PER TIME
Indicates that it has been reached the max. number of programs/articles to copy or delete.
- ELABORATION FAILED
The occurrence of a fault caused an interruption of the elaboration
- PROGRAM/ARTICLE NOT COPIED
A program/article has not been copied because already existing onto destination disk. The elaboration keeps running normally. To copy the program/article it is necessary to delete it from the destination disk and and copy it again selecting only the names that have not been copied already.
This procedure means to preserve the operator from unwillingly deletion of datas.
- DISK IN DRIVE "#" NOT CORRECT, RETRY ? (Y/N)
The disk has not been inserted properly or it is not the proper disk. Insert the disk correctly and press "Y" to go on or "N" to end the execution.
- DISK IN DRIVE "#" IS FULL
On the destination disk there is no room for new programs ; the elaboration ends anomalously. Format a new disk and make the copy onto it.
- DISK IN DRIVE "#" IS WRITE PROTECTED, RETRY ? (Y/N)
It is not possible to write onto destination disk because it is write protected. Remove the protection, re-insert the diskette and press "Y" to go no and "N" to end the execution.

- DISK IN DRIVE "#" IS EMPTY

The source diskette do not contain any program/article.

- ARCHIVES IN DRIVE "#" NOT CORRECT

On menu 3, indicates that on destination disk it has not been prepared the program/articles' ambient.

To the next question :

"MAKE ARCHIVES FOR PROGRAMS/ARTICLES (Y/N) ? "

answer Y to confirm, other way press N.

PROGRAMS' EDIT

Video Editor mode is selectable in machine menu or in menu "Auxiliary programs of debug".

Video Editor mode allows to create programs to run the machine :
A program is the layout of processings such as boring, milling,...
The max. size of a program has been settled in 300 executive lines.

A program is defined with :

- An alphanumerical univocal name
- A whole of preliminar informations (measure units, panel's size, comment to the program)
- A list of executive instructions determining the processing sequence (displacements, type of process, processing speed, tools selection, ...).

In Video Editor mode, the execution of a program it is always possible : in fact the first window menu asks to enter the program name. If <--' is pressed without entering the program name, it is imposed the fixed name NEW to identify the program involved.

The name of a program :

- can be defined up to a max. of 8 characters.
- The first character has to be a letter (A-Z)
- The letters (A-Z) and the numbers (0-9) are allowed.

If the name contains spaces, they will be deleted.

The restriction of using only alphabetical letters (A-Z), and numbers (0-9) is advised to avoid to fall in the restrictions imposed by the MS-DOS operative system when assigning names to the programs. In that case, can happen recording troubles and a second reading of the same program.

The characters absolutely NOT VALID to assign a program's name, are :

.	dot (point)	"
[]	square brackets	,	comma
?	interrogative mark	+	sign plus
\ /	transversal bars	*	asterisk
=	equals	:	colon
;	semicolon	< >	minor maggiore
	graphical characters and space		

Few names are reserved to and forbidden by the MS-DOS :

AUX,CLOCK\$,LPT1,LPT2,LPT3,COM,COM1,COM2,COM3,COM4,CON,
NUL,PRN,CTTY.

Three RUNNING MODES are available :

- Insert
- Revise
- Command

Mode INSERT allows insertion of text lines on a new or already existing program.

Mode REVISE allows modifications of existing text lines.

Mode COMMAND is the mode with which every other command (Merge, Delete, Line, etc.) runs.

With inserted text lines it is always kept a "current line", indicated with the line cursor ">" positioned within the line itself.

SPECIAL KEYS HANDLED IN EDIT MODE

key PGUP (on numerical keyboard)

displays the previous text page

key PGDN (on numerical keyboard)

displays the next text page

Keys PGUP and PGDN do not result qualified with line cursor located on the line "Begin" .

Keys PGUP and PGDN are qualified only in Command Mode.

Cursor keys (on numerical keyboard)

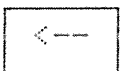


(arrow up)

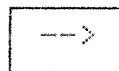


(arrow down)

move the cursor up and down along the displayed text.



(arrow to the left)



(arrow to the right)

move the cursor in the line.
(in Modes Insert and Revise).

key END (on numerical keyboard)

Ends the modes Insert and Revise. In the other commands, allows not to run the command.

key HOME (on numerical keyboard)

shifts the cursor at the beginning of the displayed text .

key INS (on numerical keyboard)

When pressed, allows to write in the text, and the cursor becomes square shaped. Pressing INS again the mode is disabled and the cursor becomes normal again.

This key is qualified in modes Revise and Insert.

key

DEL

 (on numerical keyboard)

Acts on the cursor position on the screen (modes Revise and Insert). Any time the key is pressed the character onto which is the cursor, it is deleted and the text at the right side of the cursor shifts toward the left of one position.

key

ESC

Deletes every character on the line, starting from the cursor position to the end of the line (modes Insert and Revise).

key

<--
-->

 (TABULATOR)

Allows to shift the cursor to the next tabulation mark. (modes Insert or Revise)

function key

Roll UP

 (soft key 7)

Makes visible the line next to the displayed text.

function key

Roll DW

 (soft key 8)

Makes visible the line previous to the displayed text.

MODE "WAIT COMMAND"

In this mode the Editor is qualified to receive a command. Every command is given by pressing one of the soft keys displayed on the last line of the screen.

To change the commands page press soft key ETC (F10).

If there are no text lines some commands are not accepted.

The keys allowed in mode "wait command" are :

cursor up	HOME
cursor down	ROLL UP
PG UP	ROLL DW
PG DN	ctrl + T

In stage of selection of a command line the keys available are :

ESC	
DEL	
END	(disables the command)

When many commands are selected, a new programming is proposed on the line of soft keys :

F1: from	
F2: thru	F7: :LF
F3: all	F8: /
F5: to	F9: into

Please see specific commands clarifications.

This is an example of an edit page display, with a program text running :

```
Begin COMMENT TO NNNN
  1 G10 X100 Y100 Z10
>  2 G10 X200 Y300
  3 G12 X300 Y400 I200 J400
  4 G00 X700
End
```

- Name : NNNN — Dim : 1000 x 500 x 20 [mm] — N. lines: 4 —

Wait command

1	2	3	4	5	6	7	8	9	10
Insert	Revise	Dim	Delete	Merge	Graph	RollUp	RollDw	Line	etc.

The words "Begin" and "End" (automatically entered) delimit the program text.

Next to the word "Begin" is reported the comment related to the program.

The numbering on the lines runs automatically.

On the lower part of the screen are displayed :

- program's name
- panel's dimensions (length x height x thickness) with indication of the measure unit adopted ([mm] o [inch]).
- Number of text lines .

The function keys reported above correspond to the first of the two commands lines available. Pressing the function key ETC (F10) the second command line is displayed :

1	2	3	4	5	6	7	8	9	10
Dir	Kill	Rename			List	Store	End	New	etc.

COMMANDS PROGRAMMED ON FUNCTION KEYS

INSERT

The command INSERT allows to insert one or more text lines, starting from the position next to the "current line" (line indicated by the line cursor ">").

The keys to operate with in mode Enter are :

- left cursor
- right cursor
- PG UP
- PG DN
- INS
- DEL
- ESC
- -->! (TAB)

Syntax : press the correspondent function key .

REVISE

The command REVISE allows to modify one or more text line(s) already entered. When this command is selected, the cursor "_" shifts on the first character of the "current line" .

The keys to operate with in mode Revise are :

- cursor up
- left cursor
- right cursor
- cursor down
- home
- PG UP
- PG DN
- END
- INS
- DEL
- ESC
- <-- (back space)
- -->! (TAB)

Syntax : press the correspondent function key.

DELETE

Deletion of text lines.

Syntax :

DELETE delete the current line
DELETE all delete all lines
DELETE thru n delete starting from current line to line n (included).

MERGE

The command MERGE allows to merge an entire program or part of it with the program on editing.

Syntax :

MERGE (progr.name) merge the whole specified program.

MERGE from (ni) (progr.name)
 merge from line ni of the specified
 program .

MERGE from (ni) thru (nf) (progr.name)
 merge rom line ni to line nf of the
 specified program .

The commant MERGE is disabled when handling panel's dimensions and comment to the program. Besides, it acts only when the same measure unit is involved.

LINE

The command LINE allows to position on the desidered line.

Syntax : LINE first text line
 LINE n specified text line
 LINE 999 last text line

LIST

Sends the edited text or part of it, to the printer (:LP)

Syntax :

LIST to :LP to the printer
LIST ni/nf to :LP the same starting from ni to nf

The print takes place on pages of 55 lines, with indication of :
- Program's name and description.
- Current date
- Text lines, with indication of the line number

DIM

Allows introduction and/or modification of :
- measure unit related to the program
- panel's dimensions
- comment to the program

The dimensions are displayed next to the name (with indication of the measure unit), the comment is displayed on the first text line.

Syntax : Press the correspondent function key .

Then, are required :

- Measure unit
- Panel's length
- Panel's height
- Panel's thickness
- Comment to the program (max. 30 characters)

If a parameter does not have to change, just press <--'.

If the measure unit corresponds to the one entered as general machine configuration parameter : it is enough to press <--' when required.

The dimensions parameters accept only positive values, on max. 6 characters .

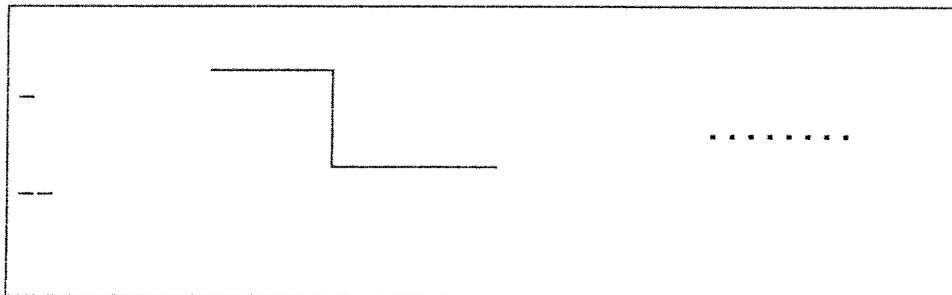
GRAPH.

Displays the panel's graphical representation.

Syntax : Press the correspondent function key

Here, it is proposed an example of graphic representation of a program .

┌ TEST Dimensions 1000 x 500 x 20 [mm]



Zoom : (Ø , 1000)-(Ø , 500)
Tool Ø 0=Screen options or <--' to exit
 -->>

The central rectangle represent a panel.

The axes are : -X horizontal axis
 -Y vertical axis

The axes' starting point is indicated by the graphic symbol "┌" :
in the case represented the starting point is on the upper edge,
on the left (system of reference XY : Ø). In case of system of
reference XY 1 : the starting point would be in the lower edge on
the left.

On the first line are displayed :

-
- Program's name (PROVA)
- dimensions, in shape of (length x height x thickness), and unit of reference ([mm] o [inch]).

Every operation performed on the panel is represented projected on the plane XY.

With a colour screen the various operations are appreciated at their best :

- Vertical borings (circled dot) are displayed in highly bright white colour.
- Horizontal borings (lines drawn perpendicular to one side of the panel) are displayed in highly bright light blue colour.
- Millings (linear and circular) on plane XY are represented in highly bright red colour.
- Millings (linear and circular) on a different plane are displayed in white.
- Blade grooves are represented in highly bright yellow.

Next to the word "Tool" is displayed the tool number referred to the processings set in evidence. (See further : option Tool).
Next to the word "Zoom" is displayed the window displayed (See further : option Zoom).

In the lower part of the screen is reported :
" 0=Screen options or <--' to quit "

Press :

- <--' To go back to the Video Editor menu
- 0 To run one or more graphic option(s)

In the second case, are displayed all the Graphic options availables :

1. option "tool selection" : It is possible to set in evidence the vertical borings performed by a certain tool . Enter one of the following choices :

- Tool number and <--' :
In correspondance of every boring performed by the indicated tool, is displayed a cross (High density red colour with a white dot in the center) .
- Q <--' To quit the graphic mode.
- (space) <--' or
Ø <--' To reset an eventual selection of a tool made earlier.
- <--' To leave as it is the previous selection.

2. option "Zoom" : Displays a part of the panel enlarged till the maximum allowed by the graphic mode. Enter :

- Q <--' : To quit the graphic mode.
- (space) <--' : To access again the "graphic mode" on the entire panel
- (Xinitial) (Xfinal) (Yinitial) (Yfinal) <--' ,to define the window to be displayed :

Xinitial is the 1st position X of window definition
Xfinal is the 2nd position X of window definition
Yinitial is the 1st position Y of window definition
Yfinal is the 2nd position Y of window definition

Between a entered term and the other leave a space.

Example of pattern : 100 500 Ø 300 <--' .

Each pattern is enterable in max. 12 characters, included the eventual sign "-" in negative values (while in case of positive value the sign "+" can be omitted) and the "." (dot) for the decimals.

Concerning each axis, one of the positions can be negative but not both ; besides, the two positions should not be coincident. The window can also end out of the panel.

The measure unit employed to define the values displayed in the window is the same as the one employed to define the values of the program ([mm] o [inch]).

- A <-- ; in that case the zoom keeps on the whole panel (on the programmed dimensions of length and height) ; The minimum zoom defined this way can be enlarged on the max. X and Y quotes programmed out of the panel (on negative quotes and on positive quotes beyond the programmed dimensions).
- <--' To keep qualified the previous selection.

3.option "Grid" : Sets a grid onto the panel,with long pitch X and Y variables and independent. Enter :

- Q <-- : to quit the graphic mode.
- (space) <--' : to avoid displaying the grid .
- (dX) (dY)<--' : To enter the pitch x and y of the grid :
dX is the pitch in direction X
dY is the pitch in direction Y

Leave a space between a pattern and the next.

Example of pattern : 100 50 <--' .

Each pattern can be entered on max. 12 characters,including the "." (dot) of the decimals.

The grid pitches should be entered as positive or void values. Value 0 means to be disabled to employ grid bars in the relative direction. Example : 100 0 <--' displays a grid with only vertical bars, spaced out of each other of 100.

The measure unit employed to set the grid values is the same as the one employed to run the program ([mm] or [inches]).

- <--' to keep qualified the previous selection.

4.option "step" : Composes the programmed processings graphic display in subsequent steps, with inquiry of operator assent to proceed to the next step . Enter :

- Q <--' : to quit the graphic mode
- N <--' : to exclude the option "step"
- S <-- : to qualify the option "step"
- (else) <--' : to keep qualified the previous selection

When running a program graphic with the option "step" qualified :

- In the upper corner on the left of the screen is updated the step running : on the line number corresponding to the edited text.
- On the lower part of the screen are reported : the program line corresponding to the last step displayed and the previous program line.
- To qualify the next step press <--' .
- To disable a step press END (on numerical keyboard).

5. Screen option for Print-Screen : Allows to present the program on a less detailed graphic so that the print can take place with a common use printer. Once the new display is ready press key "Print Screen" to print it.

KILL

The command KILL allows to delete a program already recorded. If the program is under a directory, it displays :

Program under directory, press Y to delete it.

Syntax : KILL program name

STORE

Records the text edited on disk.

Syntax :

STORE writes the text into the program which name has already been specified. (Note 1).

STORE into (nome) writes the text into the program called (name). (Note 2).

Note 1 :

In case of a new program, with name not yet assigned, it is asked: " Program name ? ". If, instead of giving the name, the key END is pressed the edited text is not stored on disk.

Note 2 :

If the program exist in a directory, it asks :
"Program already in directory, delete old program ? (Y/N)".
Answer Y to store the new program and delete the old.
Answer N to return to command STORE.

Note 3 :

The command STORE does not result qualified on a empty program (number of lines = 0).

The procedure of STORE implies a double recording for every edited program.

A first recording correspond to the text as entered and listed in edit, definable as "source text".

The second recording is an elaboration of the source text, and it is definable as "text to run": this program is organized in a scheduler structure that makes the data easier accessed by the program "Machine Handling", upon request of running the program.

The elaboration procedure of this text can last a few seconds and could end with a Fault signal. (Please refer to the detailed description of the milling and boring commands and to appendix A for the definition of all possible errors).

In case of error, no recording is performed, nor of the source text nor of the text to run. Therefore, if the program was already recorded, it will be stored on disk the previous version.

If the elaboration ends without errors signals, both versions are stored on disk with display of the number of lines on the "text to run" (words "N. lines exec. ") .

DIRECTORY

The command DIR displays the programs directory.

Syntax : DIR

The directory is displayed in subsequent pages, till the end of the list of programs in the memory. Every page displays a max. of 16 programs, with : name ; comment : date of entering . The programs directory is listed in alphabetical order.

RENAME

Allows to change name to a program on disk.

Syntax : RENAME old name to new name

NEW

Opens the Editor mode on a new program or on one already stored.

It is displayed the confirming question :

"Are you sure ? (Y/N) " ; by answering :

- Y (yes) the program text onto screen disappears and it is asked the name of the program to load (or to create, if new).

- N (no) the command NEW is ignored.

Syntax : NEW

END

The command END allows to quit the edit.

If the program text is to be saved, operate with "Store". To the question :

"Are you sure (Y/N)" , answer as required.

Syntax : END



This sequence (press key CTRL together with key T), if performed when the first string of softkeys is displayed (1 insert 2 modify) opens a window describing what type of bits is mounted on each mandrel, their diameter or radius for mills and the work field in X and in Y of each tool.
This operativity is qualified only if a loaded program exists.

The key (arrow down) displays the page following.
The key (arrow up) displays the page preceding.
The key END closes the descriptive window and allows to return to the Edit.
Example of window :

Tnn		HELP	
		XMAX	YMAX
T9 D	9x 32.5	3173.0	781.0 mm
T10 A		3173.0	845.0 mm
T11 A	Ø 10.0	3173.0	877.0 mm
T12 A		3173.0	781.0 mm
T13 A		3173.0	781.0 mm
T14 A		3173.0	781.0 mm
T15 A		3173.0	781.0 mm
T16 J	r.10.0	3023.0	1037.0 mm

In each window are displayed the descriptions of 8 tools.
On each line is reported :

Identification number : ex. T16

Sort of tool : ex. J

Particular parameters on the specified sort :

- if sort J : value of mill radius
- if sort D : (n. of bits) x (fitting pitch);
- if different sort than J, D, L, M : value assigned in parameter as mill radius (tool diameter)
- if sort L or M : no display is performed.

Work fields in X and in Y : These values are obtained by subtracting the correspondent value of the corrector (in X or Y) from the axis' positive limit.

All these values are assumed from the machine parameters.

NB. The informations concerning values are proposed in the measure unit specified in program's edit.

ERROR SIGNALS WHEN HANDLING THE EDIT

ARCHIVE NOT READY - PRESS ANY KEY !

It is displayed in case the programs directory results not configured : The signal is proposed when accessing the Edit mode and at every inquire of access any command related with programs handling.

PRINTER NOT "ON LINE"

After command "LIST to LP" means that the printer does not result in mode "on line".

PAPER TO PRINTER HAS ENDED

After command "LIST to LP" : signals that the paper has ended.

NAME TOO LONG

PROGRAM NAME NOT VALID

Indicate a wrong assignment of the program name :

- Name longer than 8 characters (first message)
- First character of the name is not alphabetical. (A-Z).
(Second message)

PROGRAM NOT FOUND

After command "MERGE" if the program specified is not on disk.

DIFFERENT MEASURE UNIT

After command "MERGE", when the program specified relies on a different measure unit than the one specified for the program in Edit.

DISK IS FULL

On the disk there is no room for new programs.

MEMORY IS FULL

This sort of error can take place upon commands INSERT and REVISE, and the message points out the impossibility of running or performing the selected command, being impossible the insertion of more lines in the source text (max. number of lines = 300) .

Upon command INSERT :

1. The message appears when the command is given if the source text is already on 300 lines. Therefore, the command cannot be performed.

2. The message appears after the insertion of the 300th program line. In this case will be displayed :
"MEMORY IS FULL press <--' END .
To quit the command Insert, therefore,press first <--' (enter) and then END (on numerical keyboard). If the command is run again, or other keys than the mentioned ones are pressed, a beep will point out the fault.

Upon command MERGE : The fault takes place when the sub-program lines summed up with the lines of the program in EDIT exceed the number of 300.

PTP400'S MILLING AND BORING COMMANDS

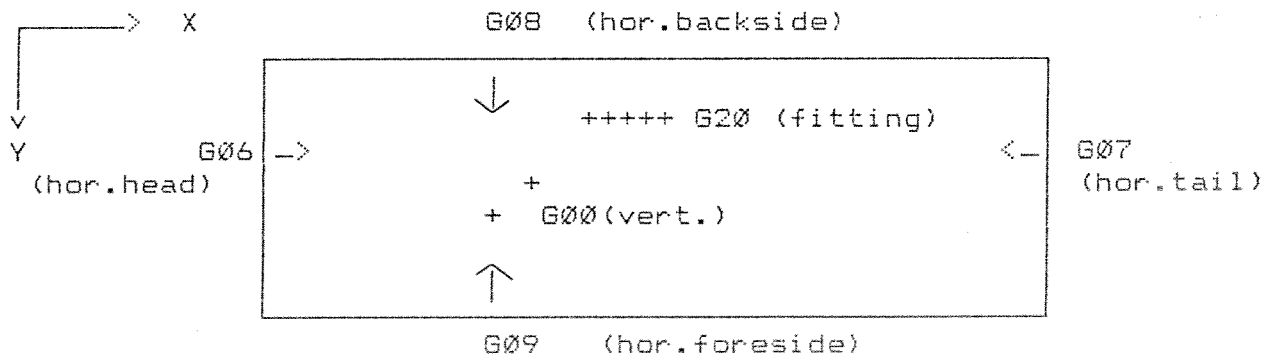
The program source text contains instructions (alphanumerical strings max. 71 characters long), which order determines the operative sequence running the program .
Every instruction is "made" of a number of functions, that define completely the kind of operation required.
The kind of execution is defined by the function Ggg : depending from the value defined in it, the other functions can take different meanings.
It is not, in general, assigned any obliged order to the functions specified on a instruction line.
A text line can besides be entirely defined on a comment : this happens when the character ";" (semicolon) or "'" (apostrophe) is typed at the beginning of the line.

DESCRIPTION OF THE FUNCTIONS

- PREPARATORY FUNCTIONS Gnn

Programmable value :
G followed by 2 numbers (see next board)

With reference to the picture following, the admitted preparatory boring functions are :



In the following board are listed all the preparatory functions.

Ggg	Description
G00	. Vertical boring . End point fitting . Linear interpolation with tool raise at end point. . Blade displacement with tool raise at end point.
G02	.Clockwise circular interpolation with mill raise at end point (*) .
G03	.Same as G02, with c.clockwise rotation (*)
G04	Call function (optional)
G05	Call sub-program
G06	Horizontal boring on the left side of the panel
G07	The same, on the right side
G08	The same, on the backside
G09	The same, on the foreside.
G10	.Quick positioning with mill lowering .Linear interpolation (if run with mill qualified)
G11	Origins programming (optional)
G12	.Clockwise circular interpolation without mill raise at end point. (*) ;
G13	.The same as G12, on c.clockwise rotation (*) ;
G14	.Program concerning clockwise performing of an oval without mill raise at end point (*)
G15	.Same as G14, on c. clockwise rotation (*) ;
G16	.Quick displacement with blade lowering in x
G17	.Quick displacement with blade lowering in y
G20	.Quick displacement to fitting with lowering of drilling tools .
G93	Function " Test panel's thickness"
G98	Programmed stop addressed to the work fied (N,M,T,R,A,S).
G99	Programmed stop.

(*) On the preparatory functions of circular interpolation or execution on a oval, added parameters can be specified as further differentiation of the kind of execution required. The modes of this further specification are defined, in general terms, in the character "/" followed by a single number between 1 and 5.

Example: G03/2 X...
G12/ X...

- AXES' CO-ORDINATES (X, Y, Z)

The co-ordinates are programmed through a letter specifying the axis' name, followed by the sign (if -) and by the value to reach referred to the panel zero.

The letters to address are : X Y Z

Programmable values :

from ± 0.0001 to ± 9999.9999 mm.

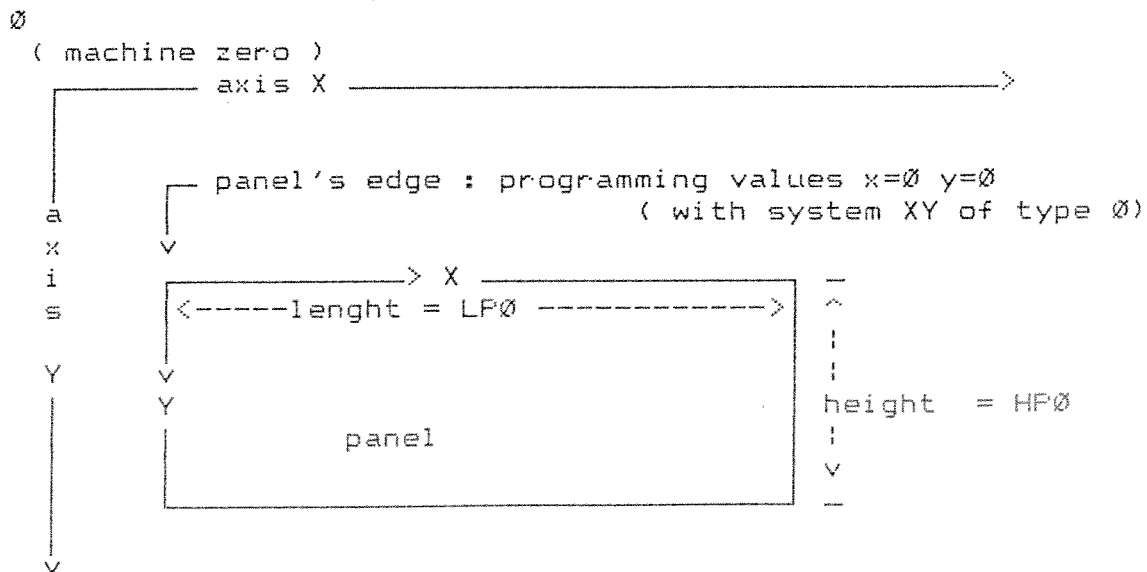
The sign + before the value should be omitted.

With [inches] as measure unit :

from ± 0.00001 to ± 999.99999 inches.

NOTE : If axis Z does not result qualified, the patterns related to this axis are not significant (values functions on Z and K ; speed on VZ) .

NOTE : Programming of the work position takes place assuming as reference system the panel :



NOTE

It is recommended a reasonable use of decimals when programming values functions (X, Y, Z, I, J, K) in order to reduce to a minimum the error signals due to non valid positions (see cases of circular interpolations programming). In particular, avoid the use of decimals not significant as they are more precise than the machine resolutions. For instance :

- if axis X has resolution $R = 0.05$ mm
avoid patterns such as:

```

                                X 1000.0695
                                X 500.0024
program instead                : X 1000.07
                                X 500
```

- CO-ORDINATES OF THE CIRCUMFERENCE CENTER (I , J , K)

In circular interpolations, the co-ordinates of the circumference centre are expressed by the letters :

```

I on the co-ordinate in X
J on the co-ordinate in Y
K on the co-ordinate in Z
```

Programmable values : same as axis' X, Y, Z co-ordinates functions.

- CIRCUMFERENCE RADIUS (I)

To define a circular interpolation through the radius, is employed the function I with sign \pm (the sign + should be omitted).

Programmable value : same as axis' X, Y, Z coordinates functions.

- CIRCUMFERENCE POINT OF INTERMEDIATE PASSAGE (I , J)

When defining a circular interpolation passing through three points, assign the intermediate point on the functions :

```

I on the co-ordinate in X
J on the co-ordinate in Y
```

Programmable value : same as axes' X, Y, Z co-ordinates functions.

- FUNCTION OF PLANE SELECTION WHEN MILLING (P) :

Defines the plane, or area, in milling operations :

PXY plane XY (when milling in linear or circular interpolations)
PXZ plane XZ (when milling in linear or circular interpolations)
PYZ plane YZ (when milling in linear or circular interpolations)
PXYZ area XYZ (when milling in helicoidal or linear
interpolations)

Milling on different planes than XY is possible only for those machines that allows control of axis Z in interpolation.

- FUNCTIONS OF TOOL SELECTION (T) :

Define the tool(s) required for the preparatory function (G) specified.

They are programmed with the letter T followed by 2 numbers.

In vertical and horizontal drilling functions is admitted the contemporary lowering of up to 9 tools.

With selection of contemporary lowering of more than a tool :

- Air values of vertical borings are reckoned on the basis of the lower tool.
- Air values of horizontal borings are reckoned on the basis of the correctors of the first specified tool.
- The work positions (on the panel upper plane or on one of the side faces) are related to the first specified tool.

- USER'S FUNCTIONS (F) :

They define user's functions exploitable in the program. They are programmed with the letter F followed by three numbers among 130 and 150.

Function F should appear alone on the program line.

- VELOCITY (VX VY VZ) :

It is programmed with VX, VY and VZ followed by the value desired.
Programmable values :
from 0.1 to 99 mt/min.
from 0.01 to 99.99 inches/sec.

The values programmed should comply with the max. velocity entered in Parameters for that kind of processing.

Here are listed all possible uses of velocity functions :

G00 : Vertical boring

VZ defines the tool's descent velocity from the air value to the working value .

G20 : Boring on fitting

VZ defines the tool's descent velocity from the air value to the working value assumed from every fitting's reference point.

G10 : Quick positioning with mill lowering

VX tangential velocity in interpolation
VZ defines the mill's descent velocity from the air value to the work value.

G10 G00 : Linear interpolation
G02 G03 G12 G13 : Circular interpolation
G14 G15 : Programming of ovals

VX Tangential velocity in interpolation.

G06 G07 G08 G09 : Horizontal boring

VX Velocity of entry into the panel

G16 G17 : Grooving with blade

VX Blade's X and Y displacement velocity
VZ defines the tool descent velocity from the air value to the working value.

NOTE

For every velocity omitted it is assumed the value entered in Parameters for the kind of processing involved.

Function VY has a specific use when programming simplified connections (see chapter : Simplified arcs programming).

Infact, it defines the interpolation velocity on the connecting arc. If the VY programming is not present, on the connecting arc is applied the valid interpolation speed, programmed on fuction VX or assumed from Parameters.

Programs on function VY not related to simplified connections are filtered.

A proper use of the velocity functions VX, VY, VZ is reccomended : that means to program the functions Vi as above specified for every operation defined on function G.

Programs not responding to the criteria exposed can produce difficulties of comprehension, in program interpretation and when employing the drilling section.

For instance, programming the function VZ on a G milling function (G12, G13, ...) does not affect the speed variations related to axis Z.

Similarly, are not significant speed programs of the G00 of closing fitting (G00 that follows a function G20), or of a blade processing (G00 that follows a function G16 or G17) .

- MANDREL'S ROTATION VELOCITY (VM) :

It is programmed with VM followed by the value desired.
Programmable value : from 0 to 1000.

The values programmed should be in compliance with the velocities entered in Parameters for the kind of processing required, in its minimum and maximum values.

Here are listed all possible uses of the velocity functions :

G00	:	Vertical boring
G20	:	Borings on fitting

VM defines the mandrel's rotation velocity upon descent of axis Z from the air value to the working value.

G10	:	Quick positioning with mill lowering
-----	---	--------------------------------------

VM defines the mandrel's rotation velocity upon descent of axis Z from the air value to the working value.
The mandrel speed keeps the same till the end of the milling cycle.

G06 G07 G08 G09	:	Horizontal boring
-----------------	---	-------------------

VM defines the mandrel's rotation velocity upon descent of axis Z to the working value. The mandrel speed is kept till the end of the boring cycle.

G16 G17	:	Grooving with blade
---------	---	---------------------

VM defines the mandrel's rotation velocity upon descent of axis Z to the working value. The mandrel speed is kept till the end of the grooving.

For all omitted velocities is assumed the (maximum) value entered in Parameters for the kind of processing involved.

It is recommended a proper use of the velocity function VM : program the function as above specified, for each mode defined on function G. For instance, function VM programmed on a milling function G (G12, G13, ...) does not produce any effect, on the variation of the mandrel rotation speed.

Similarly, to program velocities on the G00 of end fitting (G00 following a function G20), or of a blade processing, (G00 following a function G16 or G17) does not result significant.

- FUNCTION "MILL RADIUS" (R)

Indicates the correction value to apply to a milling cycle.

Programmable value : same as functions of axes' co-ordinates.

- FUNCTION "(A)"

This function is employed to define different parameters.

Programmable values : from ± 0.0001 to ± 9999.9999

1) FUNCTION : RADIUS OF THE AUTOMATIC CONNECTION

Indicates the radius value when an automatic connection is required.

2). RADIUS UPON DEFINITION OF A OVAL

Indicates the smaller radius, in the geometric determination of an arc of oval.

3). SELECTION IN ARCS AUTOMATIC PROGRAMMING

When programming an arc on tangent straight line, the value 0 on function A indicates the selection of this kind of program. Therefore, in these cases, to function A is not related any dimensional parameter.

- FUNCTION "ORIGIN" (O)

Function O is usable in two different ways :

- 1). Selects the reference origin for axes with co-ordinates XY.
Four origins are pre-defined (with values ranging from 0 to 3) and each is reported to one of the four panel edges.
Besides, 6 more Origins are available, with values ranging from 4 to 9, free to be programmed on the panel. In this case, on function O can be specified an added parameter to further define the selection of the programmed origins.
Example : O5 .
- 2). Programs the panel reference origins.
The origins programmables are identified with numbers ranging from 4 till 9.
Syntax : O(n) X... Y... with n as a value from 4 till 9.

- FUNCTION "SUB-PROGRAM" (S)

Displays the name of the program loaded as sub-program.

- FUNCTION "ROTATION" (D)

Indicates the eventual rotation to use when running a sub-program.
Patterns accepted are :

D0	D270
D90	DX
D180	DY .

For a complete description of programming functions see the chapters following.

DESCRIPTION OF COMMANDS DISPLAYS

VERTICAL BORING COMMAND

G00 : Vertical boring

X xxxx.xxxx value x of the hole, referred to the origin selected;
Y yyyy.yyyy value y of the hole, referred to the origin selected;
Z zzzz.zzzz depth of the hole, automatically corrected according to panel's thickness.
T tt tools selected (1 .. 99). It is allowed the presence of more codes T for borers with contemporaneous lowering, with a max. of 9 tools.
VZ vv.v velocity of entry in the panel, for suited borers.
VM mmmm Rotational speed of the tool working (for suited borers)
O o Origin on the co-ordinated axes.
; after a semicolon it is possible to insert a comment.

Example : G00 X100 Y100 Z10 T4 VZ3.5 ;comment

In the vertical boring syntax, it is obliged to specify the field T.

NOTE

A symbology such as "X xxxx.xxxx" is meant to give an immediate indication of the max. usable size for the function to enter (here on X, that means [4.4]). It should be remembered that the measure unit referred to in general is [mm].
If the measure unit is [inches], interpretate the functions of value and speed according with the max. sizes related to this unit ([5.3] on quotes, [2.2] on velocity).

NOTE

Any reference to the program's origin will be generally omitted, in the commands defined further, in order to simplify their comprehension. Where the origin is reported it is as necessary fulfillment to the comprehension of the command itself. However, it has to be remembered that the definition "absolute value" is not utilized in a totally proper way as it always refers to the last origin selected in the program.

NOTE

The comment can be inserted in any program line.

COMMAND OF LOAD SUB-PROGRAM

G05 : Load sub-program

X xxxx.xxxx value X of the first point programmed in the sub-program

Y yyyy.yyyy value Y of the first point programmed in the sub-program

Ssssss name of the sub-program to be loaded.
After the sub-program's name it is obliged to type a space. Any program can be loaded as sub-program, as long as it results defined on the same measure unit ([mm] or [inches]) used for the program running.

Dd Sub-program's rotation code :
Ø - no rotation
9Ø - 9Ø degrees clockwise rotation.
18Ø - 18Ø degrees clockwise rotation.
27Ø - 27Ø degrees clockwise rotation.
X - mirrors the sub-program along axis X
Y - mirrors the sub-program along axis Y
if this parameter is not specified, code Ø (no rotation) is assumed.

Ttt Point out the tool working on the milling operations existing in the sub-program. If this parameter is not specified, the sub-program is loaded with the tools specified in its text.

R rrrr.rrrr specifies the mill radius correction value to enter in the interpolations existing in the sub-program to load.

Oo Origin's code. It specifies the panel edge to which are referred the values X and Y of the first point of the sub-program (see paragraph : Origins)

Example : G05 X1ØØ Y1ØØ SPIPFO D9Ø O2 R1Ø ;comment

HORIZONTAL BORING COMMANDS

G06 Horizontal boring on the left side of the panel

X xxxx.xxxx hole's depth.
Y yyyy.yyyy hole's absolute value y.
Z zzzz.zzzz value of tool entry in the panel (automatically corrected according with panel's thickness)
tt Tools selected (1 .. 99). It is allowed the presence of more T codes for borers with contemporaneous descent.
VX vv.v Velocity of entry in the panel.
VM mmmm Rotation velocity of the tool working, for suited borers.

Example : G06 X10 Y200 Z10 T6 VX3.5 ;comment

G07 Horizontal boring on the left side of the panel

X xxxx.xxxx Hole's depth
Y yyyy.yyyy Hole's absolute value y
Z zzzz.zzzz Value of tool entry in the panel (automatically corrected according with panel's thickness).
T tt Tools selected (1 .. 99).It is allowed the presence of more T codes for borers with contemporaneous descent.
VX vv.v velocity of entry in the panel.
VM mmmm Rotation velocity of the tool working (for suited borers).

Example : G07 X10 Y200 Z10 T7 VX3.5 ;comment

G08 Horizontal boring on the backside of the panel

X xxxx.xxxx Hole's absolute value x
Y yyyy.yyyy Hole's depth
Z zzzz.zzzz Value of tool entry in the panel (automatically corrected according to panel's thickness).
T tt Tools selected (1 .. 99). It is allowed the presence of more T codes for borers with contemporaneous descent.
VX vv.v velocity of entry in the panel.
VM mmmm rotation velocity of the tool working (for suited borers)

Example : G08 X200 Y10 Z10 T8 VX3.5 ;comment

G09 Horizontal boring on the foreside of the panel

X xxxx.xxxx hole's absolute value x.
Y yyyy.yyyy hole's depth.
Z zzzz.zzzz value of tool entry in the panel (automatically corrected according to panel's thickness)
T tt Tools selected (1 ..99). It is allowed the presence of more T codes for borers with contemporaneous descent.
VX vv.v velocity of entry in the panel
VM mmmm rotation velocity of the tool working (for suited borers).

Example : G09 X200 Y10 Z10 T9 VX3.5 ;comment

COMMANDS OF PANEL MILLING

G10 Mill descent on panel (first milling instruction)

X xxxx.xxxx absolute value x of descent
Y yyyy.yyyy absolute value y of descent
Z zzzz.zzzz depth of mill entry (automatically corrected according with panel's thickness)
T tt Tools selected (1 ..99). It is allowed the descent of only one milling tool .
VX vv.v velocity of tangential milling.
(max. velocity allowed : the one of Parameters)
VZ vv.v velocity of entry in the panel (for suited borers).
VM mmmm Rotation velocity of the tool working (for suited borers)
R ± rr.r mill diameter expressed in mm. if specified. It defines the contouring correction. (see paragraph Mill radius)
P ppp plane (area) in interpolation.

Example : G10 X100 Y100 Z10 T10 VX1.5 ;comment

G10 Mill linear displacement onto panel without raise

Plane XY : X xxxx.xxxx absolute value x of end point milling.
Y yyyy.yyyy absolute value y of end point milling.

Example : G10 X200 Y200 PXY ;comment (plane xy)

Plane XZ : X xxxx.xxxx absolute value x of end point milling.
Z zzzz.zzzz absolute value z of end point milling.

Example : G10 X200 Z15 FXZ ;comment (plane xz)

Plane YZ : Y yyyy.yyyy absolute value y of end point milling.
Z zzzz.zzzz absolute value z of end point milling.

Example : G10 Y200 Z15 FYZ ;comment (plane yz)

Plane XYZ : X xxxx.xxxx absolute value x of end point milling.
Y yyyy.yyyy absolute value y of end point milling.
Z zzzz.zzzz absolute value z of end point milling.

Example : G10 X200 Y200 Z15 PXYZ ;comment (plane xyz)

G02 Circular clockwise interpolation with final mill raise
G03 Circular c.clockwise interpolation with final mill raise
G12 Circular clockwise interpolation without final mill raise
G13 Circular c.clockwise interpolation without final mill raise

Plane XY : X xxxx.xxxx absolute value x of end point
Y yyyy.yyyy absolute value y of end point
I iiii.iiii absolute value x of the center
J jjjj.jjjj absolute value y of the center

Example : G12 X200 Y200 I100 J200 ;comment (plane xy)

Plane XZ : X xxxx.xxxx absolute value x of end point
Z zzzz.zzzz absolute value z of end point
I iiii.iiii absolute value x of the center
K kkkk.kkkk absolute value z of the center

Example : G13 X80 Z5 I80 K15 ;comment (plane xz)

Plane YZ : Y yyyy.yyyy absolute value y of end point
Z zzzz.zzzz absolute value z of end point
J jjjj.jjjj absolute value y of the center
K kkkk.kkkk absolute value z of the center

Example : G13 Y80 Z5 J80 K15 ;comment (plane yz)

G02 , G03 , G12 , G13 Helicoidal interpolation (with the same specifications, on function GT, given for circular interpolations)

The digit on function P should be : PXYZ.

X xxxx.xxxx absolute value x of end point
Y yyyy.yyyy absolute value y of end point
I iiii.iiii absolute value x of the center
J jjjj.jjjj absolute value y of the center
Z zzzz.zzzz absolute value z of end point

Example : G12 X200 Y200 I100 J200 Z15 ;comment

G00 linear displacement with mill raise

Plane XY : X xxxx.xxxx absolute value x of end point milling
 Y yyyy.yyyy absolute value y of end point milling

Example : G00 X80 Y100 ;comment (plane xy)

Plane XZ : X xxxx.xxxx absolute value x of end point milling
 Z zzzz.zzzz absolute value z of end point milling

Example : G00 X80 Z10 ;comment (plane xz)

Plane YZ : Y yyyy.yyyy absolute value y of end point milling
 Z zzzz.zzzz absolute value z of end point milling

Example : G00 Y100 Z10 ;comment (plane yz)

Plane XYZ : X xxxx.xxxx absolute value x of end point milling
 Y yyyy.yyyy absolute value y of end point milling
 Z zzzz.zzzz absolute value z of end point milling

Example : G00 X80 Y100 Z10 ;comment (plane xyz)

**G14 Oval trajectory programming, with clockwise interpolation,
without mill final raise**

**G15 Oval trajectory programming, with c.clockwise interpolation,
without mill final raise**

X xxxx.xxxx absolute value x of end point
Y yyyy.yyyy absolute value y of end point
I iiii.iiii semiaxis along x
J jjjj.jjjj semiaxis along y
A aaa.aaaa smaller radius.

Example : G14 X200 Y200 I50 J100 A20 ;comment

Programmings of ovals are accepted only on plane XY.

NOTE

The mill's lowering command G10 should always be interpreted as "vertical milling". That means : the plane XY defines the milling plane, while the tool enters the panel always along axis Z.

NOTE

In the mill's displacement commands G12,G13,G02,G03,G14,G15,G10, G00 :

the functions T, R, VZ, and VM (respectively : Tool, value of mill radius, velocity of tool entry in the panel, and mandrel's rotation velocity) are accepted only when the mill lowering command (beginning with G10) is given .

All tangential speed commands can be varied at every milling line with the command VXvv.v, to run passages with different interpolation speed.

NOTE

The command to select the work plane can be specified and varied at every milling line : according with the processing required, the operator should specify the interpolation plane.

Example :

1. G10 X100 Y100 Z10
2. G10 X200 Y200 PXY
3. G00 Y300 Z5 PYZ

On line 1. the function G10 runs the mill descent onto the panel : the values on axes X and Y define the panel's processing origin. The value of Z defines the depth of mill descent (entry) into the panel.

On line 2. the function G10 runs a linear displacement onto plane XY.

On line 3. the function G00 runs a linear displacement onto plane YZ with tool's final raise.

BLADE GROOVING COMMANDS

G16 Blade grooving toward X (blade lowering)

X xxxx.xxxx absolute value x of descent
Y yyyy.yyyy absolute value y of descent
Z zzzz.zzzz depth of blade entry (automatically corrected according to panel's thickness)
T tt tool selected (1 .. 99). It is allowed the descent of only one blade tool.
VZ vv.v velocity of entry into the panel (for suited borers)
VX vv.v velocity of shifting on axis x (max. allowed velocity : the one entered in Parameters) .
VM mmmm rotation velocity of the tool at work (for suited borers)

G17 Blade grooving toward Y (blade lowering)

X xxxx.xxxx absolute value x of descent
Y yyyy.yyyy absolute value y of descent
Z zzzz.zzzz depth of blade entry (automatically corrected according to panel's thickness)
T tt tools selected (1 .. 99). It is allowed the descent of only one blade tool.
Vz vv.v velocity of entry into the panel (for suited borers)
VX vv.v velocity of displacement on axis y (max. allowed velocity : the one entered in Parameters)
VM mmmm rotation velocity of the tool at work (for suited borers)

G00 linear displacement with blade raise

X xxxx.xxxx absolute value x of blade end point for command G16
or :
Y yyyy.yyyy absolute value y of blade end point for command G17

Example : G16 X-10 Y200 VX10 VZ2 T16 ;comment
G00 X1010

Example : G17 X100 Y-10 VX10 VZ2 T16 ;comment
G00 Y460

The values X are automatically updated, at the moment of running mirrored programs, to keep unchanged the direction of blades displacement.

Therefore, program for instance blade displacements in x always from the left to the right (x ----> x+) and blade displacements in y always from up to down (y ----> y+).

FITTING COMMANDS

G20 Multiple and double fitting along axis x

X xxxx.xxxx absolute value x of start fitting
Y yyyy.yyyy absolute value y of start fitting
Z zzzz.zzzz Hole's depth (automatically corrected according to panel's thickness)
T tt Tools selected (1 .. 99)
VZ vv.v velocity of entry into the panel (for suited borers)
VM mmmm rotation velocity of the tool at work (for suited borers).

G00 end fitting

X xxxx.xxxx absolute value x of end point fitting
Y yyyy.yyyy clearance between two rows of fitting (for borers suited to handle a fourth axis)

Example 1 : (fitting on a single row of holes)
G20 X20 Y100 Z10 T20 VZ3
G00 X800

performs a row of holes with pitch ?? mm (same as assembled head) with the first hole at X20 Y100 and the last at X800 (*).

Example 2 : (only for borers handling the fourth axis)
G20 X20 Y100 Z10 T20 VZ3
G00 X800 Y200

performs two rows of holes with pitch ?? mm (same as assembled head) with the first row at point X20 Y100 and the last hole at point X800, and the second row with the first hole at point X20 Y200 and the last hole at point X800 Y200 (*).

Example 3 : (fitting with independent mandrels)
G20 X20 Y100 Z10 T11 T12 T13 T14 T15
G00 X800

performs a row of holes with pitch ?? mm (same as the distance between two mandrels) qualifying the specified tools Tnn , with the first hole at X20 and the last at X800 (*).

NOTE (*)

In any case the descents are automatically reckoned. The first hole of the fitting correspond exactly to the first value entered; the last hole, instead, can also not be at the end quote if the fitting pitch does not arrange it exactly : in this case the last hole approximates per default the end point.

PROGRAMMED STOP COMMANDS

G98 Field stop programmed

performs a significant function according with the kind of processing going on (N, T, ..) : that means a function per every work field foreviewed.

A typical use (hence the name given to the code G98) consists in stopping the processing cycle, and start it again later according with the modalities earlier defined in the function selected . (example : press key Start for assent).

The functions run by "G98" are described on Resident Functions (Chapter : SYSTEM'S AUXILIARY PROGRAMS)

FUN 041 / 047 : run execution on area N long / short
FUN 042 / 048 : run execution on area T long / short
FUN 043 / 049 : run execution on area M long / short
FUN 044 / 050 : run execution on area A long / short
FUN 045 / 051 : run execution on area S long / short
FUN 046 / 052 : run execution on area R long / short

G99 Programmed Stop

Function similar to the previous one, but with stop independent from the kind of processing going on.

The function, handled on G99, is defined on resident functions, as FUN 040.

INSERTION OF USER'S FUNCTIONS IN BORING PROGRAMS

Fnnn Insert a function Fnnn with number included among 130 and 150 to run customized procedures.
The user's functions are defined on resident functions (from FUN 130 to FUN 150).
When running user's functions, no automatic positioning is performed on the machine's axes.

INSERTION OF TEST FUNCTIONS CONCERNING PANEL'S THICKNESS

G93 function : test panel's thickness

X xxxx.xxxx absolute value x of procedure qualification
Y yyyy.yyyy absolute value y of procedure qualification
T tt tool selected (1 .. 99) it is allowed the descent of only one tool, of type H.

Example : G93 X80 Y100 ;comment

To the program on function G93 correspond the execution on function FUN 093 (resident function, parametrical).
In case of axis Z not qualified, the program on function G93 is not run, as it is filtered in the program's elaboration stage.
Please refer to "FUN 093" and "Machine cycles" for a detailed description of the procedure to acknowledge the panel's real thickness.

NOTE

Commands G98, G99, G93 and the User's functions, cannot interrupt a processing function defined on more program lines. In fact, these commands are not accepted after functions G06, G07, G08, G09, G16, G17, or when running a milling cycle .

GENERAL EXAMPLES OF PROGRAMS' LAYOUTS

Example 1 :

```

1  G00 X100 Y100 Z12 T1      ; vertical boring Ø 4 mm
2  G00 X200 Y100 Z15 T2      ; vertical boring Ø 5 mm
3  G06 X10 Y100 Z10 VX2 T6    ; 1 horizontal head
4  G06 X10 Y300 Z10 VX2 T6    ; 2 horizontal head
5  G07 X10 Y100 Z10 VX2 T6    ; 1 horizontal tail
6  G07 X10 Y300 Z10 VX2 T6    ; 2 horizontal tail
7  G17 X-10 Y50 Z5 VX8 T17    ; blade lowering
8  G00 X1010                  ; blade displacement and raise

```

Example 2 :

```

1  G10 X100 Y100 Z10 VX3      ; mill lowering
2  G10 X200                    ; linear displacement (plane xy)
3  G12 X300 Y200 I200 J200 VX1 ; circular displacement with speed
                                ; variation
4  G00 Y400 VX3                ; linear displacement and mill
                                ; raise.

```

Example 3 :

```

1  G10 X100 Y100 Z10          ; mill lowering
2  G10 X400 Y150 Z10          ; linear displacement (plane xy)
3  G12 X600 Y150 Z10 I500 J150 ; circular displacement
4  G10 X600 Y350 Z10          ; linear displacement
5  G10 X450 Y250 Z10          ; linear displacement
6  G10 X200 Y350 Z10          ; linear displacement
7  G13 X100 Y450 Z10 I200 J450 ; circular displacement
8  G10 X900 Y450 Z10          ; linear displacement
9  G13 X650 Y200 Z10 I650 J450 ; circular displacement
10 G02 X800 Y50 Z10 I800 J200 ; circular displacement and mill
                                ; raise.

```

Example 4 :

```

1  G10 X100 Y200 Z10 PXY      ; mill lowering
2  G10 X400 Y150              ; linear displacement (plane xy)
3  G10 X600 Z15 PXZ           ; linear displacement (plane xz)
4  G10 X800 Y400 Z10 PXYZ     ; linear displacement (area xyz)
5  G00 Y100 Z5                ; linear displacement and mill
                                ; raise.
6  G00 X400 Y180 T1           ; vertical boring

```

SPREADINGS AND AUTOMATIC SELECTIONS

Automatic spreading of the values

To shorten the introduction time of boring programs there is the possibility of spreading automatically the boring values, the velocities, the Gnn command and the Tnn Tool.

Example : Perform a serie of vertical holes with tool T1 at different values y but at the same value x :

```
1 G00 X100 Y30 Z10 VZ2 T1
2           Y50
3           Y120
4           Y180
5           Y400
6 X110 Y420
```

Values x and z, function G00 and tool T1 are automatically spread since the first line, and they keep qualified till line 6 where the sole value X is varied.

NOTE : If in the first program line are not specified the values of all qualified axes (X, Y, Z) on the missing axes it is assumed value 0 (zero).

Automatic spreading of the milling plane

Command P of milling plane selection is spread automatically.

Lacking any specification on this command, per default is assumed PXY : plane XY.

Command P is significant only in milling operations : any other kind of operation is run independently from what entered in this command.

Example :

```
1 G10 X100 Y200 Z10 ;mill lowering
2 X400 Y250 ;linear displ. (plane XY default)
3 G00 X600 Z15 PXZ ;linear displacement on plane XZ
and mill raise
4 G00 X700 Y200 T1 ;vertical boring
5 G10 X800 Y400 Z10 PXYZ ;mill lowering and selection of
area XYZ
6 G00 Y100 Z5 ;linear displacement (area XYZ) and
mill raise
7 G00 X400 Y180 T1 PYZ ;vertical boring.Selection of plane
YZ (not affected by the boring)
8 G10 X500 Y300 Z8 ; mill descent (plane : YZ)
9 G00 Y350 Z5 ; linear displacement on plane YZ
and mill raise.
```

Automatic spreading of Origins

Command O, selecting the program's origin, is spreaded automatically.

Lacking any specification on this command, per default is assumed O0 : Origin 0.

Automatic selection of the tool (special tools)

In vertical boring commands G00 it is obliged to specify the tool to be selected.

In : milling commands, fitting, horizontals, it is not always obliged to specify the tool, as the tool (a mill, a blade in x or in y, an horizontal tool in x or y, a fitting head) is selected automatically.

Example : perform a milling like :

```
G10 X100 Y100 Z10 ; Mill lowering
G00 X200 ; Displacement and raise
```

the tool is searched automatically in the head's configuration and it will be assigned as T the corresponding value.

If there are more tools with the same function, it will be selected the last one found, that means the one assigned to the mandrel with the highest value.

Automatic selection of velocities

Every command of velocity is optional : if omitted, is assumed as velocity the one entered in Parameters, for each running mode.

Milling velocity
Blades velocity
Velocity of entry into horizontal holes
Velocity of exit from horizontal holes
Velocity of entry into vertical holes
Velocity of aggregates lowering
Velocity of aggregates raise.

The same criterion is applied when programming the mandrel's rotation velocity, for each running mode.

Automatic preservation of a selected velocity

If a velocity value is entered for a given command (ex. G06 ... VX1.5 horizontal entry velocity ; G06 ... VM500 mandrel's rotation velocity in horizontal boring operations), it will be kept as it is for every command of the same kind, till different specification.

Example: G06 VX1.5 enter entry vel. 1.5 mt/'
 G06 it is preserved
 ...
 ...
 G10
 ...
 ...
 G06 it is preserved
 G06 VX2 enter entry vel. 2 mt/'

Therefore, it is not always necessary to enter a velocity per each function, as it exists in Parameters a related value automatically assumed and that, in general, comply with the needings of the tool at work.

NOTE : No automatic spreading is performed on :

- functions I and J
- function A
- added specifications on function G (see programming of ovals and simplified arc programming)
- functions S and D.

USE OF DRILLING PROGRAMS

Function "Vertical boring" : G00

It runs vertical borings with possibility of selecting more tools.

G00 X xxxx.xxxx Y yyyy.yyyy Z zzzz.zzzz Tn1 Tn2 ...

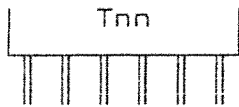
where: X xxxx.xxxx value x of boring
Y yyyy.yyyy value y of boring
Z zzzz.zzzz boring depth
Tn1 Tn2 .. boring tools : select up to 9 tools.

- It is necessary to program the tool to be utilized in the command line : in this mode, infact, the tool automatic search mode in heads'configurations does not result qualified.
- A fault is displayed (WRONG TOOL) when selecting : a J type tool (mill), a L or M tool (blade), a P or Q or R or S (hor.heads), H (measuring head) .
- When more boring tools have been selected, the specified position (x, y) corresponds to the working position of the first tool selected.
- The boring tools can have single bit (type A), multiple bits along x (types B and C), and multiple bits along y (types E, F, G, I).
- The values (x, y) programmed correspond to the working position of the head's central point (or the central tool on a head with tools in odd number).

Function "fitting" : G20

Function G20 allows to run a double fitting along axis X, with programming of the values : start (x) and end (x). Depending from the borer configuration, there are three possible programs :

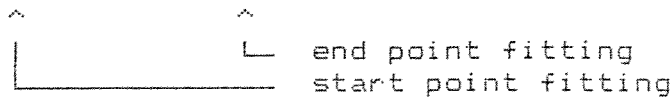
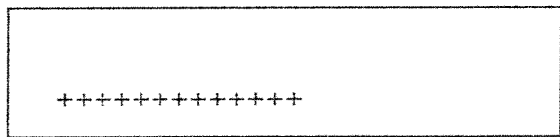
BORERS WITH FIXED SINGLE FITTING HEAD



```
G20 X xxxx.xxxx Y yyyy.yyyy Z zzzz.zzzz
G00 X xxxx.xxxx
```

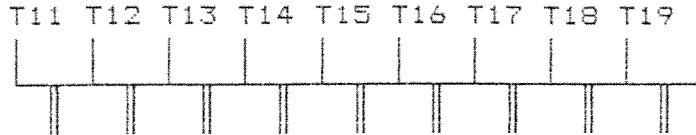
- On the first line (function G20) :
 - X , Y start point fitting
 - Z boring depth
- on the second line (function G00) :
 - x end point fitting.

- It is not necessary to program tool Tn in the command line, as it is searched automatically within the head's configurations. If no type D tool is in the head's configurations, an error will be displayed.



- The first hole of the fitting correspond to the start point.
- The last hole can also not be made in the end point if this is not at (n * pitch) mm from the start point (with "n" as a whole number) ; in this case the last hole will be at the closest possible point short of that measure.

BORERS WITH FITTING BY INDEPENDENT MANDRELS



- In this kind of borers, in which the fitting head is made out of several single mandrels (tools configured as type A tools), it is possible to lower the bits with the pitch desired by specifying in command G20 which bits are to be enabled.
- It is always necessary to specify the sequence of tools to be selected, tools that have necessarily to be set along axis x.
- Tools numbering should reflect a constant pitch (see examples).
- No control is performed concerning the relative pitch constancy between two tools called in sequence. The fitting pitch is reckoned on the value of the correctors along x of the two tools at the outermost points, and on the number of specified tools.
- What has been said concerning the fitting with fixed head is worth also to reckon the boring ref.points to be performed and, eventually, the end point approximation .

```
G20 X xxxx.xxxx Y yyyy.yyyy Z zzzz.zzzz Tn1 Tn2 Tn3 .. TnN
G00 X xxxx.xxxx
```

- On the first line (function G20) :
- X , Y start point fitting
 - Z boring depth
- on the second line (function G00) :
- X end point fitting

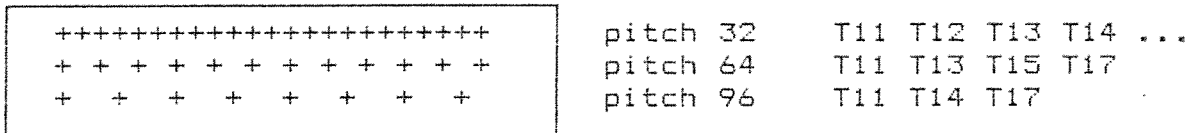
$$\text{fitting pitch} = \frac{(\text{corr. x tool nN-th}) - (\text{corr. x tool n1-th})}{\text{number of specified tools} - 1}$$

Example :let's suppose that the pitch between two bits is 32 mm

when programming : G20 Xnn Ynn Znn T11 T12 T13 T14 T15
G00 Xnn
the pitch is 32

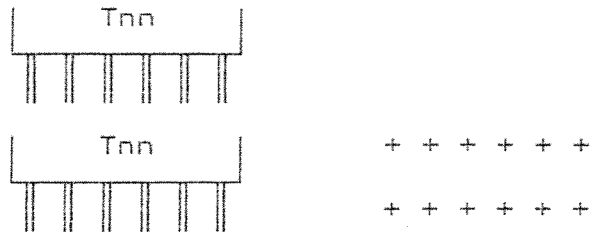
when programming : G20 Xnn Ynn Znn T11 T13 T15 T17 T19
G00 Xnn
the pitch is 64

when programming : G20 Xnn Ynn Znn T11 T14 T17
G00 Xnn
the pitch is 96



NOTE : A program such as :
G20 X xxxx.xxxx Ynn nn T11 T12 T14 T15
will not be accepted, since it does not respects a
constant pitch in the tools numbering.

BORERS WITH TWO PARALLEL FITTING HEADS



Syntax : (for 4 axes borers handling double fitting)

This kind of borers can handle 2 parallel fitting heads along axis x and makes possible to perform with a single lowering two parallel rows of holes, with different values y.

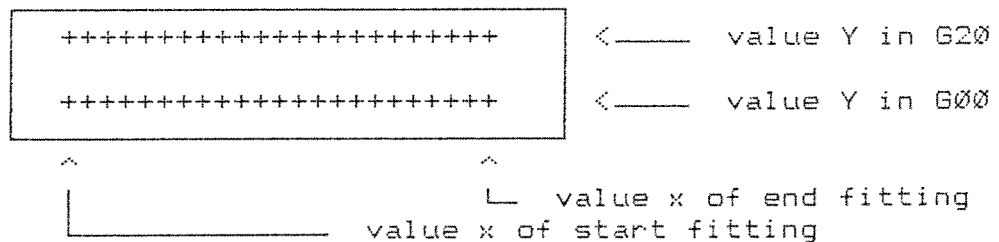
```
G20 X xxxx.xxxx Y yyyy.yyyy Z zzzz.zzzz
G00 X xxxx.xxxx Y yyyy.yyyy
```

On the first line (function G20) :

- X , Y start point fitting
- Z boring depth

on the second line (function G00) :

- X end point fitting
- Y position Y on second fitting head (mobile head)



If in function G00 value Y keeps unchanged, the fitting is performed with only one head.

The working tool can be specified :

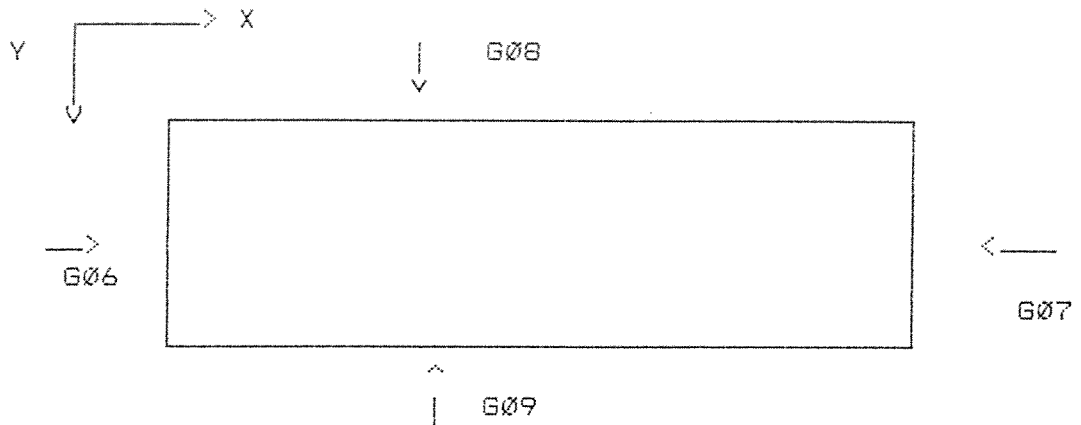
- with fixed head (D type tool)
- with independent mandrels.

ERRORS DISPLAYED WHEN EDITING THE FITTING

Error no.	Message and description
42	<p>TYPE OF MILLING NOT POSSIBLE</p> <ul style="list-style-type: none"> . On line G2Ø of tool lowering, without any tool explicitly selected, if in the configuration does not exist any tool of type D
37	<p>SEQUENCE OF G NOT VALID</p> <ul style="list-style-type: none"> . If on the line following function G2Ø the function GØØ of tool raise has not been typed.
5Ø	<p>FITTING NOT POSSIBLE</p> <ul style="list-style-type: none"> . If the number of holes required is less than the number of mandrels on D type tool. . On fitting with independent mandrels when only one tool has been selected. . With variation of value Y between the line of start fitting and the next line GØØ of end fitting , with handling of axis W disabled.
44	<p>WRONG TOOL</p> <ul style="list-style-type: none"> . In fitting with independent mandrels when calling tools not belonging to type A . In fitting with independent mandrels when the Tn sequence does not have a constant pitch or the fitting pitch = Ø . On line G2Ø of tool lowering when only one tool has been selected, and it does not belong to type D.

use of horizontal boring commands : G06 G07 G08 G09

Functions G06, G07, G08, G09 allows to perform horizontal borings, that means on the panel edge :



- G06 performs horizontal boring on the left side of the panel
G07 performs horizontal boring on the right side of the panel
G08 performs horizontal boring on the backside of the panel
G09 performs horizontal boring on the foreside of the panel
- On G06 e G07, the boring position is pointed out by Y and Z, while X is the work axis.
On G08 e G09, the boring position is pointed out by X and Z, while Y is the work axis.
- There is no need of programming the working tool : it will be automatically searched among the heads' configurations.
If no tool results of type P (Q, R, S), a fault is displayed.
(UNFOREWIEVED KIND OF EXECUTION).
- It is possible to specify more tools on the same boring line :
all tools should be of the kind correspondent to the specified G
other way "WRONG TOOL" will be displayed .

EXAMPLES OF BORING PROGRAMS

Example n.1 (picture 5.1)

```
1. G00 X100 Y100 T1
2. X200
3. X300
4. X400
5. G20 X400 Y300
6. G00 X900
7. G20 X900 Y400
8. G00 X400
9. G06 X10 Y100 Z5
10.Y200 Z10
11. Y300 Z15
12. G07 X12 Y300 Z5
13. Y400
```

Lines 1, 2, 3, 4 run single borings.

Lines 5 and 6 define a fitting on fixed head (type D), and the same for lines 7 and 8.

The tool is the same for both fittings (tool selected automatically), and the fitting pitch is 32 mm.

In this two fitting the value x of start/end fitting is exchanged: in the first case the start value is x=400 mm, and the end value is x=900 mm; the other way round in the second case.

The holes made in each fitting are 16.

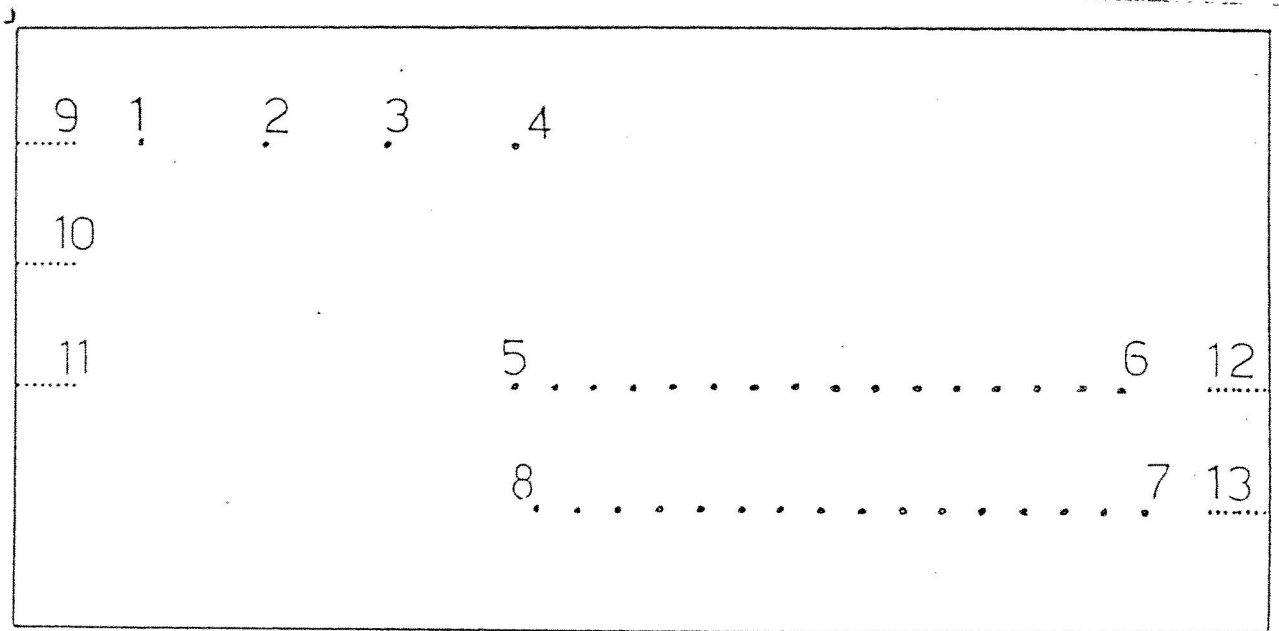
In the first case the position of the first hole is x=400 mm (value of start fitting) and the position of the last hole is x=880 mm (end value approximated short of the measure).

In the second case the position of the first hole is x=900 mm (value of start fitting) and the position of the last hole is in x=420 mm (end quote approximated short of the holes' number)

Lines 9, 10 and 11 run horizontal drillings on the left side of the panel (G06) with only one tool (automatic selection). On these three lines the working position z changes, while the boring depth keeps the same (function X).

On lines 12 and 13 are typed two horizontal borings on the right side of the panel (G07) with the same boring depth (X12).

Picture 5.1



Example n.2 (picture 5.2)

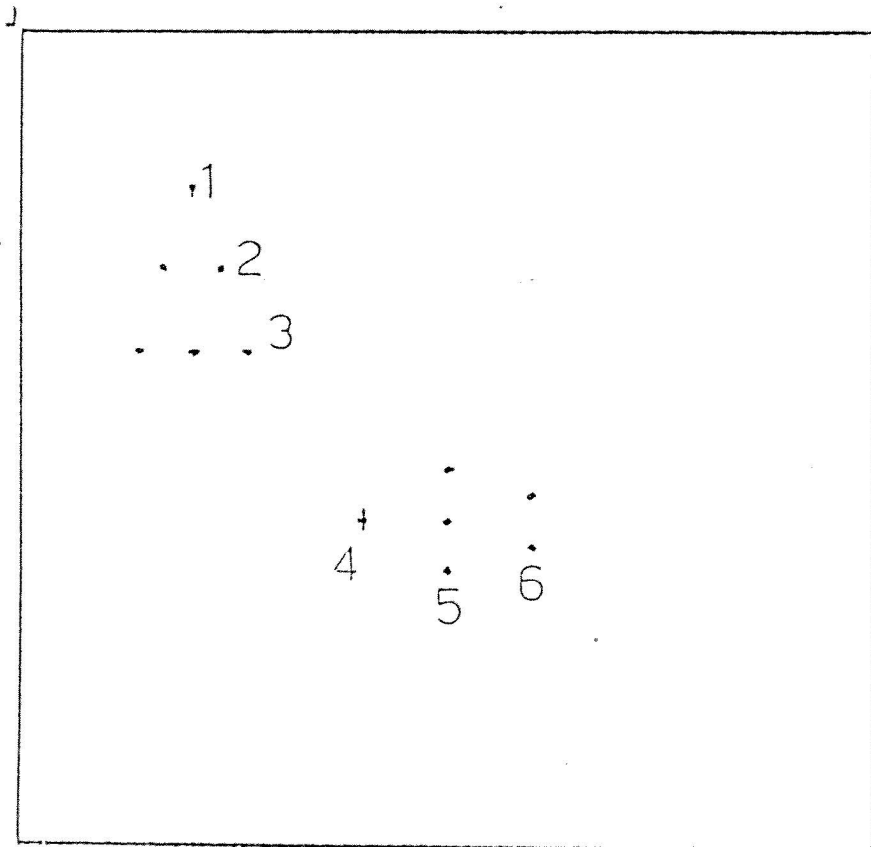
1. G00 X100 Y100 Z5 T1
2. Y150 T2
3. Y200 T3
4. X200 Y300 T1
5. X250 T4
6. X300 T5

The program runs vertical borings with selection of tools defined as different sorts :

tool	T1	type A (single hole)
tool	T2	type B (two holes along x)
tool	T3	type C (three holes along x)
tool	T4	type E (three holes along y)
tool	T5	type F (two holes along y).

The first three program lines run tools with bits along x (besides T1), with the boring position x that keeps unvaried.
The last three program lines run tools with bits along y (besides T1), with the boring position y that keeps unchanged.
Picture 5.2 shows how the tool correctors (in x and y) are set at the center of each tool.

Picture 5.2



USE OF MILLING COMMANDS

Linear milling functions

-function G10 : Mill (J type tool) lowering on panel.
Linear interpolation.

-function G00 : Linear milling with tool final raise.

SYNTAX of linear milling functions (with selection of plane XY) :

G10 X xxxx.xxxx Y yyyy.yyyy Z zzzz.zzzz [Vvv.v] [Rrrrrr.rrrrr] ..
1st. line : mill lowering on position x,y,z

G10 X xxxx.xxxx Y yyyy.yyyy .. mill displacement to pos. x,y

.....
.....

G00 X xxxx.xxxx Y yyyy.yyyy .. end at position x,y and mill
raise

Circular milling functions

- function G12 : Clockwise circular interpolation without mill final raise.
- function G13 : C.clockwise circular interpolation without mill final raise.
- function G02 : same as G12, but with mill final raise.
- function G03 : same as G13, but with mill final raise.

SYNTAX of circular milling functions (with selection of plane XY):

G10 X xxxx.xxxx Y yyyy.yyyy Z zzzz.zzzz
G12 X xxxx.xxxx Y yyyy.yyyy I iiiii.iiii J jjjj.jjjj
G....

where (on G10) :
X xxxx.xxxx value x of arc beginning
Y yyyy.yyyy " y " "
Z zzzz.zzzz milling depth

where (on G12) :
X xxxx.xxxx value x of arc end
Y yyyy.yyyy " y " "
I iiiii.iiii co-ordinate x of the center
J jjjj.jjjj " y " "

SYNTAX of circular milling functions (with selection of plane XZ):

G10 X xxxx.xxxx Y yyyy.yyyy Z zzzz.zzzz
G12 X xxxx.xxxx Z zzzz.zzzz I iiiii.iiii K kkkk.kkkk
G....

where (on G10) :
X xxxx.xxxx value x of arc beginning
Y yyyy.yyyy " y " "
Z zzzz.zzzz arc beginning depth

where (on G12) :
X xxxx.xxxx value x of arc end
Z zzzz.zzzz arc final depth
I iiiii.iiii co-ordinate x of the center
K kkkk.kkkk " z " "

SYNTAX of circular milling functions (with selection of plane YZ):

G10 X xxxx.xxxx Y yyyy.yyyy Z zzzz.zzzz
G12 Y yyyy.yyyy Z zzzz.zzzz J jjjj.jjjj K kkkk.kkkk
G....

where (on G10) :
X xxxx.xxxx value x of arc beginning
Y yyyy.yyyy " y " "
Z zzzz.zzzz arc beginning depth

where (on G12) :
Y yyyy.yyyy value y of arc end
Z zzzz.zzzz arc final depth
J jjjj.jjjj co-ordinate y of the center
K kkkk.kkkk " z " "

Helicoidal milling functions

- function G12 : Clockwise helicoidal interpolation without mill final raise.
- function G13 : C.clockwise helicoidal interpolation without mill final raise.
- function GØ2 : same as function G12, with mill final raise.
- function GØ3 : same as function G13, with mill final raise.

In helicoidal interpolations the displacement is obtained with a circular progression onto plane XY connected to a linear advancement along Z.

The digit on function P should be PXYZ.

SYNTAX of helicoidal milling functions :

```
G1Ø X xxxx.xxxx Y yyyy.yyyy Z zzzz.zzzz
G12 X xxxx.xxxx Y yyyy.yyyy Innn.nnnn J jjjj.jjjj Z zzzz.zzzz
G....
```

where X	xxxx.xxxx	value x of end point.
Y	yyyy.yyyy	" y "
I	iiii.iiii	co-ordinate x of the center
J	jjjj.jjjj	" y "
Z	zzzz.zzzz	end position along axis z.

programming ovals

-function G14 :
-function G15 :

They allow simplified arc programming upon four quadrants with possibility of achieving 1/4, 1/2, 3/4, and 1 whole oval. All programs concerning ovals are accepted only on plane XY.

A full oval is defined on four circumference arcs, identical two by two ; the start and end point are always on a quadrant; the semiaxes are always parallel to axes X and Y. The execution of functions G14 and G15 begins and ends with the tool low.

SYNTAX of functions G14 and G15 :

```
G10 X xxxx.xxxx Y yyyy.yyyy Z zzzz.zzzz  
G14/n X xxxx.xxxx Y yyyy.yyyy I nnn.nnnn J jjjj.jjjj A aaaa.aaaa  
G....
```

where (on G10) :
X xxxx.xxxx value x of start point milling
Y yyyy.yyyy " y " "
Z zzzz.zzzz milling depth

where (on G14) :
X xxxx.xxxx value x of end point milling
Y yyyy.yyyy " y " "
I nnn.nnnn semiaxis along x
J jjjj.jjjj semiaxis along y
A aaaa.aaaa smaller radius
/n is a further specification (not obliged)
on function G

The two values of the semiaxis are always assumed as absolute values (in other words, always positive). The smaller radius should always be smaller than the two semiaxis. If the two semiaxes are equals, the function is interpreted as an arc programming on the radius.(see "SIMPLIFIED ARC PROGRAMMING").

Consider the pictures 6.1 and 6.2 :

Picture 6.1 represents a full oval with major semiaxis along axis X.

Picture 6.2 represents a full oval with major semiaxis along axis Y.

The possible start/end points are : A , B , C , D.

The circumference arcs that achieve the oval trajectory are :

arc 1 : through the points (P4 , A , P1), with P4 and P1 as outermost points

arc 2 : through the points (P1 , B , P2), with P1 and P2 as outermost points

arc 3 : through the points (P2 , C , P3), with P2 and P3 as outermost points

arc 4 : through the points (P3 , D , P4), with P3 and P4 as outermost points

Arcs n. 1) and 3) have radius r1 and centers along the straight line through A and C ;

Arcs n. 2) and 4) have radius r2 and centers along the straight line through B and D.

In picture 6.1 the smaller radius results to be r1; in picture 6.2, the smaller radius results to be r2.

However : the minor radius defines the arcs with centers along the major semiaxis.

The major radius is automatically determined, so to guarantee the trajectory continuity upon radius variation.

THEREFORE :

The use of functions G14 and G15 surely makes it simpler to program closed trajectories, comparable to circumference arcs with tangency continuity. The programming modes allow execution of the full closed figure or of part of it. (1/4, 1/2, 3/4).

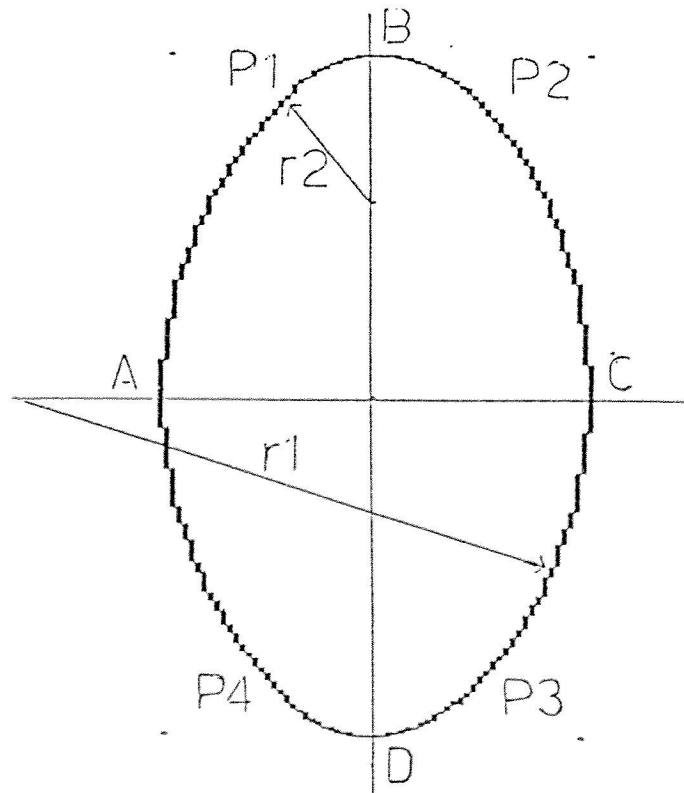
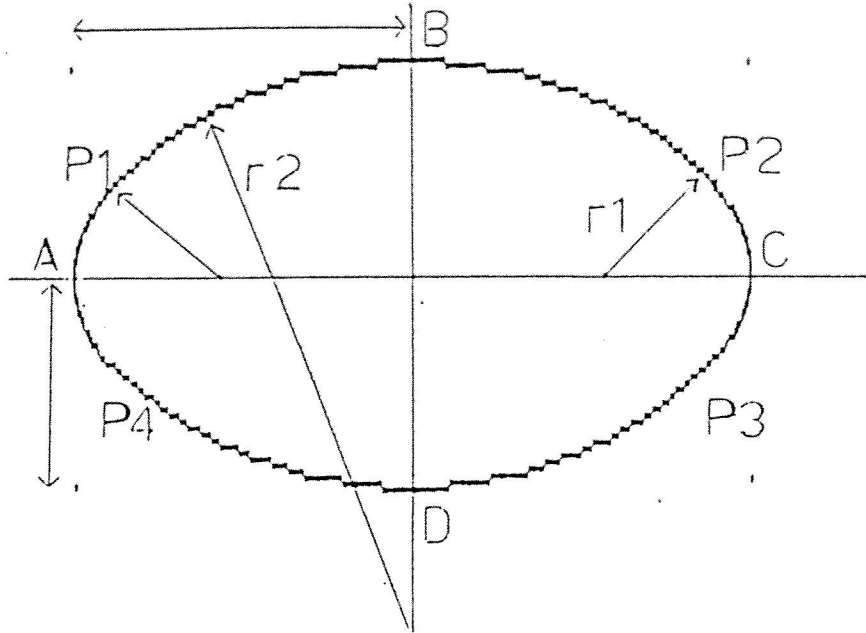
Programming on functions G14 and G15 implies the addition of executive blocks in comparison with the source text. The numbers of blocks added depends from the portion of oval in execution :

1. when executing 1/4 of oval : a block is added.
2. when executing 1/2 oval : two block are added
3. when executing 3/4 of oval : three blocks are added.
4. when executing a full oval : four blocks are added.

As already said, the execution of functions G14 and G15 begins and ends with the tool low. The tool final raise (ascent), if required, should be programmed on a function G00, at the same points X and Y already programmed on functions G14/G15.

In this case (functions G14/G15 followed by function G00 of tool raise) the number of blocks added to the execution of the oval is lessened by one.

PICTURES 6.1 and 6.2

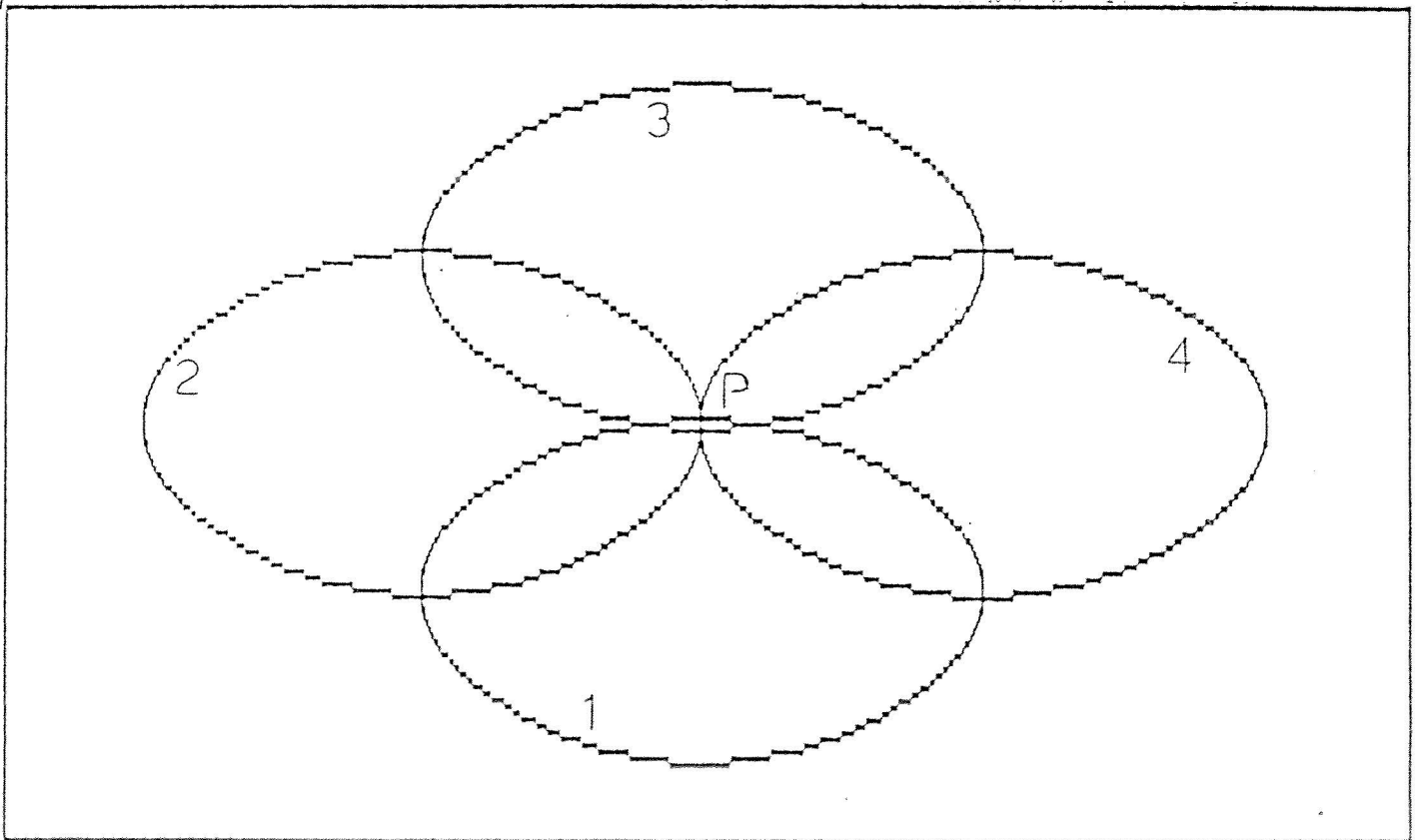


If the end point and the start point coincide, it is performed the full oval. In that case : (see picture 6.3)

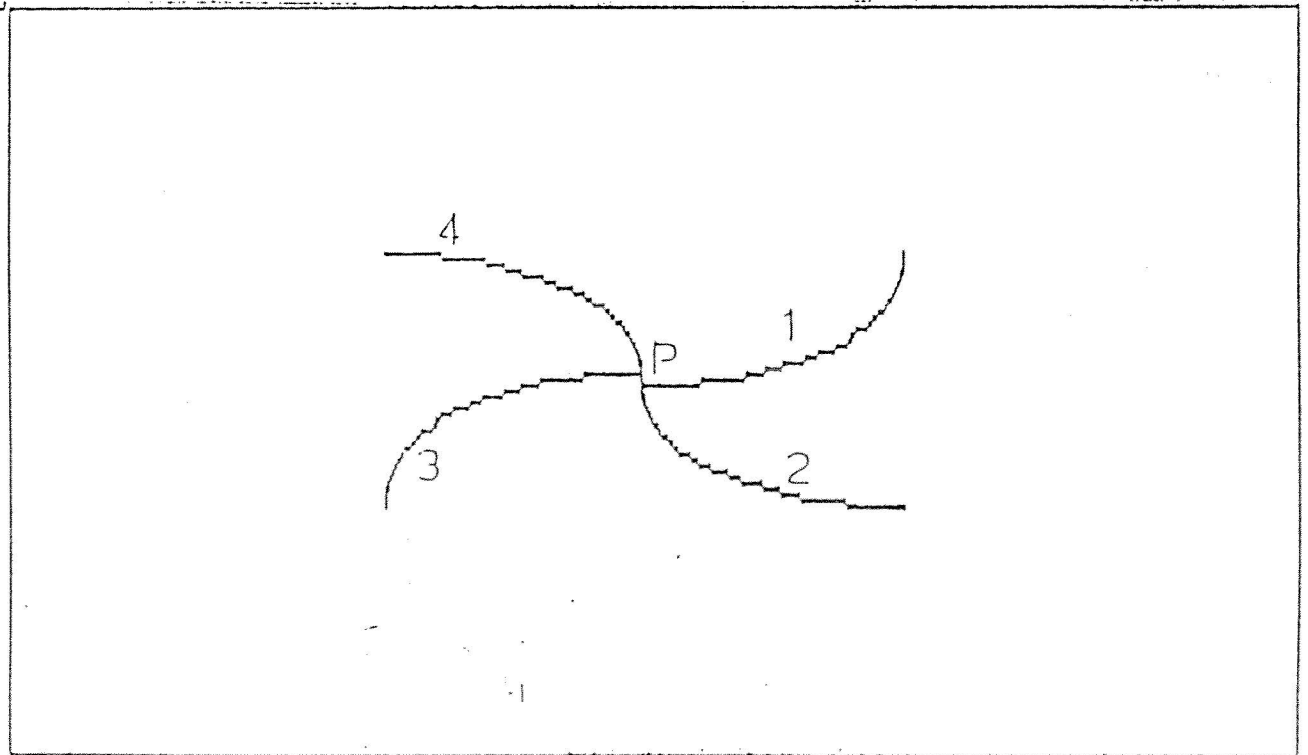
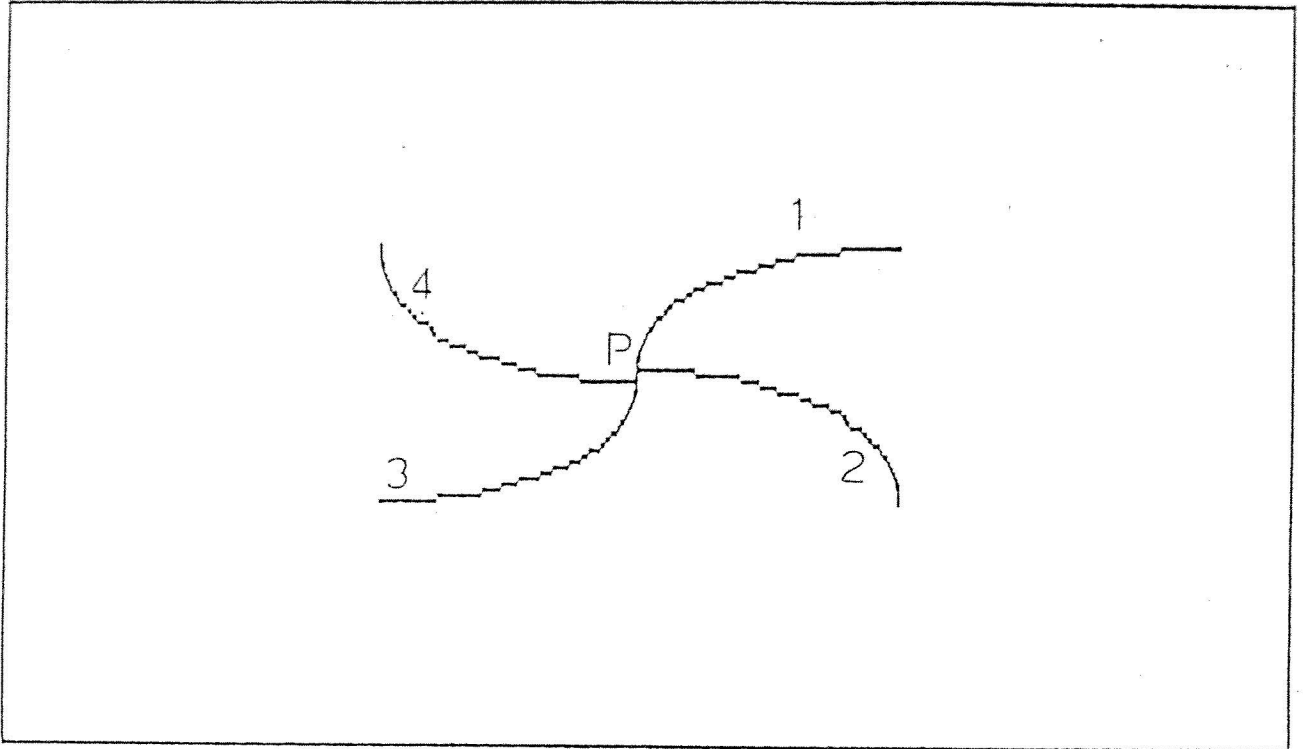
- It performs the oval (1) if nothing more is specified on G, or if "/1" is specified.
- It performs the oval (2) if "/2" is specified
- It performs the oval (3) if "/3" is specified
- It performs the oval (4) if "/4" is specified

Start and end point are marked with P.

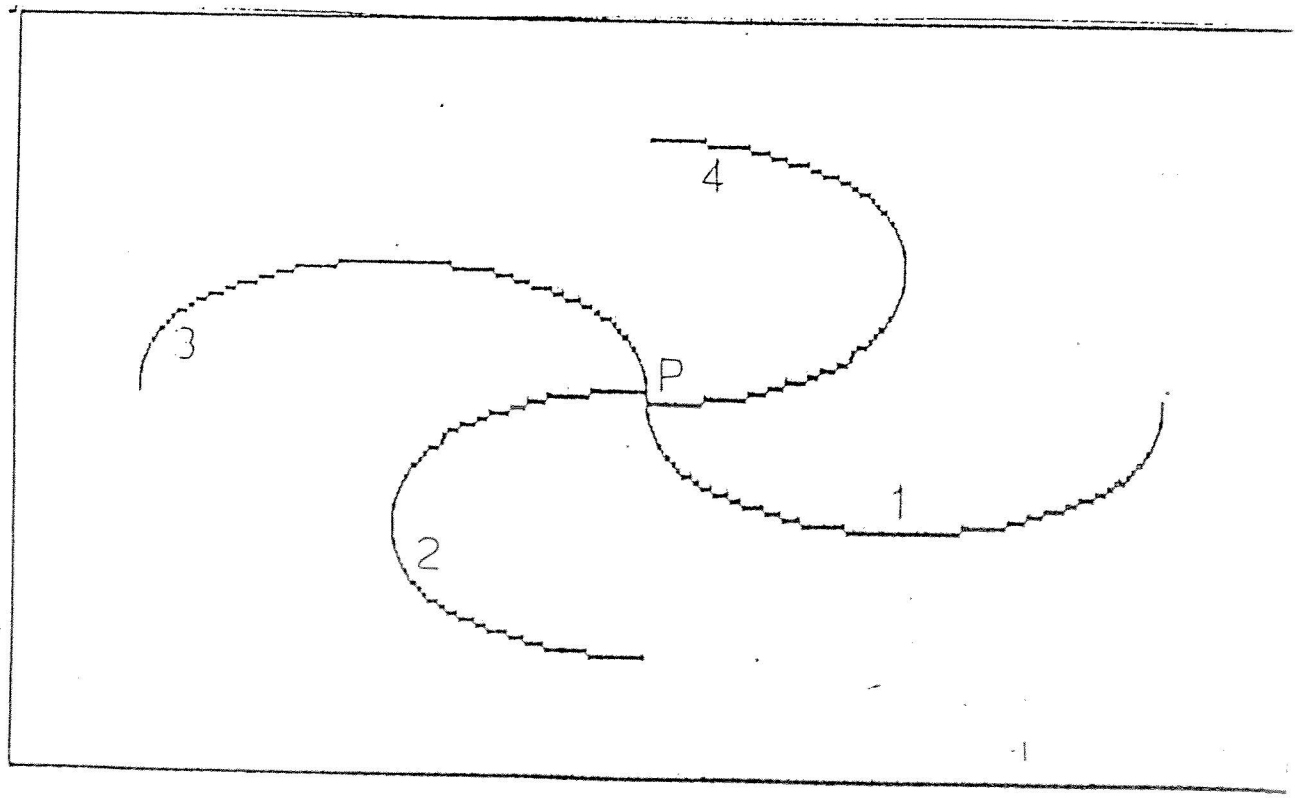
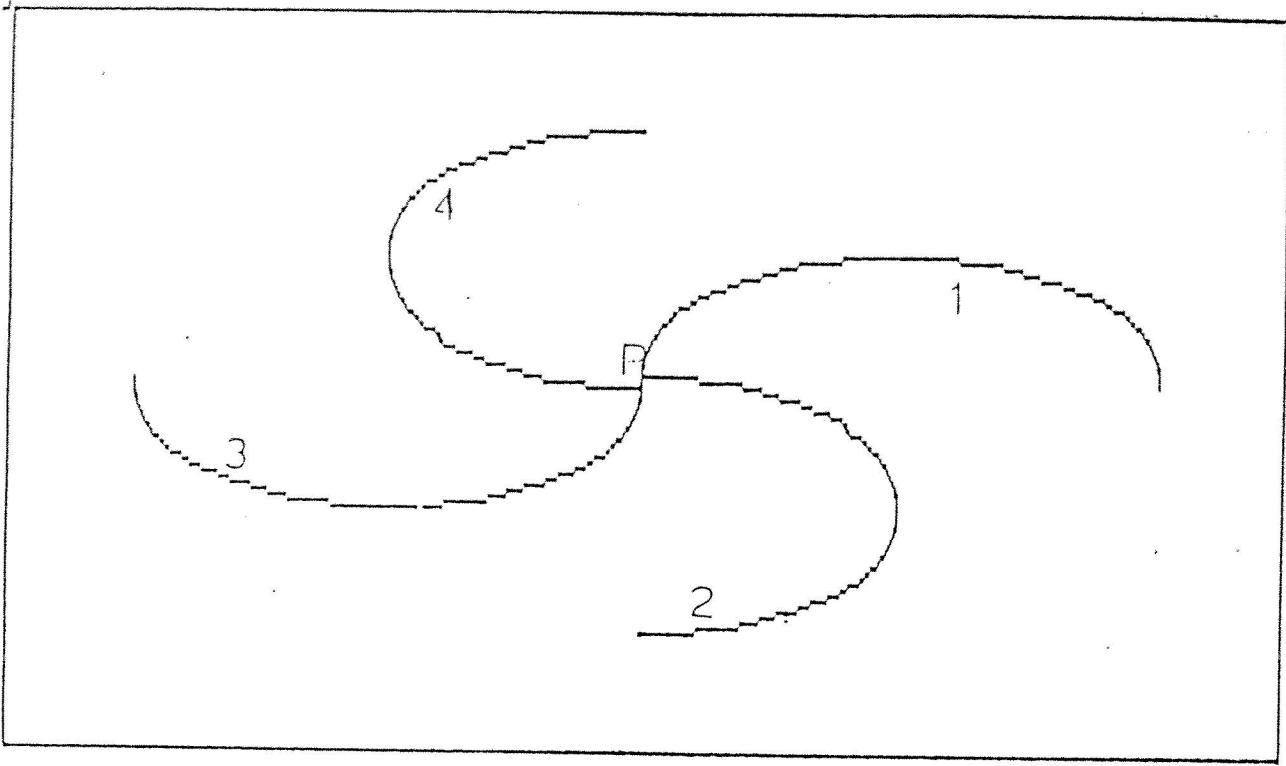
PICTURE 6.3



PICTURES 6.4 e 6.5



PICTURES 6.6 and 6.7



Pictures 6.8 and 6.9 show the possible cases when performing 3/4 of an oval, on : clockwise rotation (picture 6.8)
c.clockwise rotation (picture 6.9)

Point P is assumed as starting point.

With A is meant the semiaxis on X, with B the semiaxis on Y.

In the arcs indicated with (1), the distance between the starting point and the end point equals to :

+ A , on X
- B , on Y.

On the arcs indicated with (2), the distance between the starting point and the end point equals to :

+ A , on X
+ B , on Y.

In the arcs indicated with (3), the distance between the starting point and the end point equals to :

- A , su X
+ B , su Y.

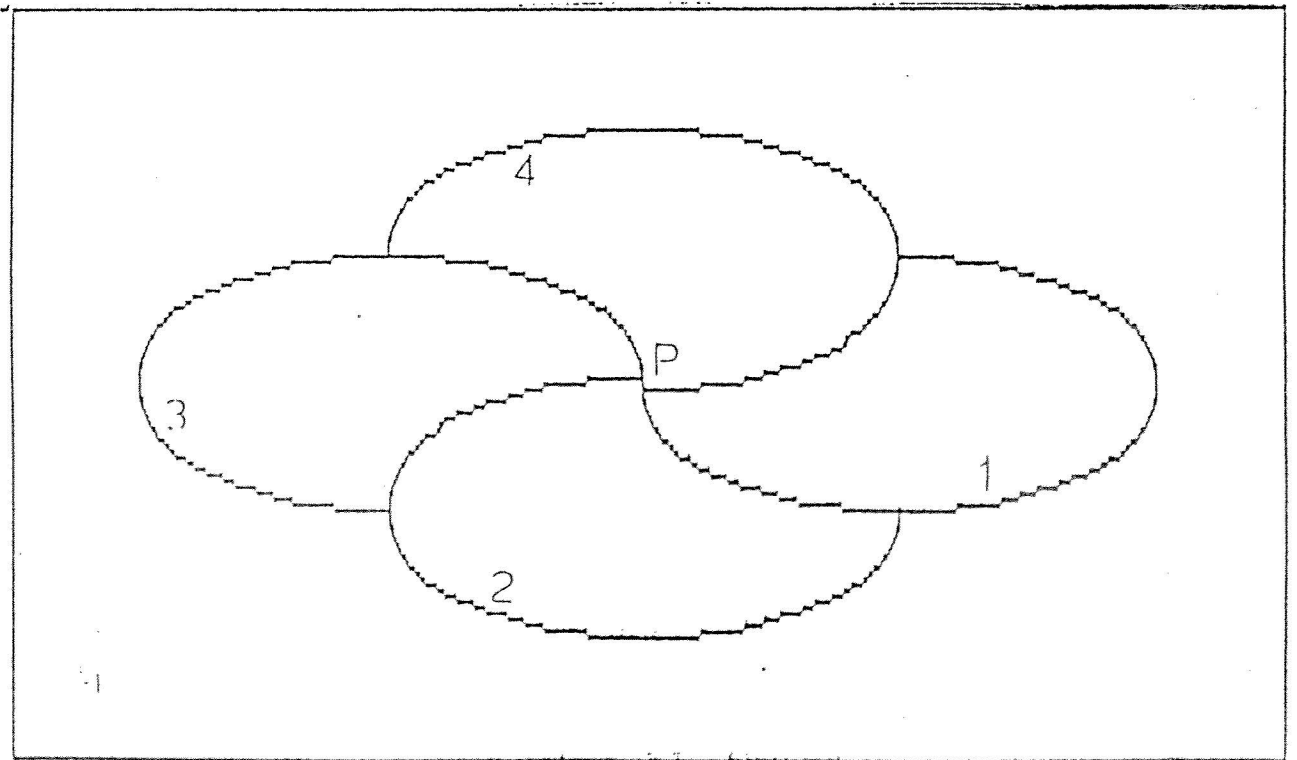
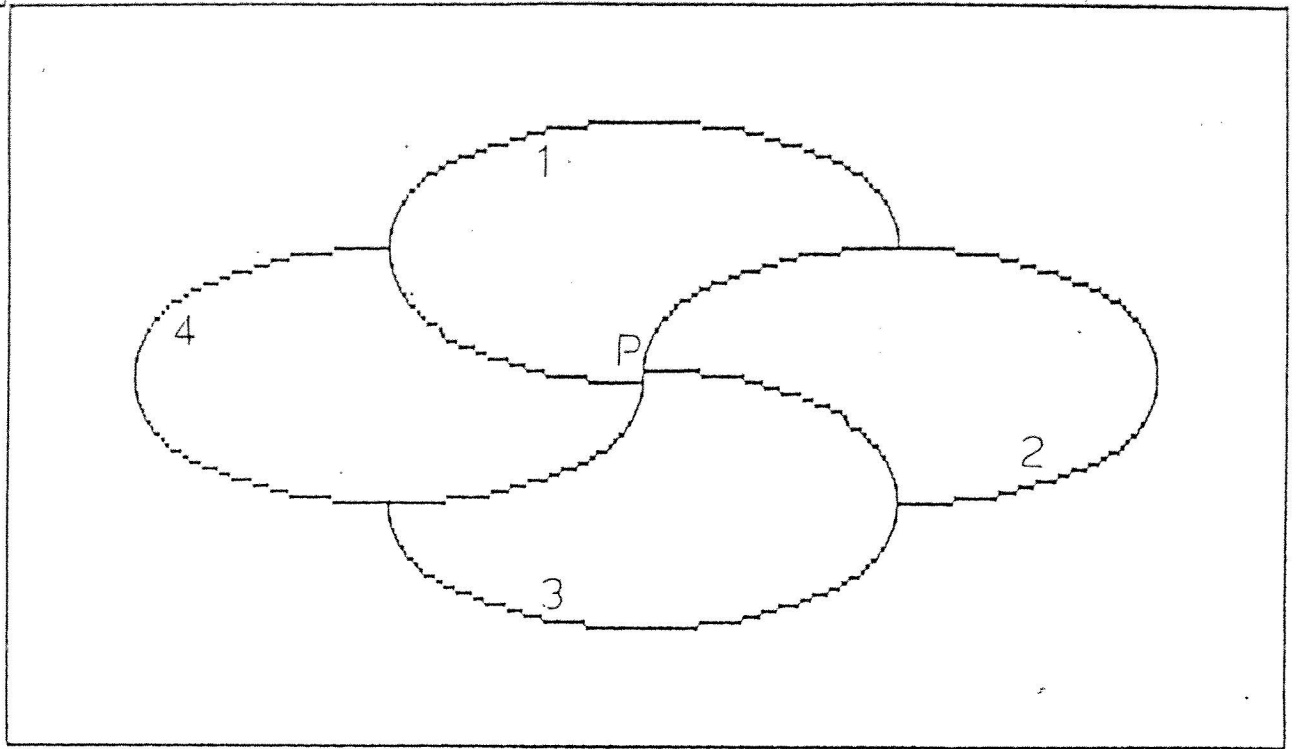
On the arcs indicated with (4), the distance between the starting point and the end point equals to :

- A , su X
- B , su Y.

It is clear that the executions of 1/4 and 3/4 of oval are completely equivalent except for the conditions given for the first point, the final point, the direction of rotation, values of semiaxes, smaller radius. Therefore :

- It is performed 1/4 of an oval if there is no other specification on function G, or if it is specified "/1";
- It is performed 3/4 of an oval if it is specified "/2" or "/3" or "/4".

PICTURES 6.8 e 6.9



REMARKS ABOUT THE USE OF MILLING COMMANDS

REMARK

Function P as selected should be valid on the interpolation line - linear, circular, helicoidal or of execution of an oval - and it should be congruent with the programmed displacements.

REMARK

As first line of any milling operation should be entered the function G10 (mill lowering).

Example :

```
G10 X xxxx.xxxx Y yyyy.yyyy Z zzzz.zzzz [VXnn] [VZnn] [VMnn]  
[R±nn] [Tnn]
```

The functions "velocity" are optional and if they miss, the ones specified in Parameters are assumed.

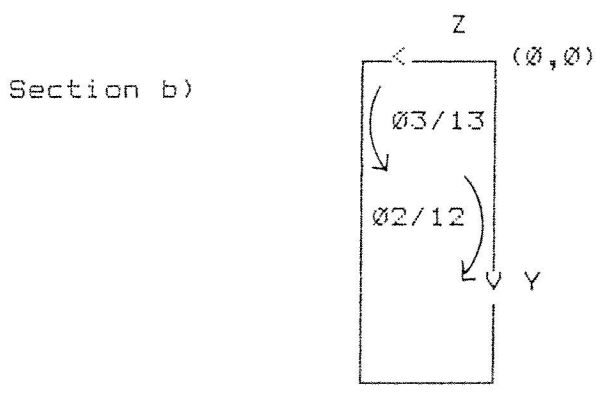
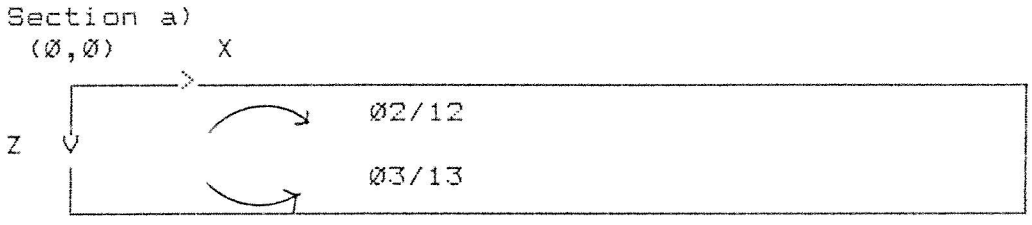
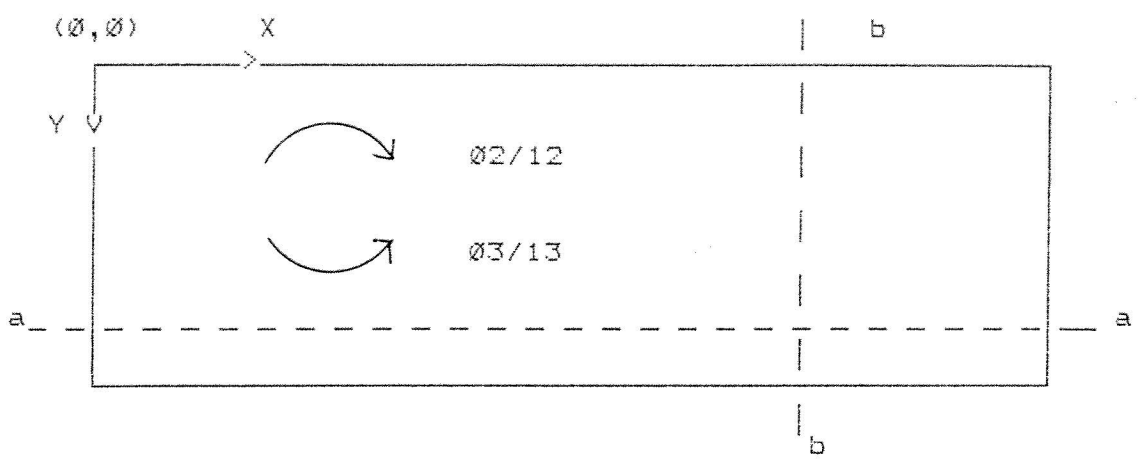
The "mill radius" function is also optional and if the milling is not to be corrected do not enter the parameter R ± nn .

For more informations, see the paragraph concerning the mill's radius.

Function "Tool" is also optional.

It is maybe opportune to clarify what is meant here for clockwise and c.clockwise rotation direction on every selectable plane.

In the next picture, are particularly set in evidence the starting points (origins) on the panel of the three axes X, Y, Z, and the rotation direction on the three interpolation planes, for convenience drawn on the upper part of the panel (plane XY) and on two opportune sections (planes XZ and YZ).



REMARK

An aspect particularly important concerns the programming of circular interpolations values :
the interpolation start and end points co-ordinated on the interpolation center. It is important to recall that as circular interpolations are to be considered also the ones programmed on the facilities available (ovals, tangent arc, ...).
At the end, a circular interpolation can, infact, always be reduced (in order to perform it) to a circumference arc defined upon :

- The arc's starting point (position on the co-ordinates X and Y, considering XY as work plane).
- The arc's end point (same as above)
- Circumference center (same as above).

On these informations a few checkings are performed in order to evaluate the correctness of the circular interpolation layout.
The checkings performed are here listed and all possible faults are displayed under a single signal, number 46 :

WRONG INTERPOLATION COMMAND

- First, are checked the radius on the interpolation's starting point (Ri) and the radius of the interpolation's end point (Ru):
Ri = distance between the start point and the center
Ru = distance between the end point and the center
- The values of the two radii can differ for a max. of $\emptyset.05$ mm included, other way the error will be displayed.
- Each of the two radii should be equal to, or longer than, $\emptyset.2$ mm (two tenths of a millimeter), other way the error will be displayed.

REMARK

A detail to be precised concerns the definition of "coincident points". Consider two points A and B, defined upon the co-ordinates:
(Xa , Ya , Za) for point A
(Xb , Yb , Zb) for point B.

The two points are considered "coincident" if the values :

Xa - Xb

Ya - Yb

Za - Zb , in their absolute value (= with positive sign), result each inferior to the resolution of the axis related to the said co-ordinate (resolution on axis X for what concerns the co-ordinates in x, and so on).

Remember besides that all interpolating axes should be defined on the same resolution.

The notion of "coincidence" between points on all three co-ordinates or on some of them it is widely used when examining milling functions, and to the term "coincident" should be given the above mentioned meaning.

Error number	Message and description
10	<p>ERROR ON OVAL : RADIUS >= SEMIAXES OR =0</p> <ul style="list-style-type: none"> . On function G14/G15 with radius (specified in A) of value major than the two semiaxes, equal to one of the two semiaxes, void, or not entered at all. . On function G14/G15 with one (or both) semiaxes voids.
11	<p>ERROR ON OVAL : CHANGE VALUE ON AXIS Z</p> <ul style="list-style-type: none"> . On function G14/G15, with variation on axis Z example : G10 X100 Y100 Z10 G14 Z14
12	<p>ERROR ON OVAL : END POINT NOT VALID</p> <ul style="list-style-type: none"> . On function G14/G15, if the final point (specified on functions X and Y) does not result quadrant point of the oval, as reckoned from the values of the two semiaxes. The end point as programmed is not, in short, coincident with the end point calculated on the basis of the two semiaxes.
29	<p>ROTATION CODE ON G NOT VALID (Gnn/.)</p> <ul style="list-style-type: none"> . On function G14/G15, with the numerical field related to the "/" of value not admitted. . If "/" is used improperly with a function G.
44	<p>WRONG TOOL</p> <ul style="list-style-type: none"> . On line G10 of tool lowering, with explicit selection of a tool not of the J type.

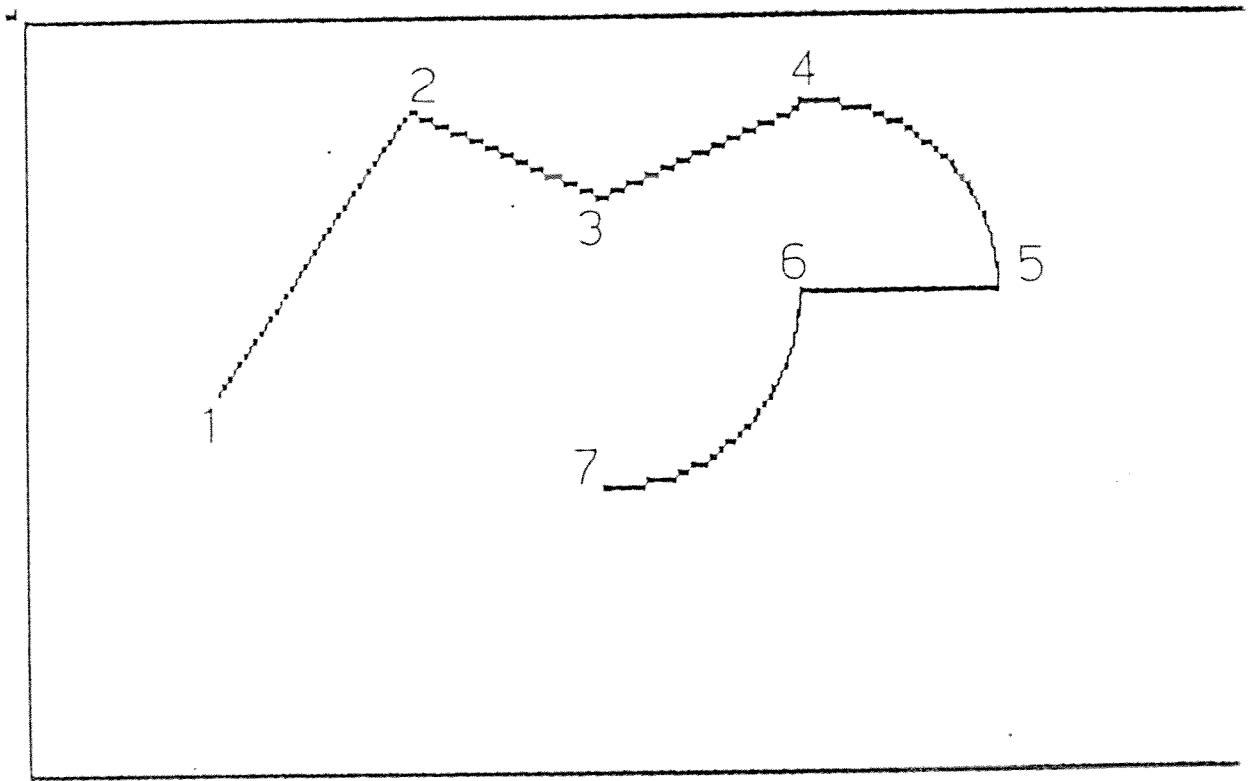
EXAMPLES OF MILLING PROGRAMS

EXAMPLE n. 1 (Picture 6.10)

```
1. G10 X100 Y200 Z5
2. X200 Y50 Z10 PXYZ
3. X300 Y100 PXY
4. X400 Y50 Z5 PXYZ
5. G12 X500 Y150 I400 J150 PXY
6. G10 X400 Z10 PXZ
7. G02 X300 Y250 I300 J150 Z15 PXYZ
```

Example concerning a cycle of interpolations with plane variation. On lines 2, 3, 4 are performed linear milling operations. Function G10 is not repeated, as it spreads automatically. Similarly, the values are specified only in the axes to be shifted. On line 5 it is programmed a clockwise circular interpolation on plane XY : function I relates with axis X, function J with axis Y. On line 6 it is programmed a function of linear interpolation on plane XZ. On line 7 it is programmed a function of helicoidal interpolation (plane XYZ) with clockwise rotation and tool final raise. The helix'axis of advancement is in Z, with increasing development of the helix on Z .

Picture 6.10

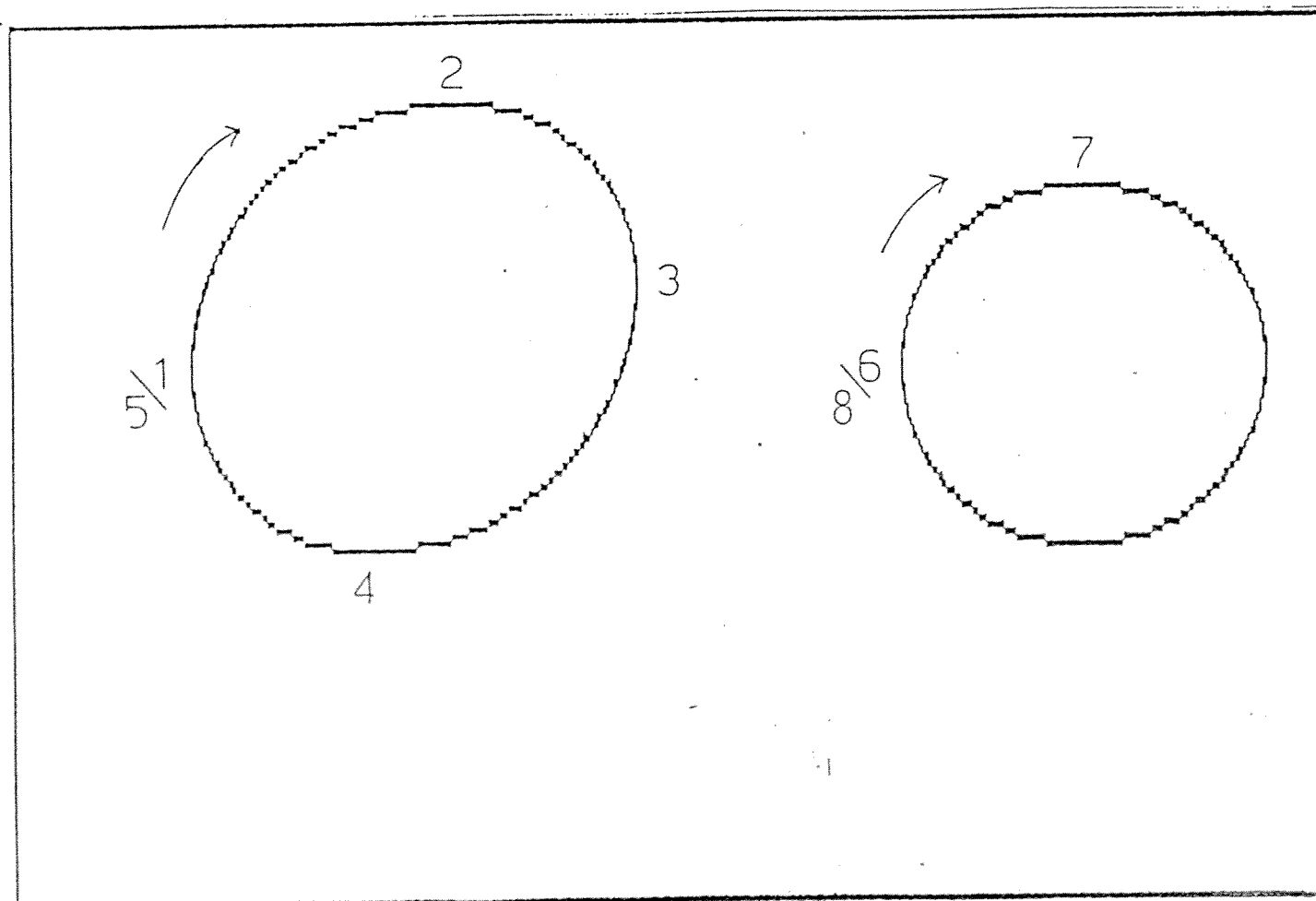


EXAMPLE n. 2 (Picture 6.11)

1. G10 X100 Y200 Z5 PXYZ
2. G12 X250 Y50 I250 J200 Z10
3. X350 Y150 I250 J150 Z5
4. X200 Y300 I200 J150 Z10
5. G02 X100 Y200 I200 J200 Z5
6. G10 X500 Y200 Z5
7. G12 X600 Y100 I600 J200 Z10
8. G02 X500 Y200 I600 J200 PXY

Example concerning a cycle of interpolations with plane variation. Two closed figures in interpolation are programmed. The first figure (lines from 1 to 5) is entirely performed in PXYZ with every operation in helicoidal interpolation. The second figure draws a helicoidal interpolation (line 7) and a semicircle on plane XY

Picture 6.11



EXAMPLE n. 3

(Picture 6.3)

1. G10 X500 Y300 Z10
2. G14 I200 J120 A70
3. G14/2 I200 J120 A70
4. G14/3 I200 J120 A70
5. G14/4 I200 J120 A70
6. G00

The program realizes the figure 6.3 previously reported. The 4 whole ovals are programmed consecutively, with tool final raise. The values specified for the semiaxes and for function A keep unchanged in the four ovals, but they have anyway to be specified on each program line : infact no automatic spreading is performed, as previously clarified. The tool final raise is obtained through function G00, without specifying any value.

EXAMPLE n. 4

(Picture 6.4)

```
1. G10 X500 Y300 Z10
2. G14 X700 Y200 I200 J100 A70
3. G00
4. G10 X500 Y300
5. G14 X700 Y400 I200 J100 A70
6. G00
7. G10 X500 Y300
8. G14 X300 Y400 I200 J100 A70
9. G00
10. G10 X500 Y300
11. G14 X300 Y200 I200 J100 A70
12. G00
```

The program realizes the figure 6.4 .
The 4 ovals' arcs are programmed on the same starting point.
The values specified for the semiaxes and for function A keep unchanged for the 4 arcs. Function G14 assigns the rotation direction (in this case : clockwise) .
The tool final raise is obtained with function G00, without specifying any value, per single arc.
Always referring to picture 6.4, are programmed in sequence the arcs number 1, 2, 3, 4.

EXAMPLE n. 5

(Picture 6.5)

```
1. G10 X500 Y300 Z10
2. G15 X700 Y200 I200 J100 A70
3. G00
4. G10 X500 Y300
5. G15 X700 Y400 I200 J100 A70
6. G00
7. G10 X500 Y300
8. G15 X300 Y400 I200 J100 A70
9. G00
10. G10 X500 Y300
11. G15 X300 Y200 I200 J100 A70
12. G00
```

The program realizes the picture 6.5 previously reported.
Compared with the text of the earlier example, the execution is on function G15 instead of G14 : this changes the rotation direction from clockwise to c. clockwise.
Always referring to picture 6.5, are programmed in sequence the arcs 1, 2, 3, 4 .

EXAMPLE n. 6

(Picture 6.6)

```
1. G10 X500 Y300 Z10
2. G14 X900 I200 J100 A70
3. G00
4. G10 X500 Y300
5. G14 Y500 I200 J100 A70
6. G00
7. G10 X500 Y300
8. G14 X100 Y300 I200 J100 A70
9. G00
10. G10 X500 Y300
11. G14 X500 Y100 I200 J100 A70
12. G00
```

The program realizes the picture 6.6 earlier reported. Function G14 assigns a clockwise rotation. Always referring to picture 6.6, are programmed in sequence the arcs number 1, 2, 3, 4 .

EXAMPLE n. 7

(Picture 6.7)

```
1. G10 X500 Y300 Z10
2. G15 X900 I200 J100 A70
3. G00
4. G10 X500 Y300
5. G15 Y500 I200 J100 A70
6. G00
7. G10 X500 Y300
8. G15 X100 Y300 I200 J100 A70
9. G00
10. G10 X500 Y300
11. G15 X500 Y100 I200 J100 A70
12. G00
```

The program realizes the picture 6.7 previously reported. Always referring to picture 6.7, are programmed in sequence the arcs number 1, 2, 3, 4 .

EXAMPLE n. 8

(Picture 6.8)

1. G10 X500 Y300 Z10
2. G14/3 X700 Y200 I200 J100 A70
3. G00
4. G10 X500 Y300
5. G14/3 X700 Y400 I200 J100 A70
6. G00
7. G10 X500 Y300
8. G14/3 X300 Y400 I200 J100 A70
9. G00
10. G10 X500 Y300
11. G14/3 X300 Y200 I200 J100 A70
12. G00

The program realizes the picture 6.8 previously reported. Function G14 assigns the clockwise rotation; the specification of "/3" selects the execution on 3/4 of ovals. Always referring to picture 6.8, are programmed in sequence the arcs number 1, 2, 3, 4 .

EXAMPLE n. 9

(Picture 6.9)

1. G10 X500 Y300 Z10
2. G15/3 X700 Y200 I200 J100 A70
3. G00
4. G10 X500 Y300
5. G15/3 X700 Y400 I200 J100 A70
6. G00
7. G10 X500 Y300
8. G15/3 X300 Y400 I200 J100 A70
9. G00
10. G10 X500 Y300
11. G15/3 X300 Y200 I200 J100 A70
12. G00

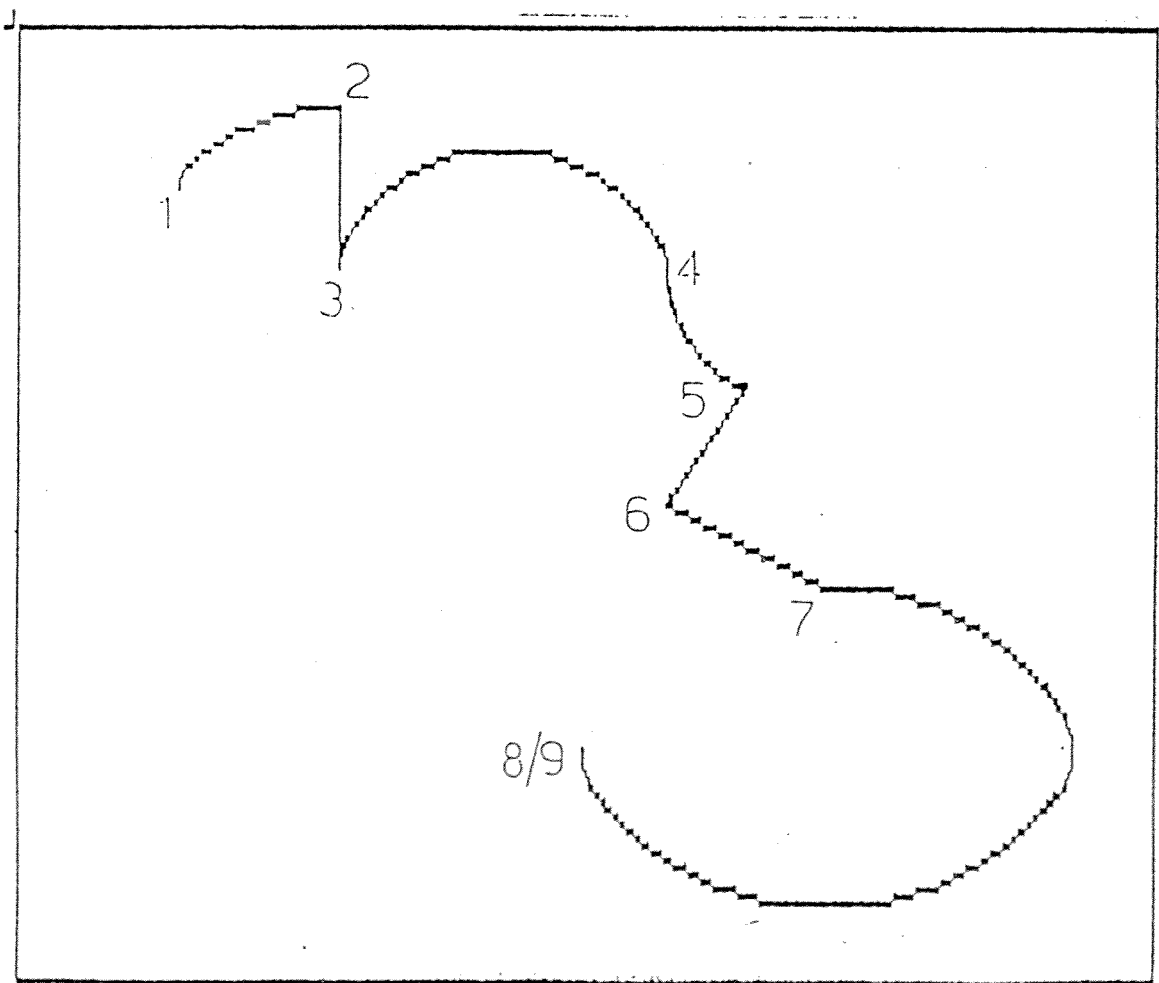
The program realizes the picture 6.9 previously reported. Function G15 assigns the c.clockwise rotation ; the specification of "/3" selects the execution of 3/4 of ovals. With reference to picture 6.9, are programmed in sequence the arcs number 1, 2, 3, 4 .

EXAMPLE n.10

(Picture 6.12)

1. G10 X100 Y100 Z10
2. G14 X200 Y50 I100 J50 A20 ;arc on 1/4 of oval
3. G10 Y150
4. G14 X400 I100 J75 A30 ;arc on 1/2 oval
5. G15 X450 Y225 I50 J75 A15 ;arc on 1/4 of oval
6. G10 X400 Y300
7. X500 Y350
8. G14/3 X350 Y450 I150 J100 A50 ;arc on 3/4 of oval
9. G00

Picture 6.12

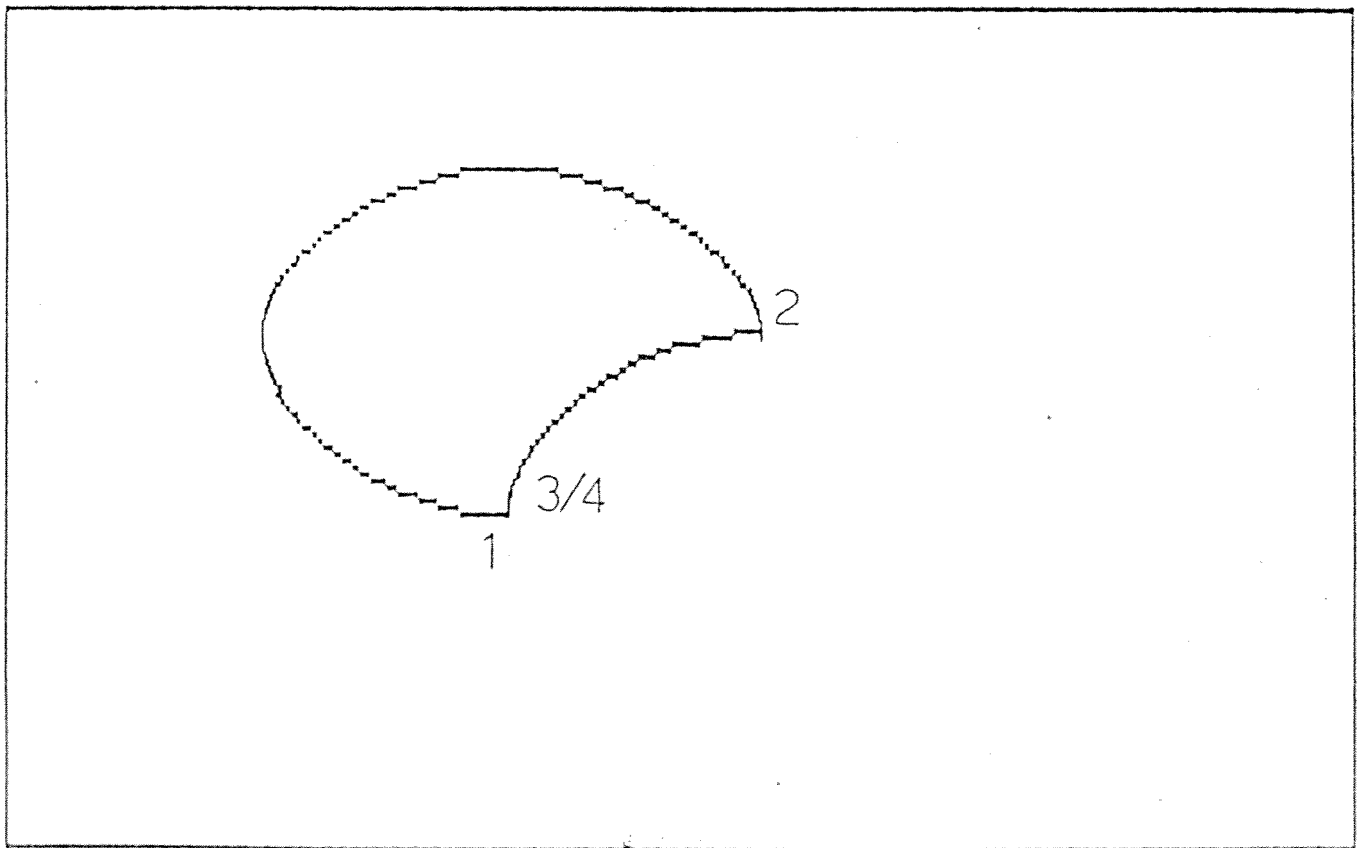


EXAMPLE n.11

(Picture 6.13)

1. G10 X300 Y300 Z10
2. G14/3 X450 Y200 I150 J100 A50 ;arc on 3/4 of oval
3. G15 X300 Y300 I150 J100 A40 ;arc on 1/4 of oval
4. G00

Picture 6.13



SIMPLIFIED PROGRAMS ON ARCS

In the previous chapters, it has been illustrated how to program a circumference arc on the co-ordinates of the end point and of the center.

In this present section, other possibilities of programming a circumference arc will be examined :

1. How to program a arc of a circle upon radius assignement.
2. How to program an arc of a circle passing through three non-aligned points.
3. How to program an arc of a circle upon assignment of the initial tangent.
4. How to program a connecting arc.

However, when elaborating and saving the program, any interpolating function falls into the case already examined concerning the center, starting and end points data.

PROGRAMMING AN ARC ON THE RADIUS

It is possible to program circumference arcs by defining :

1. Starting point (implicitly assumed from the previous block)
2. End point (explicit)
3. Rotation direction : clockwise or c.clockwise (explicit)
4. Circumference radius (explicit)

Syntax : Ggg/ Xxxxx.x Yyyy.y Iiii.i [Vvv.v]

where Ggg selects the function of circular interpolation :
G12,G02 for clockwise rotation
G13,G03 for c.clockwise rotation

/ the character "/" means to select the arc programming on the radius value.

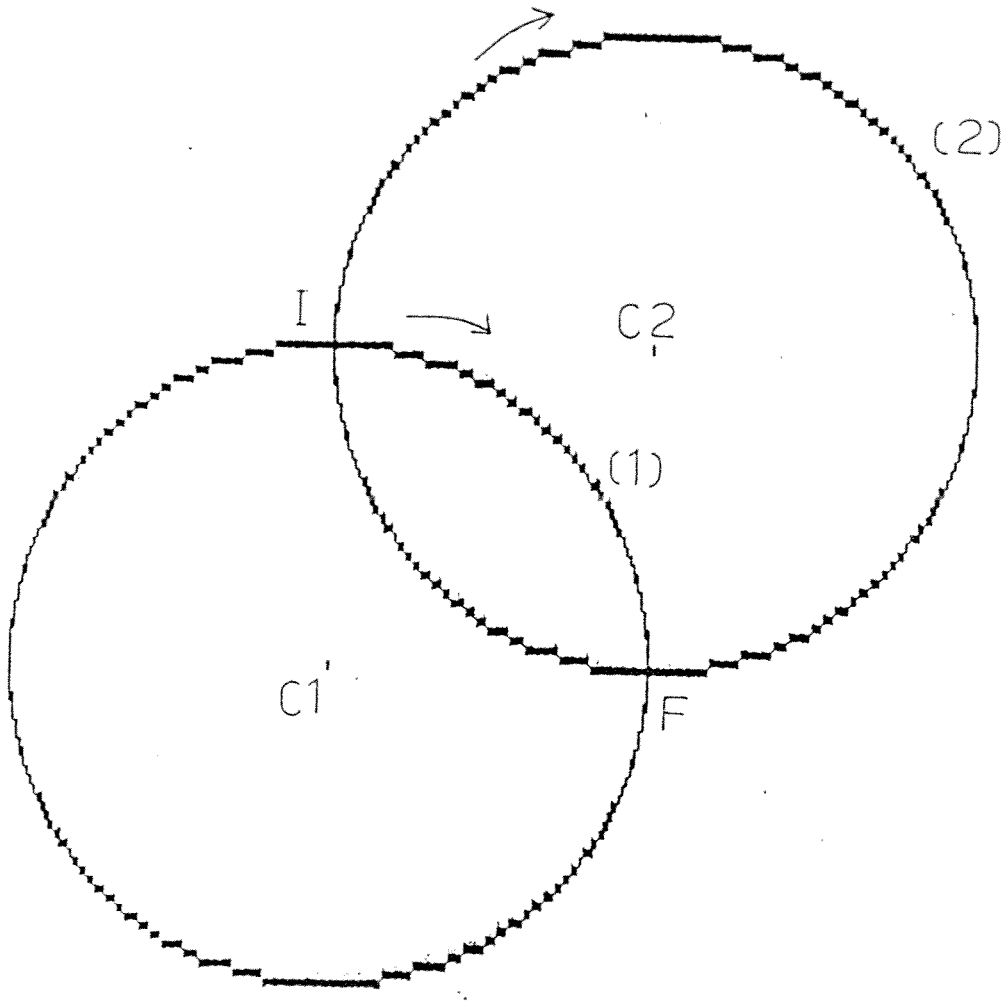
Xxxxx.x co-ordinate x of end point
Yyyy.y co-ordinate y of end point
Iiii.i In function I is given the radius' value.
The radius'sign is significant (sign + should be omitted) :
positive radius if the corner described by the arc is less than or equal to 180° ,
negative radius if the corner described by the arc is major than 180° .

The interpolation plane should be XY.

Picture 7.1 sets in evidence the problem under a geometrical point of view :

- . I indicates the starting point
- . F indicates the end point
- . The information concerning the radius' value allows to locate the two circles :
circle with center in C1
circle with center in C2;
- . The information concerning the rotation direction allows to locate in (1) and (2) the traceable arcs :
with negative radius is performed arc (1)
with positive radius is performed arc (2).

Picture 7.1

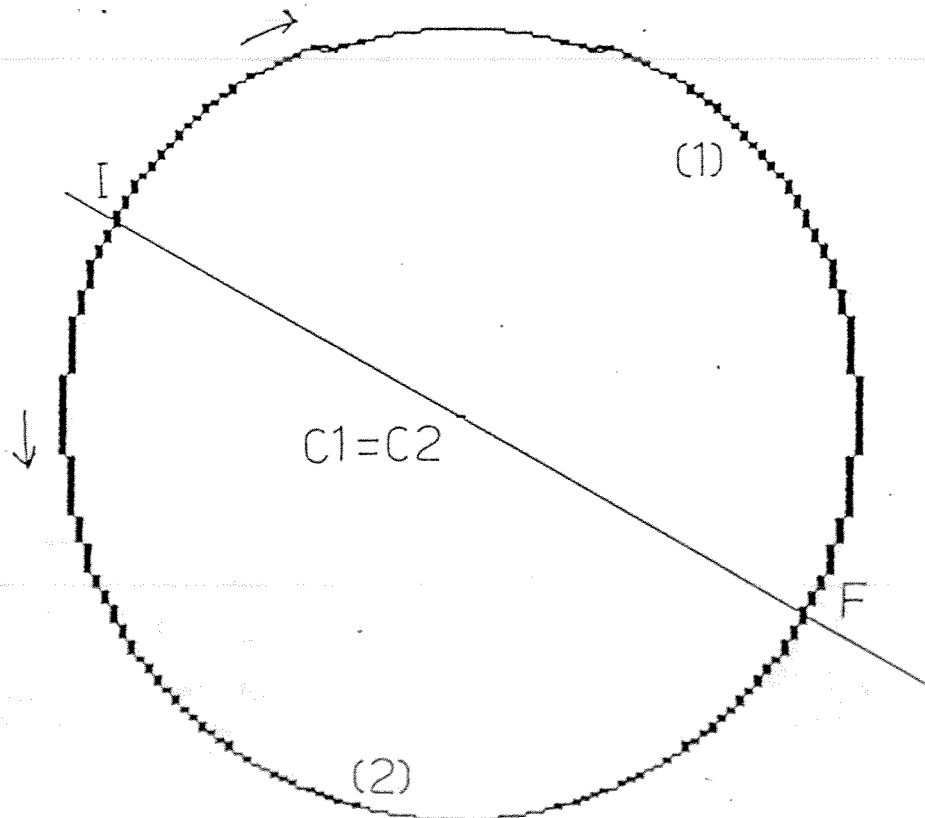


A particular situation happens if the two centers are coincident : the center is aligned with points I and F (the distance between the points I and F is equal to the value : $\text{diameter}=2*\text{radius}$). The case is defined in picture 7.2. In this case do not exist two different arcs, one per each rotation direction assigned:

Arc (1) of picture 7.2 is performed with clockwise rotation.
Arc (2) of picture 7.2 is performed with c.clockwise rotation.
The corner described by each of the two arcs is 180° .

On the other hand the problem is not solvable if the circle's diameter is smaller than the distance between the two points I and F.

Picture 7.2



If the start and the end point coincide, the program has the following syntax :

Ggg/n Iiiii.i

(see picture 7.3) where :

- The co-ordinates on the final point are spread by the previous instruction (or are defined the same).
- On field I is given the radius' value. The radius'sign is, in this case, influential.
- To function G is added a numerical specification, on "n"; values admitted : from 1 to 4 :

n=1 perform circle (1)
n=2 performs circle (2)
n=3 performs circle (3)
n=4 performs circle (4).

If no value is specified on n (that means, G13/ Iiiii.i), it will be performed per default the circle (1).

On the four circles, the distance between the centers and the processing start/end points, equals to :

in circle (1) : Ø , on x
+radius , on y

in circle (2) : -radius , on x
Ø , on y

in circle (3) : Ø , on x
-radius , on y

in circle (4) : +radius , on x
Ø , on y.

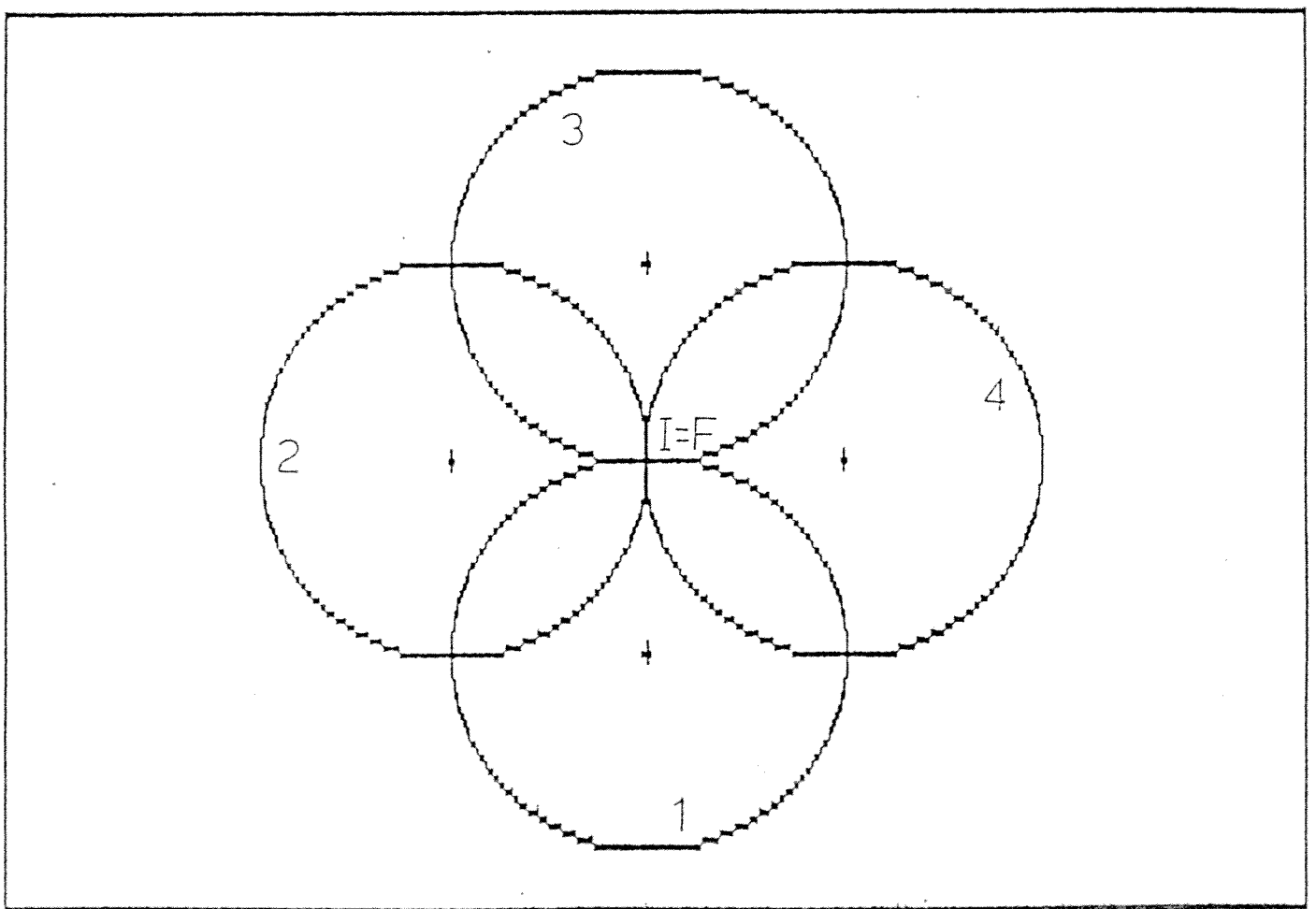
If it is necessary to perform a full circle starting from "any" point - that means, not a quadrant point - use the program on the center's co-ordinates .

The following program runs the processing displayed in picture 7.3

```
1. G10 X500 Y350 Z10
2. G02/1 I150 ;circle (1)
3. G10
4. G02/2 I150 ;circle (2)
5. G10
6. G02/3 I150 ;circle (3)
7. G10
8. G02/4 I150 ;circle (4)
```

On the program reported, every function subjected to automatic spreading has been omitted (values on X and Y).

Picture 7.3



PROGRAMMING AN ARC PASSING THROUGH THREE POINTS

It is possible to program circumference arcs by defining :

1. Starting point (implicit from previous block)
2. End point (explicit)
3. Middle point (explicit)
4. Clockwise or c.clockwise rotation (explicit).

syntax : Ggg/5 Xxxxx.x Yyyyy.y Iiiii.i Jjjjj.j [Vvv.v]

where Ggg selects the circular interpolation function
: G12,G02 on clockwise rotation
G13,G03 on c.clockwise rotation;

/5 specification on "/5" means to select the arc program of passage through three points.

Xxxxx.x Yyyyy.y co-ordinates x and y of the end point;
Iiiii.i Jjjjj.j co-ordinates x and y of the middle point

The plane of interpolation should be XY.
The three points should not be aligned.

Picture 7.4 shows the geometry :

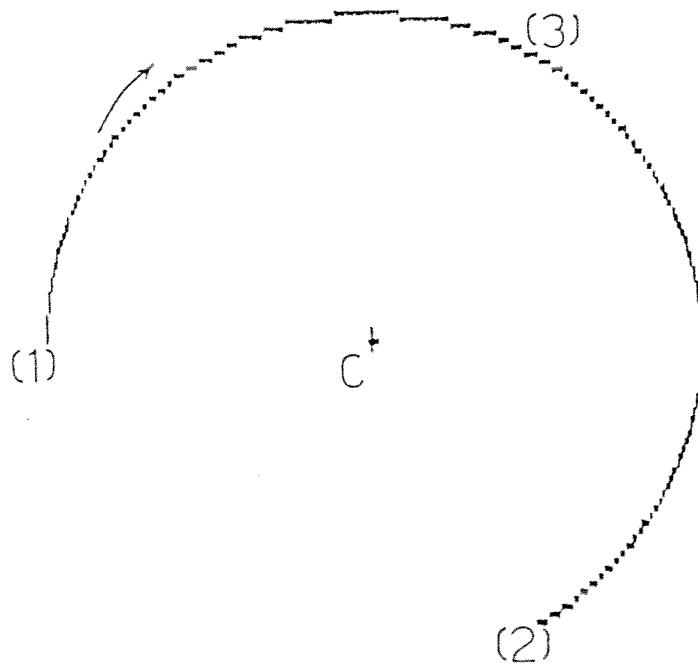
- (1) is the starting point
- (2) is the end point
- (3) is the middle point.

The three point assignement allows to point out the co-ordinates of the center C.

NOTE

The passage through the point defined as "middle" does not necessarily have to happen . In the example of picture 7.4 :
- the clockwise execution implies passing through point (3);
- the c.clockwise execution does not pass through point (3).

PICTURE 7.4



ARC'S SIMPLIFIED PROGRAMMING ON A TANGENT

It is possible to program circumference arcs by defining :

1. Starting point (implicit from previous block)
2. End point (explicit)
3. Initial tangent (implicit from previous blocks)
4. Clockwise or c.clockwise rotation (explicit).

syntax : Ggg Xxxxx.x Yyyy.y AØ [Vvv.v]

where Ggg selects the circular interpolation function
G12,GØ2 clockwise rotation
G13,GØ3 c.clockwise rotation

AØ value Ø connected to function A allows
to program the arc on the initial tangent.

Xxxxx.x Yyyy.y co-ordinates x and y of final point.

The interpolation plane should be XY : that means that both the tangent straight line and the arc itself should be defined on plane XY.

The following examples show a few cases of arcs programmed upon assignment of the tangent on the starting point.

Example n.1 (Picture 7.5)

```
G10 X(1) Y(1) ;co-ordinates of point 1
G10 X(2) Y(2) ;co-ordinates of point 2
G02 X(3) Y(3) A0 ;co-ordinates of point 3
```

The tangent on the starting point is defined by the straight line through the points (1) and (2).

Example n.2 (Picture 7.6)

```
G10 X(1) Y(1) ;co-ordinates of point 1
G10 X(2) Y(2) ;co-ordinates of point 2
G12 X(3) Y(3) A0 ;co-ordinates of point 3
G03 X(4) Y(4) A0 ;co-ordinates of point 4
```

Compared to the previous example, there is a line more of circular milling, with the second arc's tangent defined upon the straight line tangent to the first arc (arc through the points 2 and 3) in point 3.

Therefore : the arc's initial tangent can be defined indifferently on a linear or circular milling. Meaning with N the line number with the arc programmed on the initial tangent, a correct program will fall into one of the following cases :

line

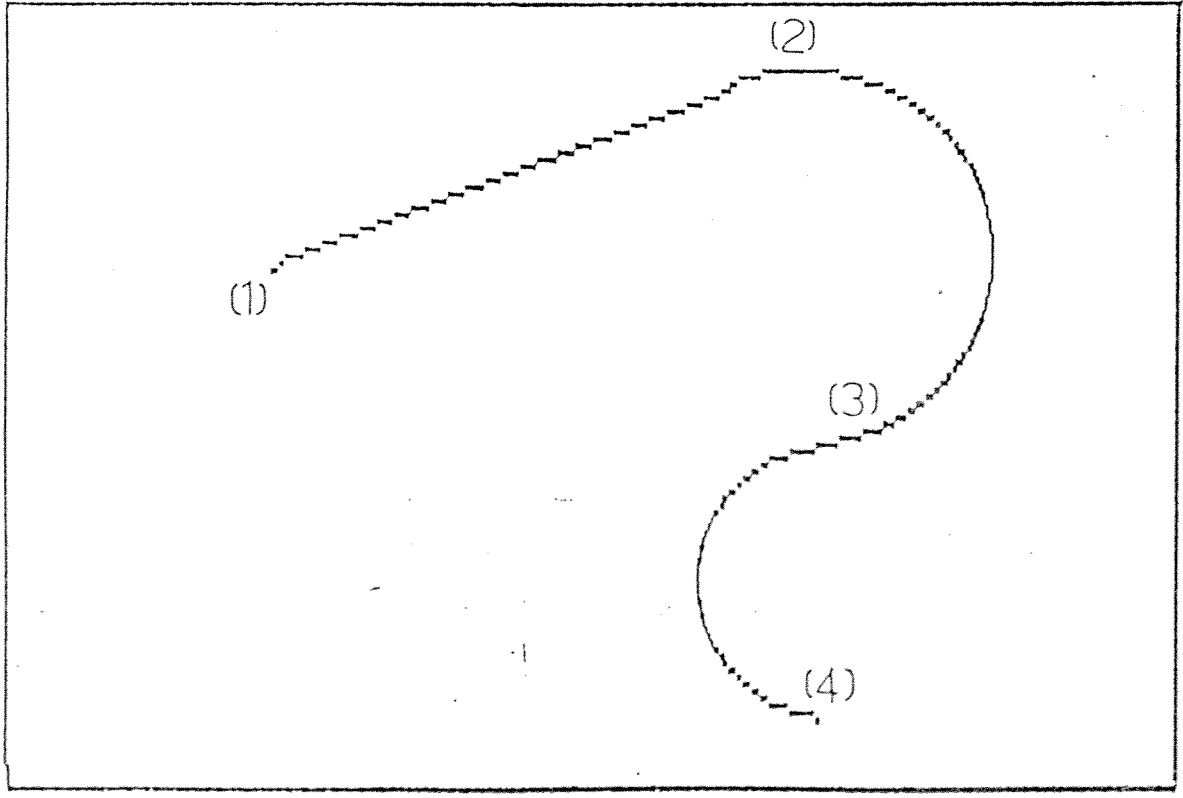
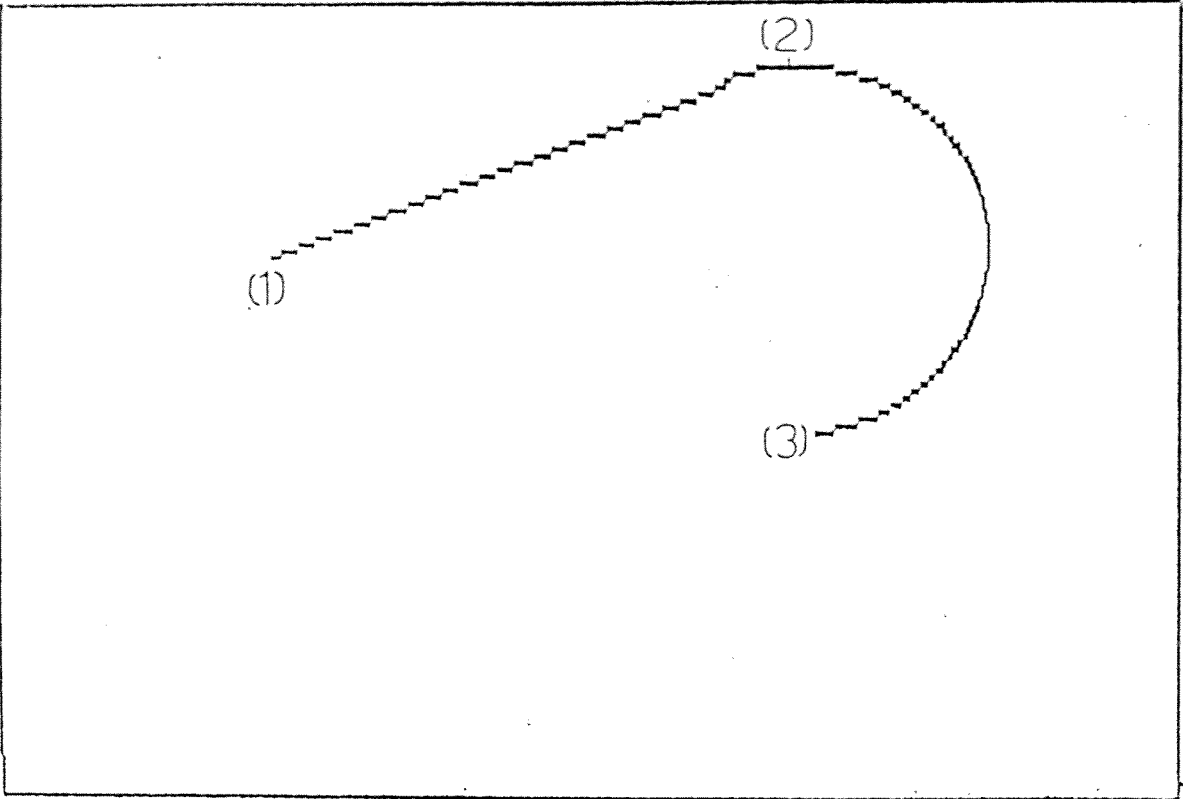
(N-2)-nth [G12/G13 with circular interpolation however defined
 G10 linear milling or function "tool lowering"

(N-1)-nth G10 linear milling
(N)- nth G12/13/02/03 X.... Y.... A0

(N-2)-nth [G12/G13 with circular interpolation however defined
 G10 linear milling or function "tool lowering"

(N-1)-nth G12/G13 Circular interpolation however defined.
(N)-nth G12/13/02/03 X.... Y.... A0

Pictures 7.5 e 7.6



PROGRAMMING CONNECTIONS

functions G12 / G13 and A on the corner of a cut.

syntax : Ggg Xxxx.x Yyyy.y Zzz.z Aaa [VXvv] [VYvv]

where :

Aaa specifies the connection's radius. ;
aa should be greater than zero

Ggg selects the circular interpolation function :
G12 clockwise rotation
G13 c.clockwise rotation.

To program the velocity functions VX and VY is an optional. If programmed, VX specifies (as usual) the interpolation speed. If programmed, VY specifies the interpolation speed to apply to the connecting arc.

If VY has not been programmed, for the connecting arc is assumed the valid interpolation speed, programmed in VX or loaded from Parameters.

The velocity programmed for VY will be used solely for the connecting arc said, without any spreading to other interpolations.

Also function A works only in the program line where it has been specified, without possibility of any spreading.

The two segments of the cut line can be defined on different planes. However, connections upon interpolation in area XYZ are not accepted. In different cases is displayed the error :
" INVALID CONNECTION ".

The following examples clarify every possible case of programming a connection, limited to the plane XY :

Example n. 1

```
G10 X100 Y300 ; co-ordinates of point 1
G12 X300 Y100 A30 ; co-ordinates of point 2
G00 X500 Y200 ; co-ordinates of point 3
```

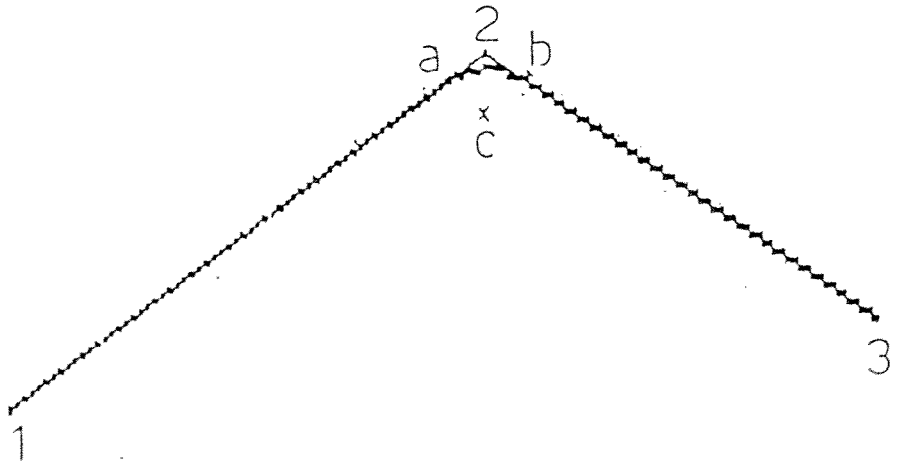
The cut line, defined on three points (1, 2, 3), is carried out with a connection of radius 30 on point 2: the milling's effective path results therefore defined on 4 points (1,a,b,3), as shown in picture 7.7 .

The following program describes the same milling path without using the parameter A to describe the connection :

```
G10 X100 Y300 ; co-ordinates of point 1
G10 X(a) Y(a) ; co-ordinates of point a
G12 X(b) Y(b) I(c) J(c) ; end point of circular
; interpolation in b, center in c.
G00 X500 Y200 ; co-ordinates of point 3
```

The usefulness of implicit programming of connecting points result clear already in this easy example : in fact, the co-ordinates of the points a, b, c, here indicated in a symbolic way, are not of immediate determination.

Picture 7.7



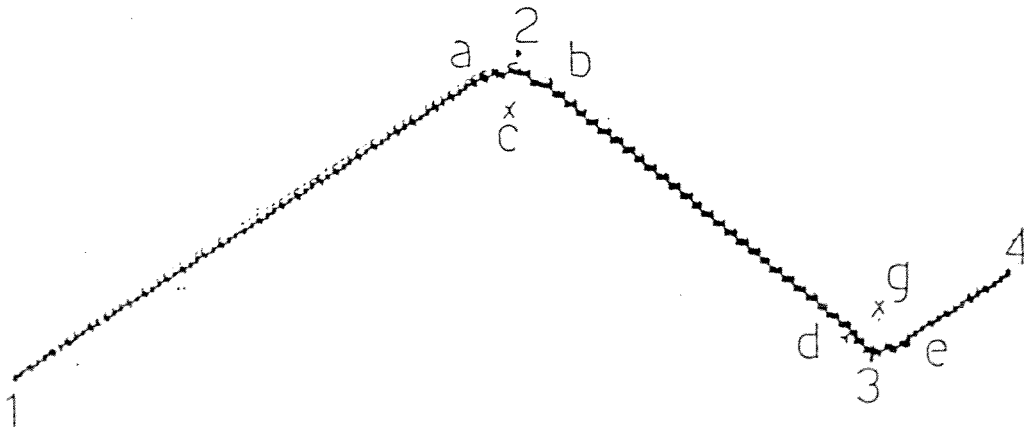
Example n. 2

```
G10 X100 Y300 ; co-ordinates of point 1
G12 X300 Y100 A30 ; co-ordinates of point 2
G13 X500 Y200 A20 ; co-ordinates of point 3
G00 X550 Y150 ; co-ordinates of point 4
```

The cut line is now defined on the 4 points 1, 2, 3, 4 with two connections: of radius 30 on point 2, and of radius 20 on point 3. The milling's effective path is described, in a completely equivalent way, in the program listed below :

```
G10 X100 Y300 ; co-ordinates of point 1
G10 X(a) Y(a) ; co-ordinate of point a
G12 X(b) Y(b) I(c) J(c) ; end point of circular
; interpolation in b, center in c.
G10 X(d) Y(d) ; co-ordinates of point d
G13 X(e) Y(e) I(g) J(g) ; end point of circular
; interpolation in e, center in g
G00 X550 Y150 ; co-ordinates of point 4.
```

Picture 7.8



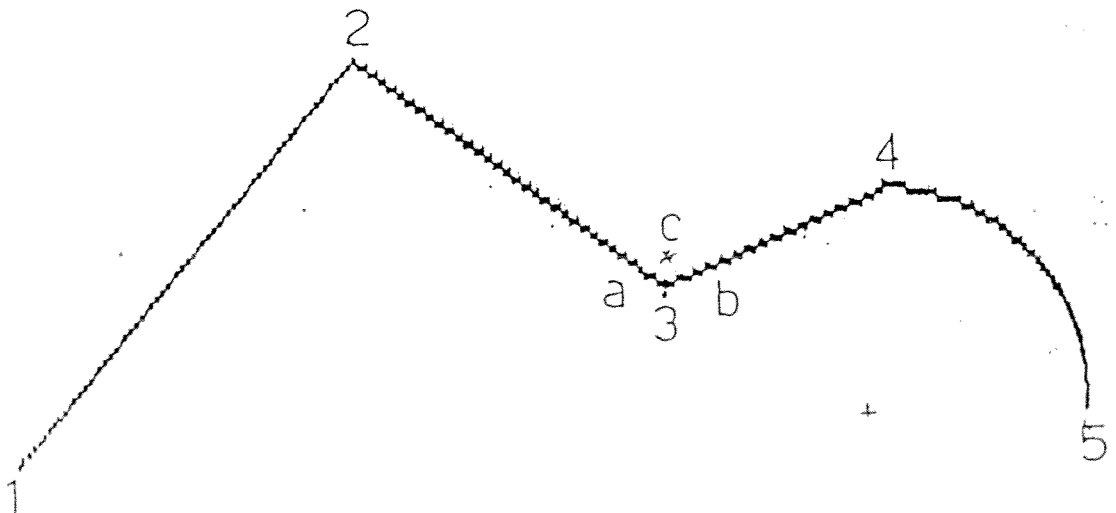
Example n. 3

```
G10 X50 Y300 ; co-ordinates of point 1
G10 X200 Y100 ; co-ordinates of point 2
G13 X350 Y200 A20 ; co-ordinates of point 3
G10 X450 Y150 ; co-ordinates of point 4
G02 X550 Y250 I450 J250 ; co-ordinates of point 5 and center
of circular interpolation.
```

Similarly to the examples above reported it is here listed the program reproducing the same milling path without connections/ simplified programming :

```
G10 X50 Y300 ; co-ordinates of point 1
G10 X200 Y100 ; co-ordinates of point 2
G10 X(a) Y(a) ; co-ordinates of point a
G13 X(b) Y(b) I(c) J(c) ; end point of circular
interpolation in b, center in c
G10 X450 Y150 ; co-ordinates of point 4
G02 X550 Y250 I450 J250 ; co-ordinates of point 5 and center
of circular interpolation.
```

Picture 7.9



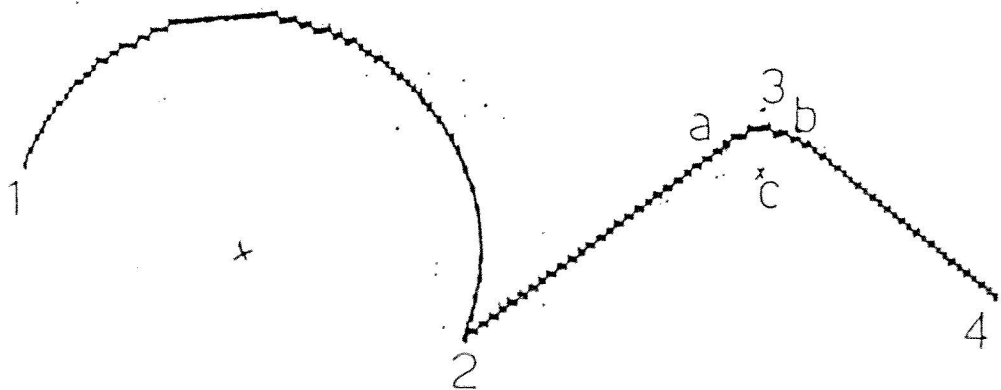
Example n. 4

```
G10 X100 Y150 ; co-ordinates of point 1
G12 X300 Y250 I200 J200 ; co-ordinates of point 2 and center
                        of circular interpolation
G12 X450 Y150 A30 ; co-ordinates of point 3
G00 X550 Y250 ; co-ordinates of point 4 .
```

The following program reproduces the same milling path without connections' simplified programming.

```
G10 X100 Y150 ; co-ordinates of point 1
G12 X300 Y250 I200 J200 ; co-ordinates of point 2 and center
                        of circular interpolation
G10 X(a) Y(a) ; co-ordinates of point a and
G12 X(b) Y(b) I(c) J(c) ; end point of circular
                        interpolation in b, center in c
G00 X550 Y250 ; co-ordinates of point 4 .
```

Picture 7.10



With the examples reported, it is now possible to define the rules to program correctly the connections :

1. A connection is localized by three points of a cut line, see example 1, with connection through the corner in 2.
2. It is anyway always possible to program connections through consecutive corners : see example 2.
3. A connecting point is defined by the corner point (X and Y), the radius (parameter A), and by the function G of rotation (G12 or G13).
4. In the line defining a connection, it is not possible to assign a function G of tool raise (ascent) like, for instance, G02.
5. The program line preceding the connection should be defined as follow :

```
G12/13  Xxxxx.xx  Yyyyy.yy  Aaa
G12/13  Xxxxx.xx  Yyyyy.yy  Iiiii.ii  Jjjjj.jj
G10     Xxxxx.xx  Yyyyy.yy
```

6. The program line following the connection should be defined in one of the here listed ways :

```
G12/13  Xxxxx.xx  Yyyyy.yy  Aaa
G10     Xxxxx.xx  Yyyyy.yy
G00     Xxxxx.xx  Yyyyy.yy .
```

- Errors of geometry

It is important to give evidence here to a few uncorrect geometrical situations possible when programming connections :

1. Points of the cut line entered aligned. The following example defines three aligned points, with inquiry of a connection in the middle point :

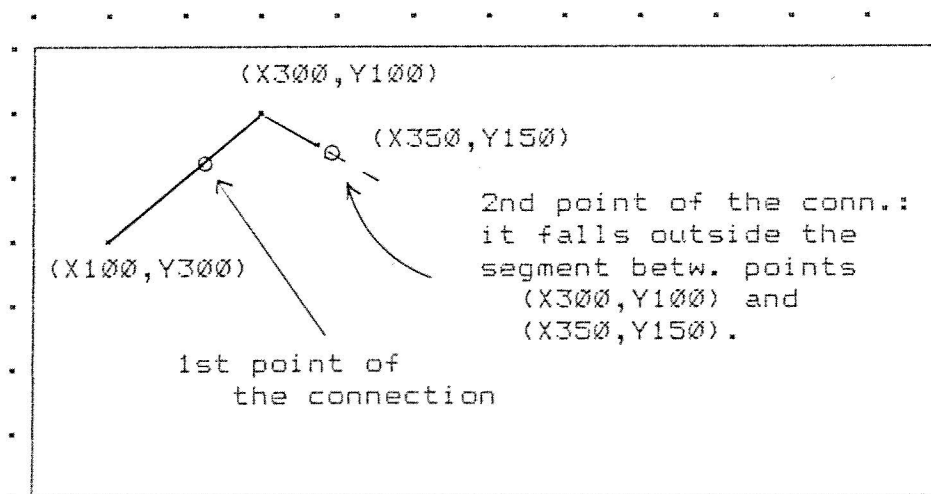
```
G10 X100 Y100
G12 X200 Y200 A20           The program will not be accepted
G00 X300 Y300
```

2. one or more points of the connection fall outside the specified points :

```
G10 X100 Y300
G12 X300 Y100 A75          The second point of the cut line
G00 X350 Y150             falls beyond the point (X350,Y150) .
                           The program will not be accepted .
```

A connection radius equals (for instance) to 70, would make the program performable.

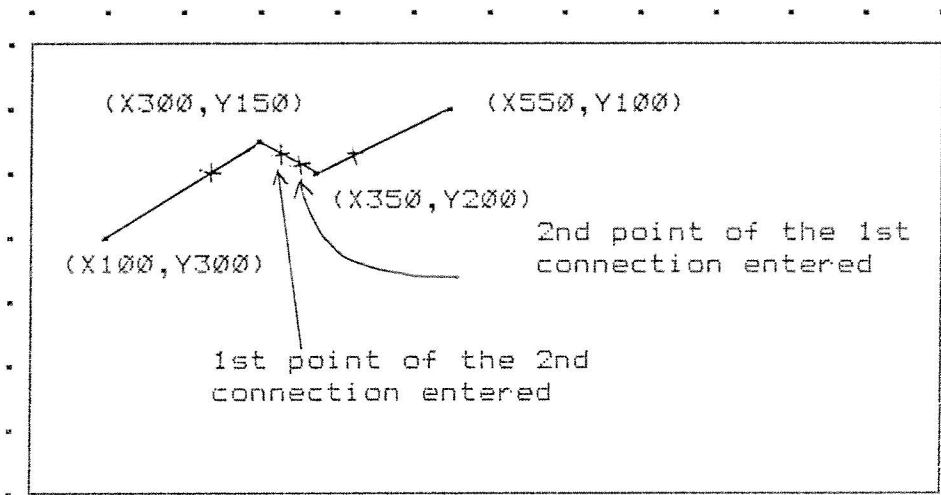
The picture represent the geometry describing the a.m. program.



3. In consecutive connections can happen situations of superimposition, on the same connecting arcs :

```
G10 X100 Y300  
G12 X300 Y150 A45  
G13 X350 Y200 A45  
G00 X550 Y100
```

The picture sets in evidence the uncorrect geometry of the program as entered. A connection radius of 40, for instance, would make the program performable.

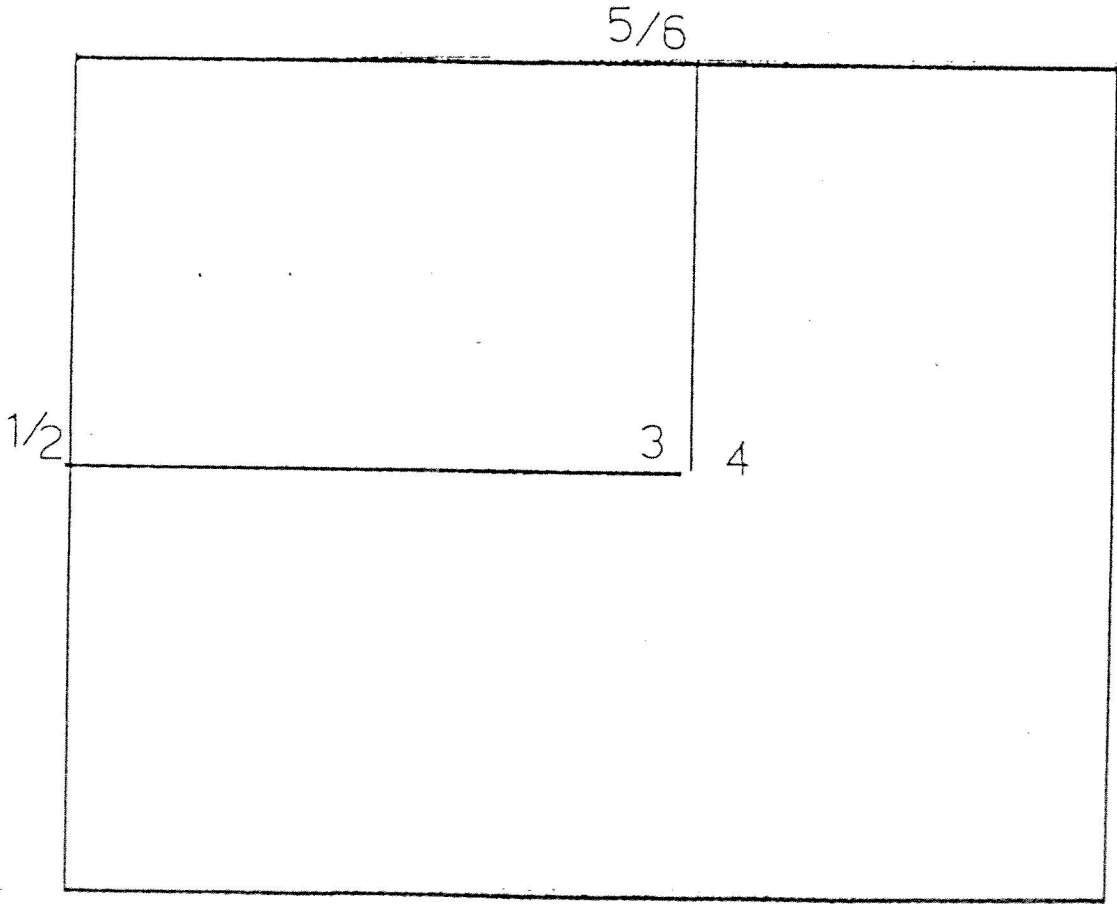


_____ simplified programs on arcs _____

Follows an example of connections programmed on planes XZ and YZ

```
G10 X-2 Y200 Z15 FXZ ; point 1 (1st point of the cut line)
G12 Z5 A10           ; point 2, corner (2nd point of the
                    ; cut line )
G00 X295            ; point 3 (3rd point of the cut line,
                    ; and tool raise )
G10 X300 Y195 PYZ  ; point 4 (1st point of the cut line)
G13 Y-2 A10        ; point 5, corner (2nd point of the
                    ; cut line )
G00 Z15            ; point 6 (3rd point of the cut line
                    ; and tool raise .
```

Picture 7.11



ERRORS DISPLAYED WHEN EDITING

Error number	Message and description
13	<p>UNCORRECT PROGRAM. OF ARC ON RADIUS VALUE</p> <p>When programming an arc on radius :</p> <ul style="list-style-type: none"> .with value less than 0.2 mm or not entered. .if the distance between the arc two outermost points is major than the diameter.
15	<p>POINTS ON LINE (ARC PROGRAM. ON 3 POINTS)</p> <ul style="list-style-type: none"> . If the three points result on the same line
14	<p>COINCIDENT POINTS IN ARC DEFINED UPON 3 POINTS</p> <ul style="list-style-type: none"> . If the three points do not differ (two or three are coincident) . If the middle point (to be defined in functions I and J) has not been inserted
19	<p>ARC ON RADIUS OR ON 3 POINTS NOT ON PLANE XY</p> <ul style="list-style-type: none"> . The value on axis Z changes, in respect to the previous line (variation superior to the resolution defined on axis Z).
58	<p>ARC PROGRAMM. ON COINCIDENTAL POINTS</p> <ul style="list-style-type: none"> . Upon definition of an arc by the initial tangent, if the arc's start or end point are coincident.
57	<p>ARC'S SIMPLIFIED PROGRAMMING NOT ALLOWED: PLANE NOT XY</p> <ul style="list-style-type: none"> . Upon definition of an arc by the initial tangent, with tangent straight line not defined on plane XY, or with the arc itself not defined on plane XY.
28	<p>VALUE ON FIED A NOT VALID</p> <ul style="list-style-type: none"> . When to function A has been assigned a negative value.
29	<p>ROTATION CODE ON G OF NOT VALID (Gnn/.)</p> <ul style="list-style-type: none"> . When on function G has been specified a value not acceptable

Error number	Message and description
25	<p>CONNECTING POINTS, OR ARC, IN LINE</p> <ul style="list-style-type: none"> . Upon program of a connection, with the 3 definition points of the cut line aligned. . Upon program of an arc through the tangent, with tangent defined on linear interpolation and arc's end point aligned with the tangent.
26	<p>CONNECTING POINTS OUTSIDE THE SEGMENT</p> <p>When programming a connection with :</p> <ul style="list-style-type: none"> . one or both connecting points falling outside the cut line's outermost points . There are consecutive connections, or situations of superimposition of the arcs themselves .
27	<p>INVALID SEQUENCE OF CONNECTIONS</p> <ul style="list-style-type: none"> . Upon program of a connection, if the programming rules on the lines before and after the connection were not followed. . Upon program of an arc through the tangent, if the programming rules on the lines before and after were not followed.
54	<p>RADIUS PROGRAM. NOT ALLOWED</p> <ul style="list-style-type: none"> . When a connection has been programmed in XYZ.

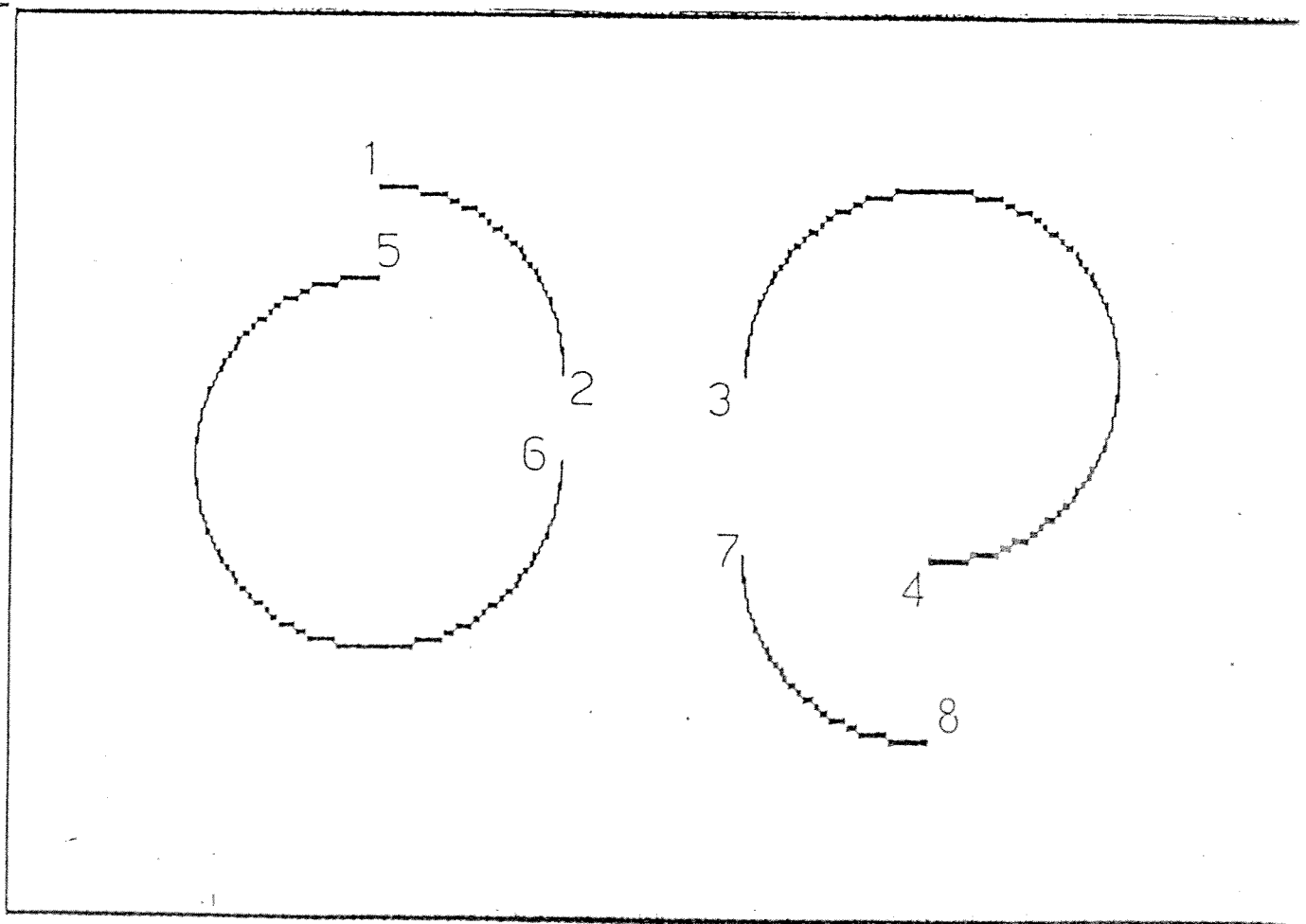
E X A M P L E S

EXAMPLE n.1

(Picture 7.12)

- | | |
|-------------------------|---------------------------|
| 1. G10 X200 Y100 Z10 | |
| 2. G02/ X300 Y200 I100 | ;arc on the radius' value |
| 3. G10 X400 Y200 | |
| 4. G02/ X500 Y300 I-100 | ;arc on the radius' value |
| 5. G10 X200 Y150 | |
| 6. G03/ X300 Y250 I-100 | ;arc on the radius' value |
| 7. G10 X400 Y300 | |
| 8. G03/ X500 Y400 I100 | ;arc on the radius' value |

Picture 7.12

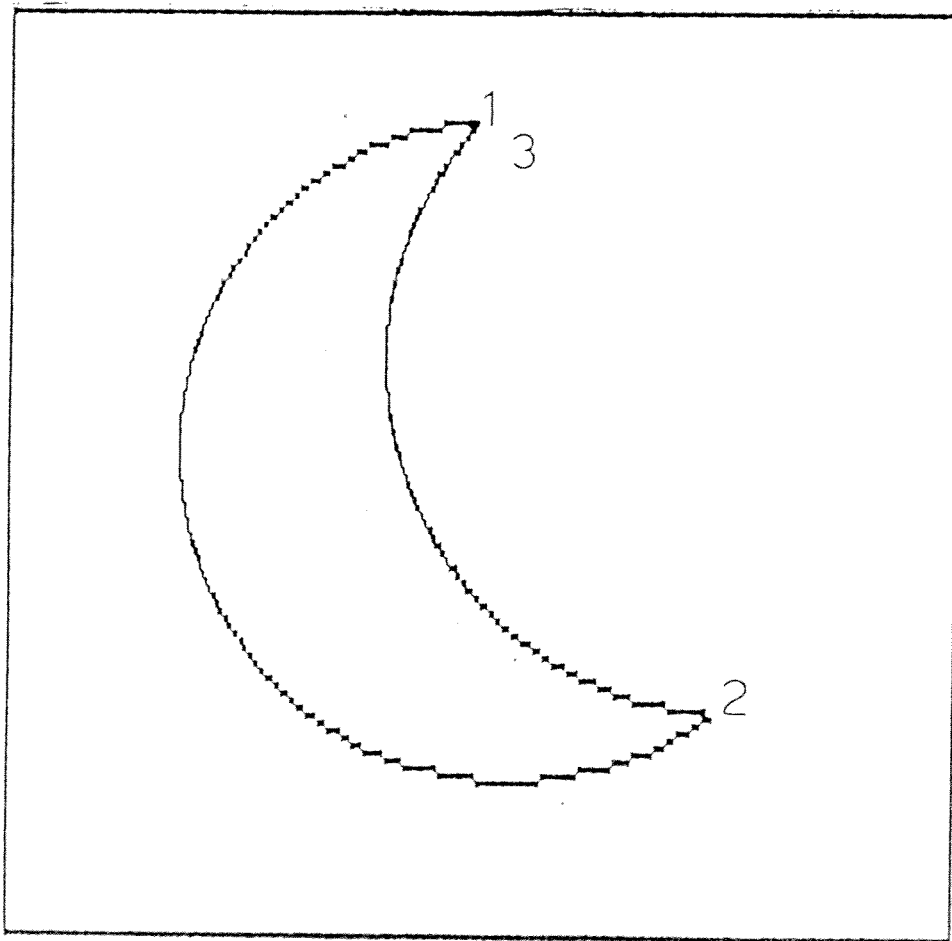


EXAMPLE n.2

(Picture 7.13)

- 1. G10 X100 Y25 Z10
- 2. G13/ X150 Y150 I-70 ;arc on the radius' value
- 3. G02/ X100 Y25 I75 ;arc on the radius' value

Picture 7.13

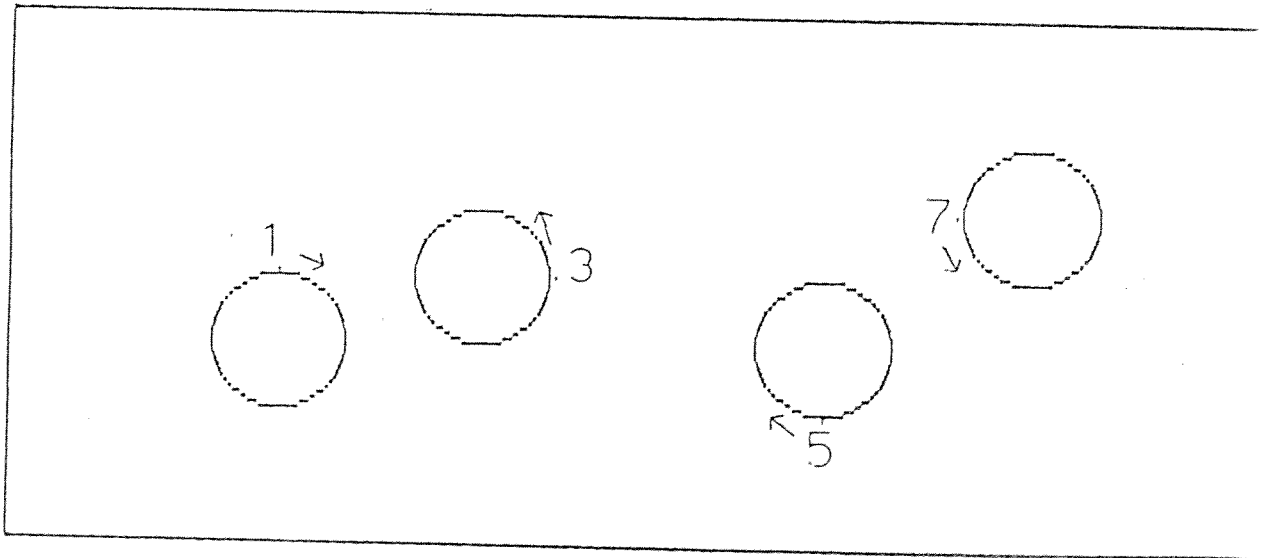


EXAMPLE n.3

(Picture 7.14)

- | | |
|----------------------|------------------------|
| 1. G10 X200 Y200 Z10 | |
| 2. G02/ I50 | ;full circle on radius |
| 3. G10 X400 | |
| 4. G03/2 I50 | ;full circle on radius |
| 5. G10 X600 Y300 | |
| 6. G02/3 I50 | ;full circle on radius |
| 7. G10 X700 Y150 | |
| 8. G03/4 I50 | ;full circle on radius |

Picture 7.14

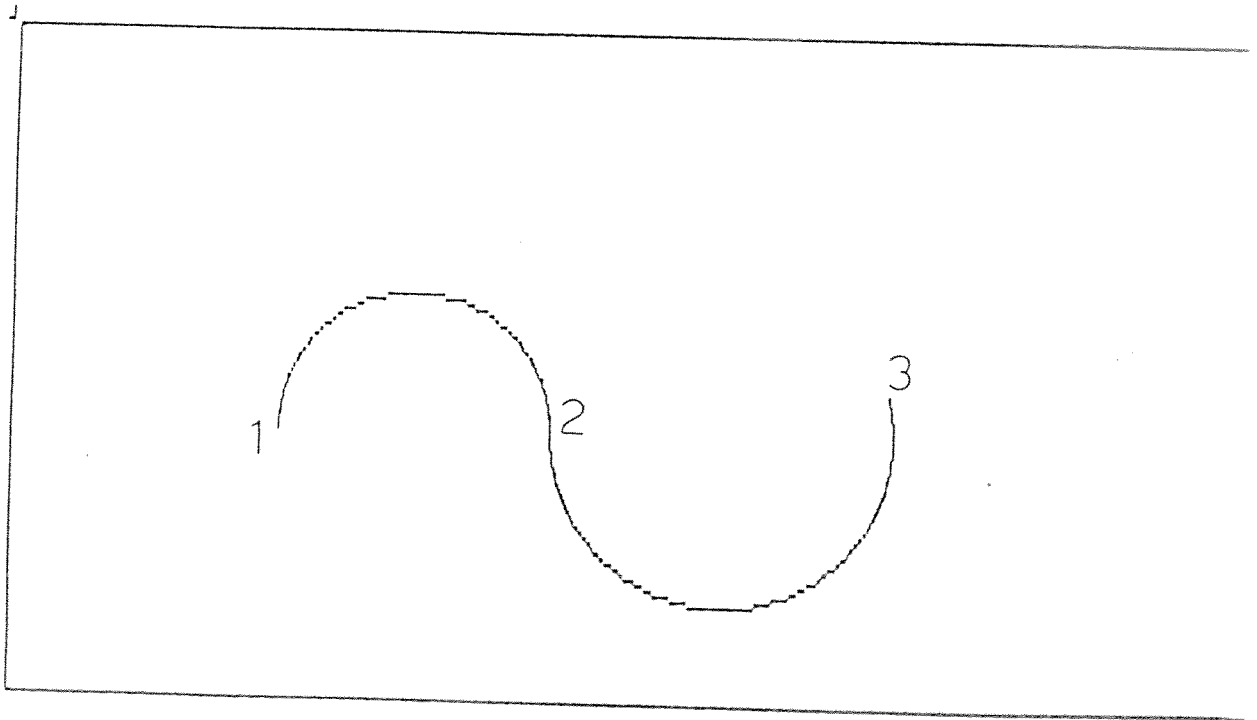


EXAMPLE n.4

(Picture 7.15)

1. G10 X200 Y300 Z10
2. G12 X400 I300 J300
3. G03 X650 Y270 A0 ;arc on initial tangent

Picture 7.15

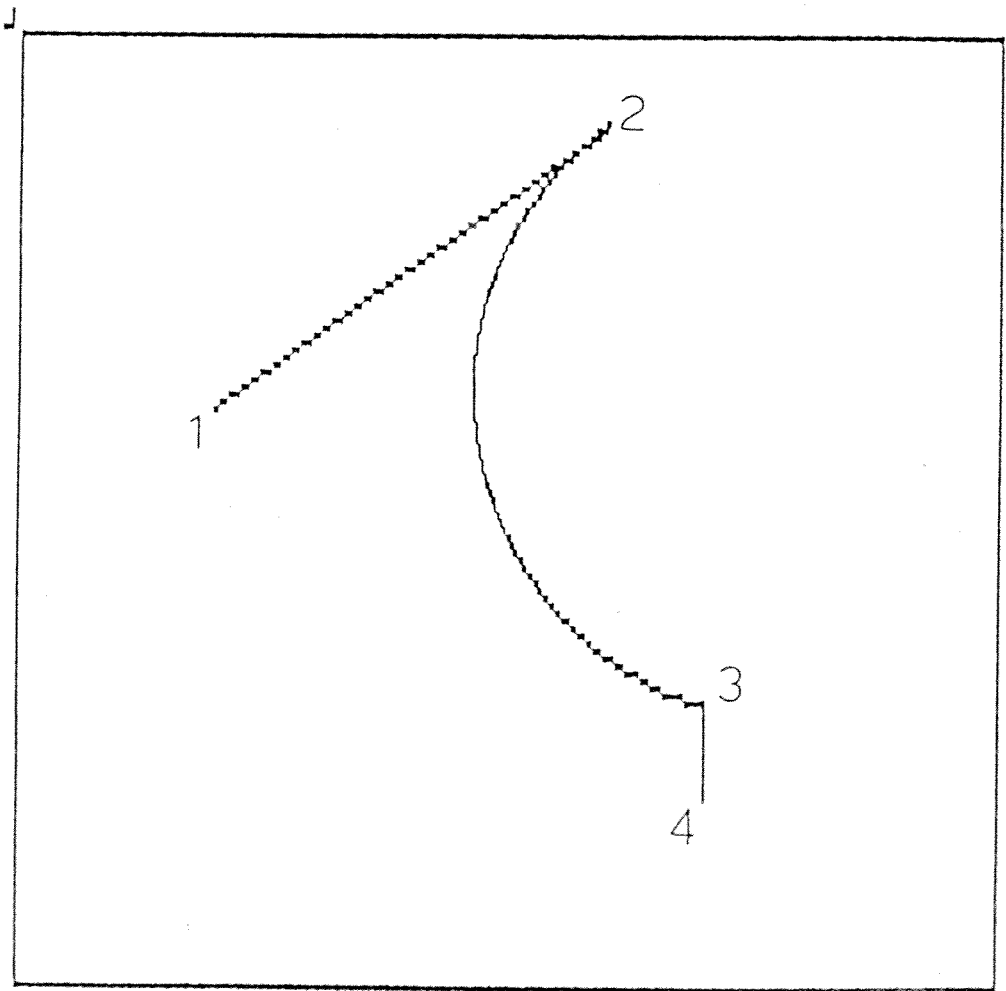


EXAMPLE 5

(Picture 7.16)

- 1. G10 X100 Y200 Z5
- 2. X300 Y50
- 3. G13 X350 Y350 A0 ;arc on initial tangent
- 4. G00 Y400 Z10 PYZ ;final exit on plane YZ

Picture 7.16

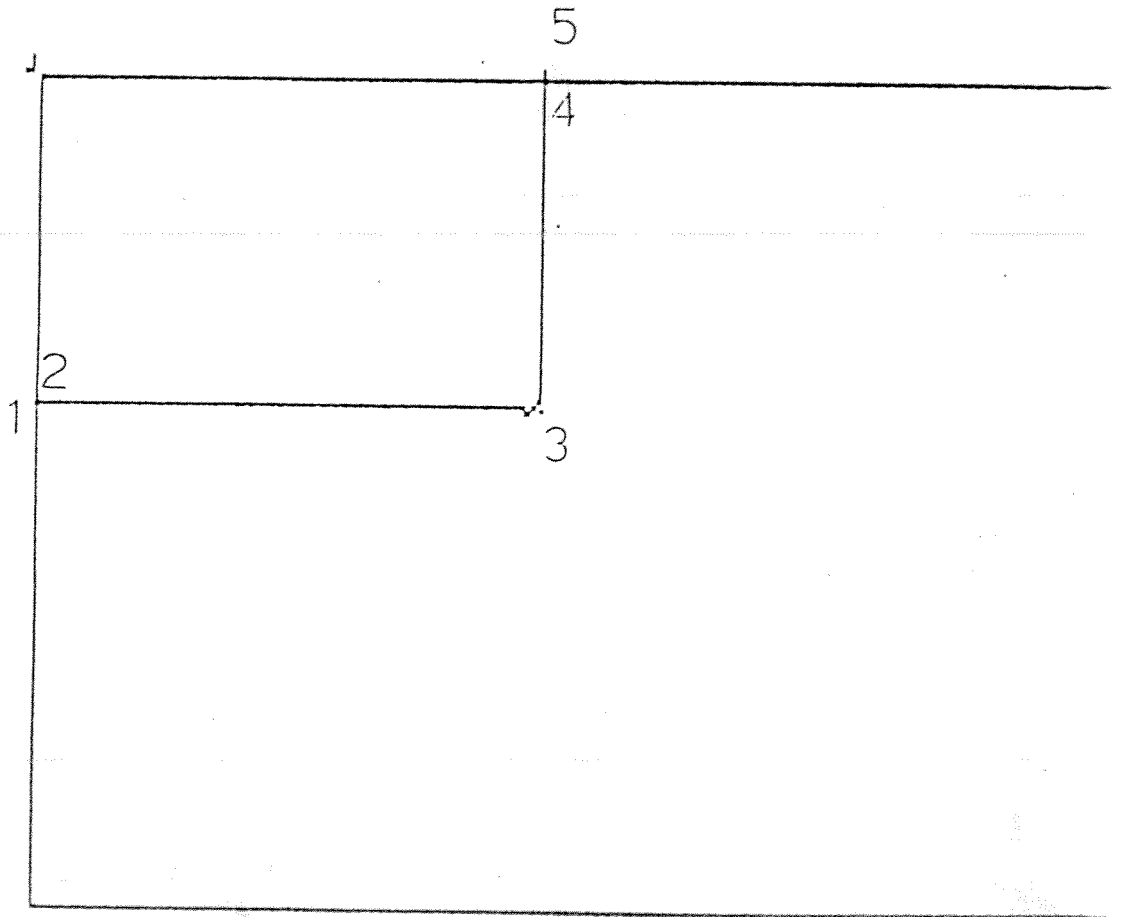


EXAMPLE n.6

(Picture 7.17)

1. G10 X-2 Y200 Z15 PXZ
2. G12 Z5 A10 ;conn.between planes XZ and XY
3. G13 X300 A10 PXY ;connection on plane XY
4. G13 Y-2 A10 PYZ ;conn.between planes XY and YZ
5. G00 Z15

Picture 7.17

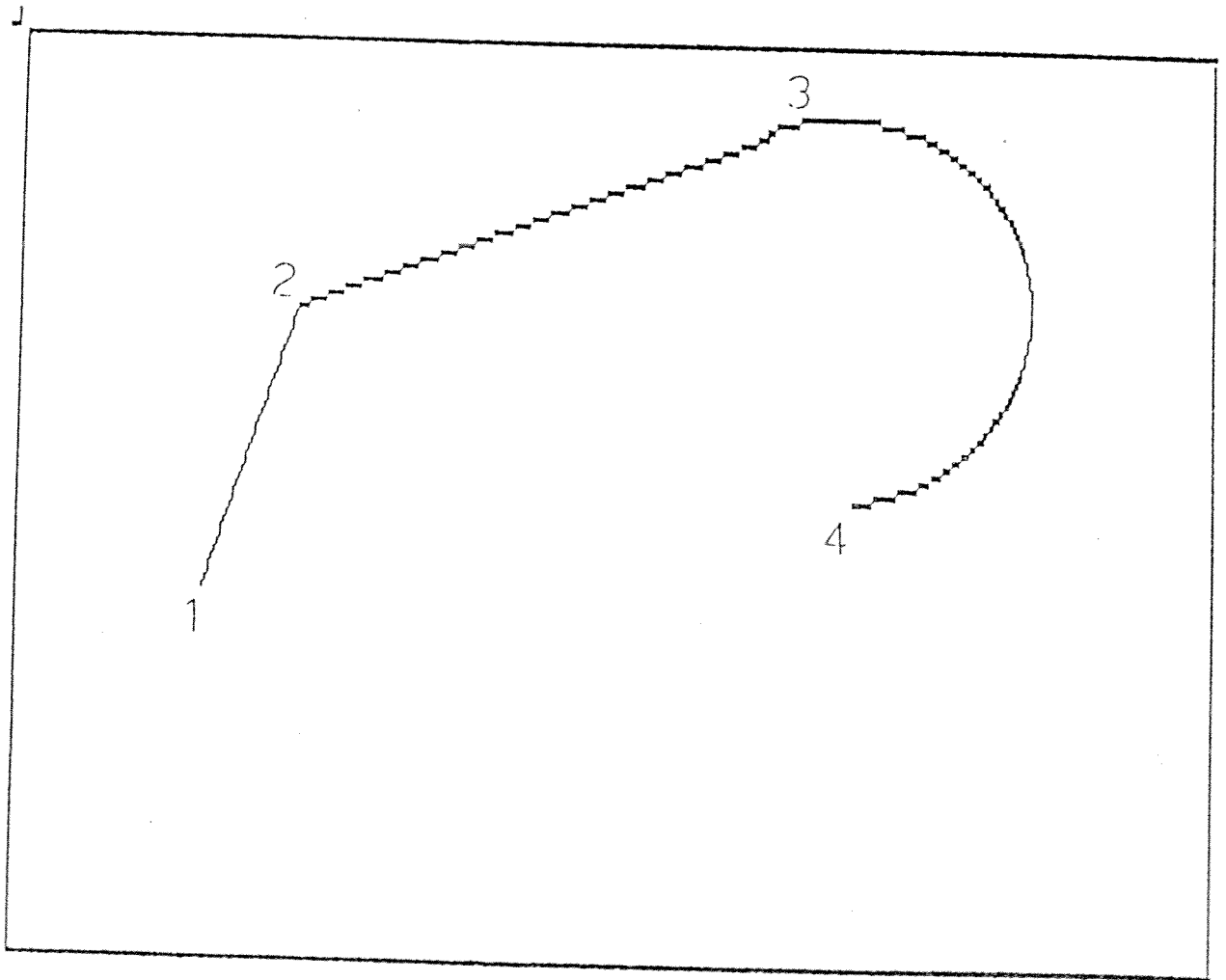


EXAMPLE n.7

(Picture 7.18)

1. G10 X100 Y300 Z10
2. X150 Y150
3. X400 Y50
4. G02 X450 Y250 A0 ;arc through initial tangent

Picture 7.18

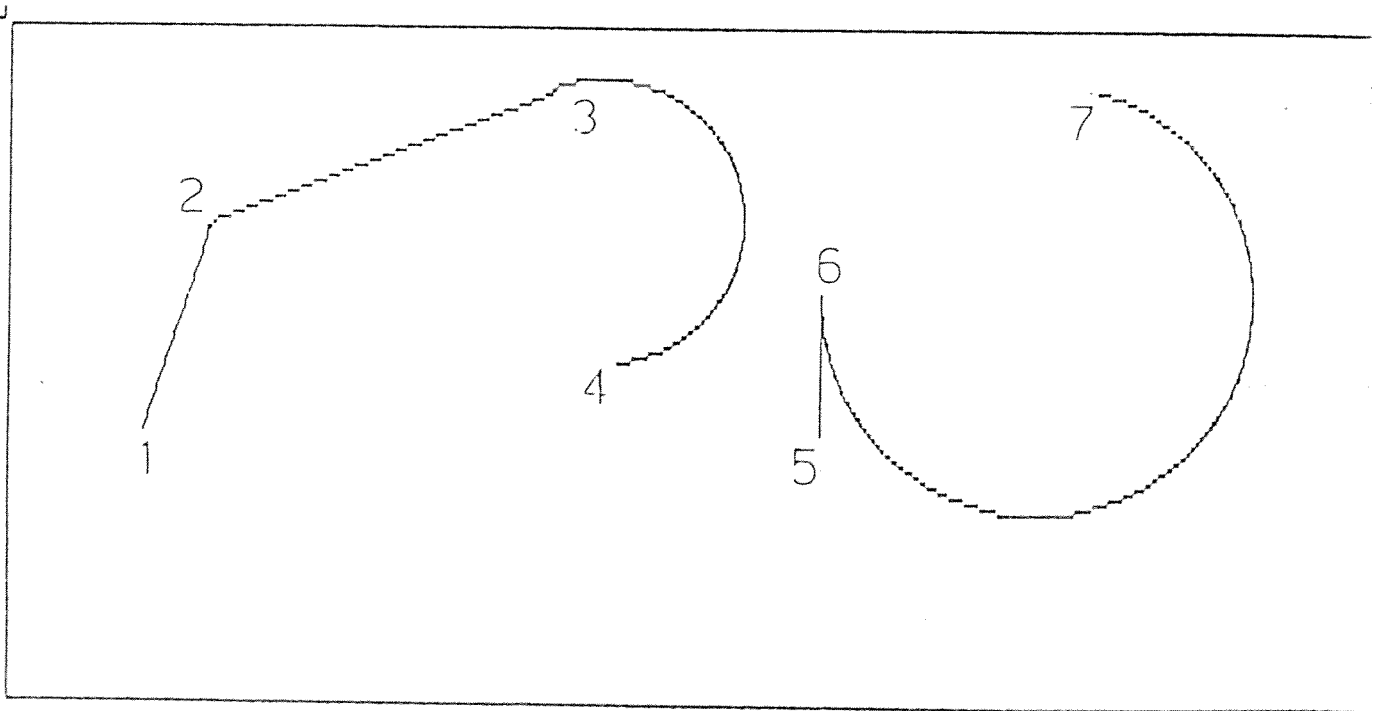


EXAMPLE n.8

(Picture 7.19)

- 1. G10 X100 Y300 Z10
- 2. G12 X150 Y150 A20 ;connection
- 3. G10 X400 Y50
- 4. G02 X450 Y250 A0 ;arc trough initial tangent
- 5. G10 X600 Y300
- 6. Y200
- 7. G03 X800 Y50 A0 ;arc through initial tangent

Picture 7.19

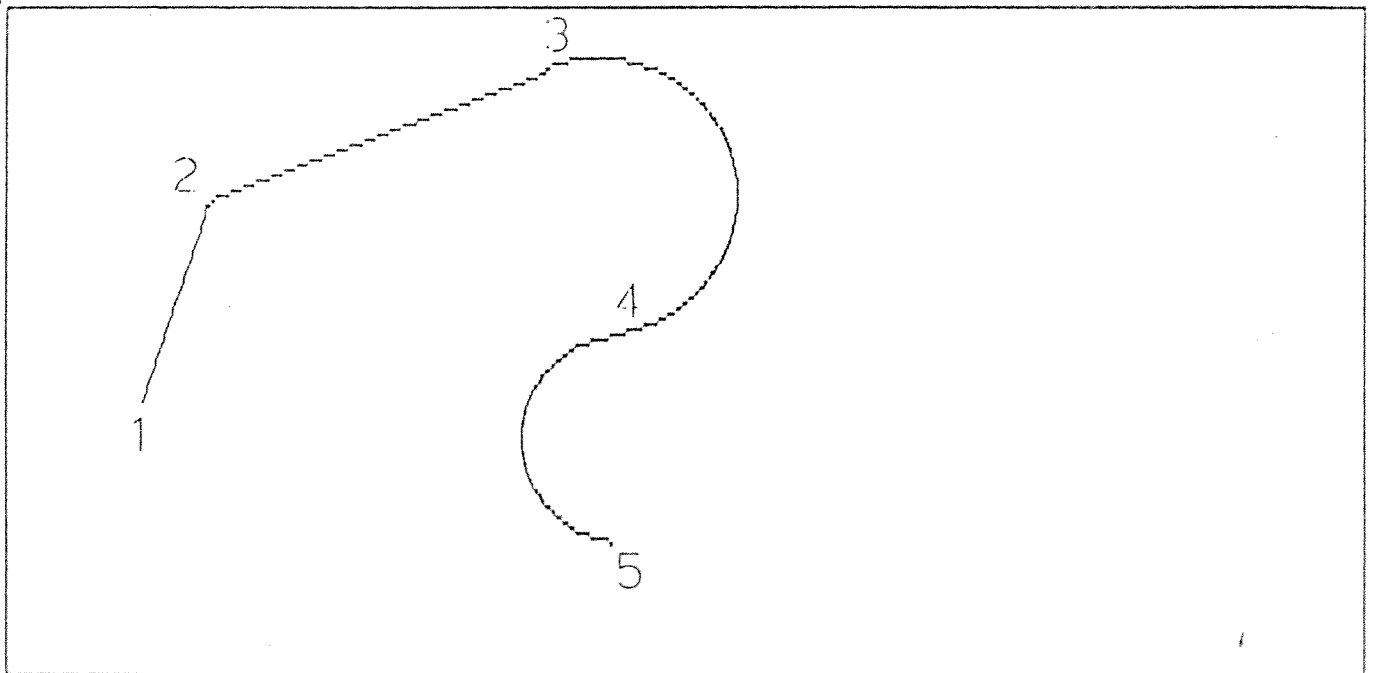


EXAMPLE n.9

(Picture 7.20)

- 1. G10 X100 Y300 Z10
- 2. G12 X150 Y150 A20 ;connection
- 3. G10 X400 Y50
- 4. G12 X450 Y250 A0 ;arc through initial tangent
- 5. G03 Y400 A0 ;arc through initial tangent

Picture 7.20

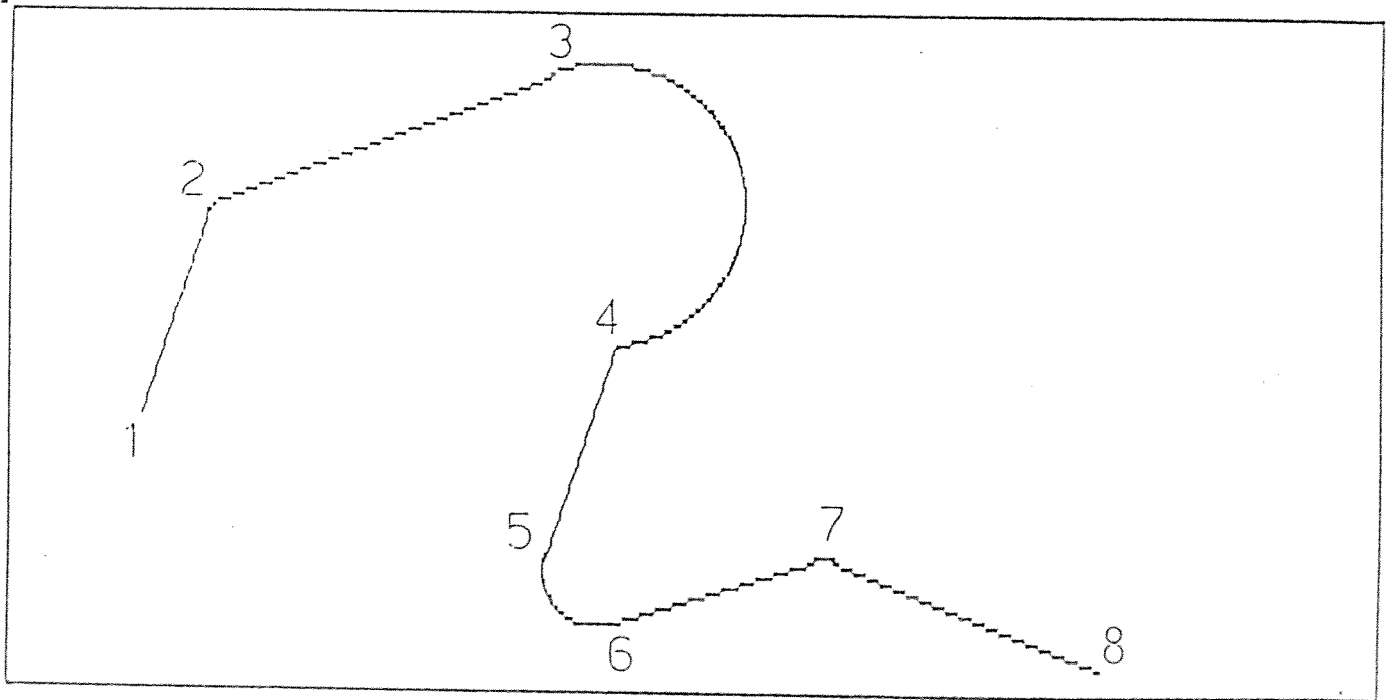


EXAMPLE n.10

(Picture 7.21)

```
1. G10 X100 Y300 Z10
2. G12 X150 Y150 A20 ;connection
3. G10 X400 Y50
4. G12 X450 Y250 A0 ;arc through initial tangent
5. G10 X400 Y400
6. G13 X450 Y450 A0 ;arc through initial tangent
7. G12 X600 Y400 A20 ;connection
8. G00 X800 Y480
```

Picture 7.21



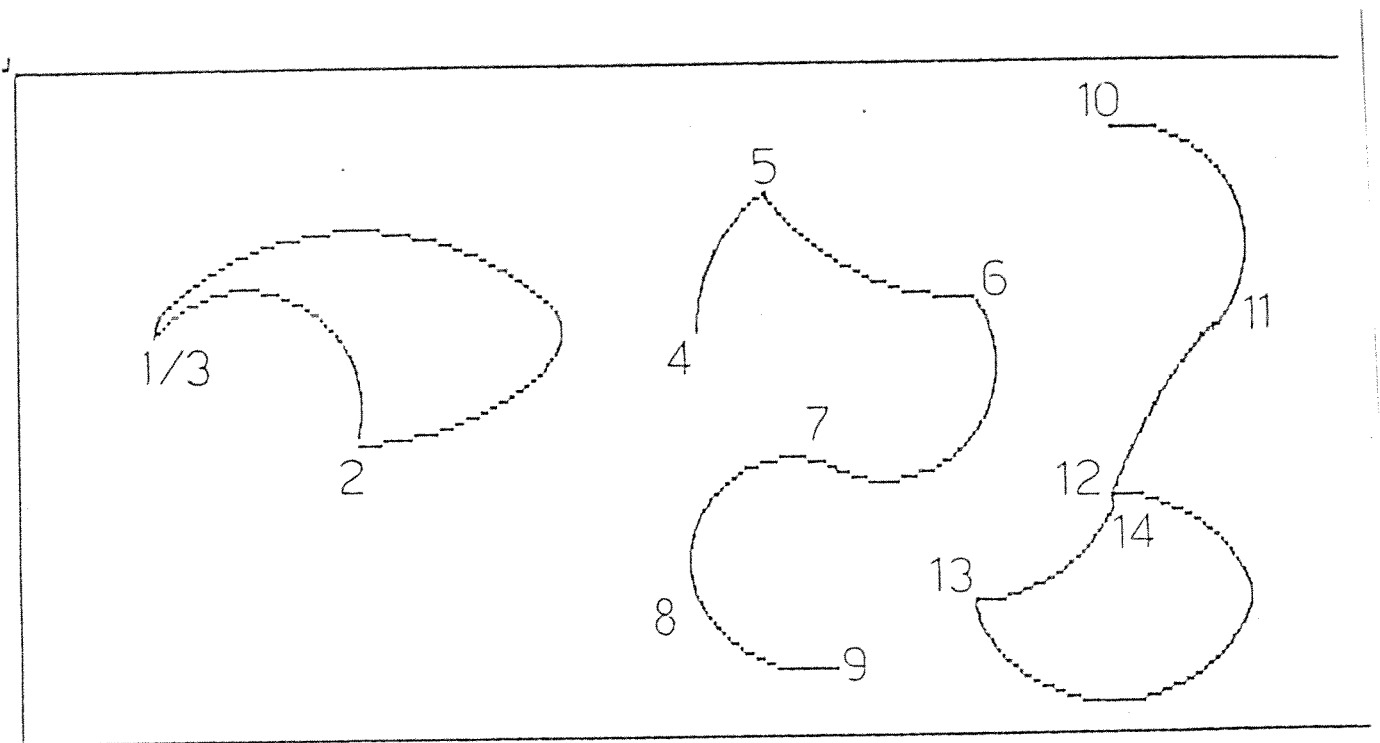
EXAMPLE n.11

(Picture 7.22)

```
1. G10 X100 Y200 Z10
2. G14/3 X250 Y275 I150 J75 A30 ;arc on 3/4 of oval
3. G03/5 X100 Y200 I200 J175 ;arc through three points
4. G10 O2 X500 Y200
5. G14 X450 Y100 I50 J100 A30 ;arc on 1/4 of oval
6. G15 X300 Y175 I150 J75 A30 ;arc on 1/4 of oval
7. G12/5 X400 Y300 I300 J275 ;arc through three points
8. G13 X500 Y400 A0 ;arc through initial tangent
9. G03 X400 Y450 A0 ;arc through initial tangent
10. G10 O3 X200 Y450
11. G12/ X125 Y300 I90 ;arc on radius' programming
12. G13 X200 Y175 A0 ;arc through initial tangent
13. G14/3 X300 Y100 I100 J75 A30 ;arc on 3/4 of oval
14. G15 X200 Y175 I100 J75 A30 ;arc on 1/4 of oval
15. G00 ;tool raise once reached
    the oval's final point
```

In the example are utilized the origins (starting points) on O2 and O3, besides O0 for initial default. See the next chapter for a detailed explanation concerning the use of the origins.

Picture 7.22



SELECTING AND PROGRAMMING ORIGINS

In this chapter will be examined two modes to define the use of different origins :

- How to select an origin
- How to program an origin.

ORIGIN'S SELECTION

Function O in a program line however specified.

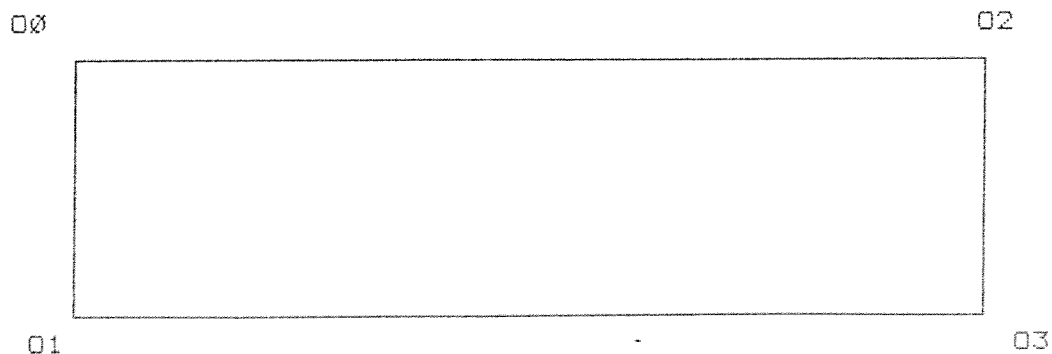
Syntax : Oo[/n] [Ggg] [Xxxxx.xxxx] [Yyyy.yyyy]

where Oo specifies the origin selected ;
o should be specified as a number between 0 and 9 ;
o specifies the reference origin for the axes
co-ordinated XY ;
/n indicates an option (not obliged) on function 0

NOTE : The function "origin" applies only to the axes X and Y.

Function O of Origin's selection spreads to the next program lines till a different specification is given.

There are four Origins predefined (functions from O0 till O3), each related to one of the panel's four corners programmed. They are the specific item's origins, because they are positioned on the four corners of that panel under processing. The picture shows how this relation is assigned :



_____ selecting and programming origins _____

It is opportune to emphasize here that what has been said applies implicitly with control configurated on "System XY of type Ø". Please refer to appendix C for more information.

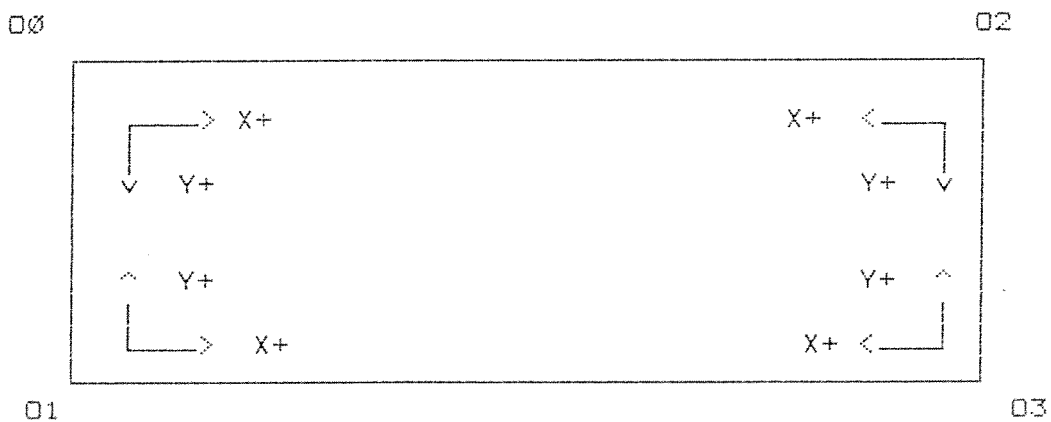
The default origin is OØ : that means that missing any other specification this will be the origin assumed as valid, positioned on the panel's corner next to the machine zero point.

Meaning with LPØ the panel's lenght
HPØ the panel's height

the four Origins result positioned as here specified in respect to the point zero of panel programming :

- OØ { X=Ø ; Y=Ø }
- O1 { X=Ø ; Y=HPØ }
- O2 { X=LPØ ; Y=Ø }
- O3 { X=LPØ ; Y=HPØ }

As clear on the following picture, to each origin refers an orientation of the co-ordinated axes, so to developpe the positive values anyhow within the panel.



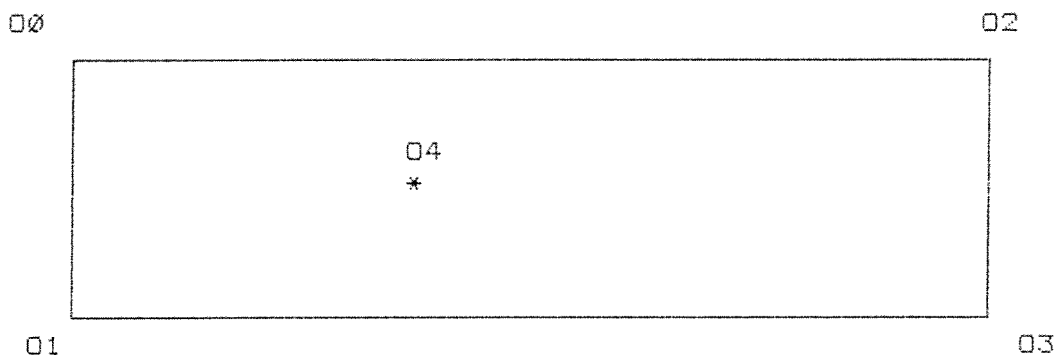
To program negative values remain possible as long as valid for what concerns the verification bonds upon the panel's sizes, and the general limits of the machine.

_____ selecting and programming origins _____

Six more Origins (functions from O4 to O9) are besides available, free to be programmed on the panel.
(see further : ORIGIN'S PROGRAMMING).

In this case on function O can be added the option "/n", in which :
- n can assume a value between 0 and 3 (/0 is by default).

The picture shows an Origin programmed in O4 (signed with "*"), and the four Origins programmed by default :



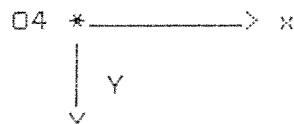
therefore, there are four possible ways to select Origin O4 :
- O4 (equivalent to : O4/0)
- O4/1
- O4/2
- O4/3.

The selection on Origin value 4, assigns the Origin's position (the co-ordinates x and y of the Origin point as determined by the Origins programming function).

By selecting in "option" , from "/0" to "/3", the orientation of the axes X and Y, is assigned on the position in O4.

So :

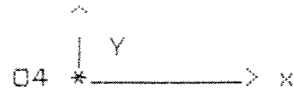
- to O4 (or O4/0) corresponds an axes'orientation as depicted in the picture :



In analogy to Origin O0 .

_____ selecting and programming origins _____

- to 04/1 corresponds an axes' orientation as shown in the following picture :



in analogy to Origin 01.

- To 04/2 correspond an axes' orientation as in the picture :



in analogy to Origin 02.

- to 04/3 corresponds an axes' orientation as depicted in the picture :



in analogy to Origin 03.

ORIGIN'S PROGRAMMING

Syntax : O(o) [Ggg] [Xxxxx.xxxx] [Yyyy.yyyy] [On]

where o should be specified as a number from 4 to 9 ;
 o specifies the origin under programming for the axes co-ordinated XY .

The function "Origin's programming", automatically associates to the function G11 : therefore, the assignment on function Ggg is not obliged. However, if existing, it should be assigned G11.

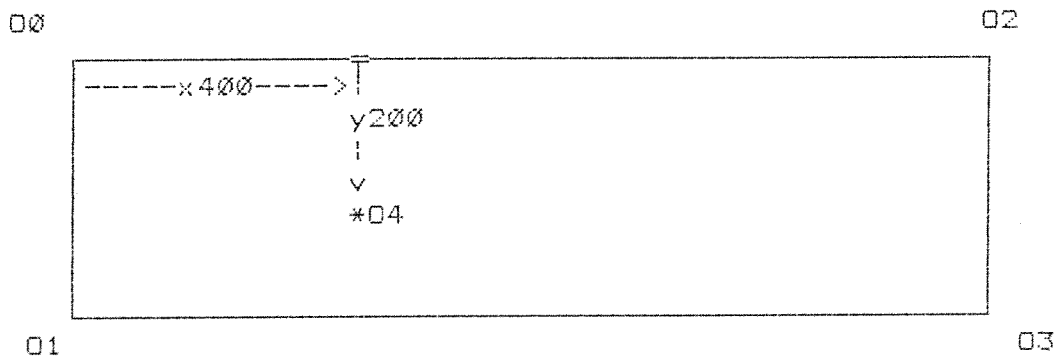
The function On is employed to select the specific Origin onto which is based the programming of O(o). If not existing, it is assumed the origin spreaded from the preceding line.

The Origin to enter in Oo can be anyone among the valid ones :

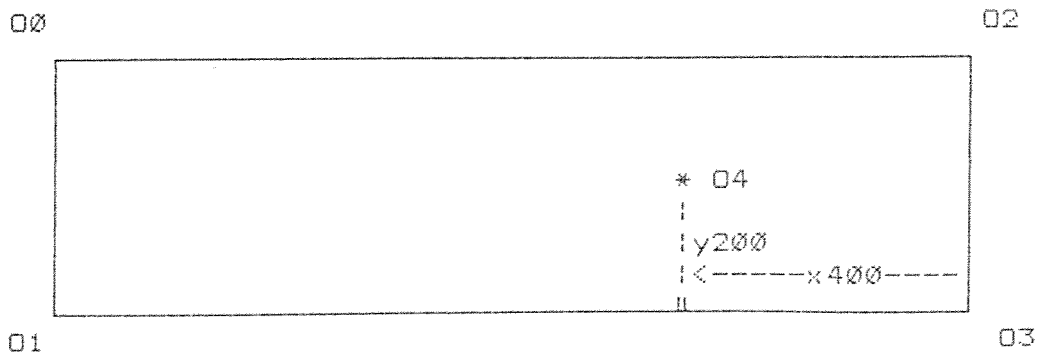
- from O0 to O3, always valid.
- from O4 to O9, among the ones already programmed in the preceding lines.

NOTE : the round brackets are obliged.

Example : O(4) X400Y200 O0



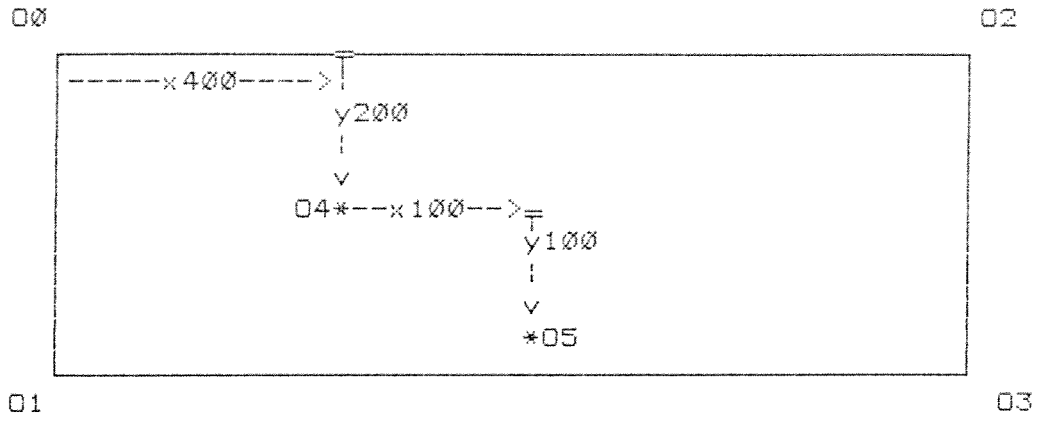
Example : O(4) X400Y200 O3



selecting and programming origins

Example: O(4) X400Y200 O0

O(5) X100Y100 O4

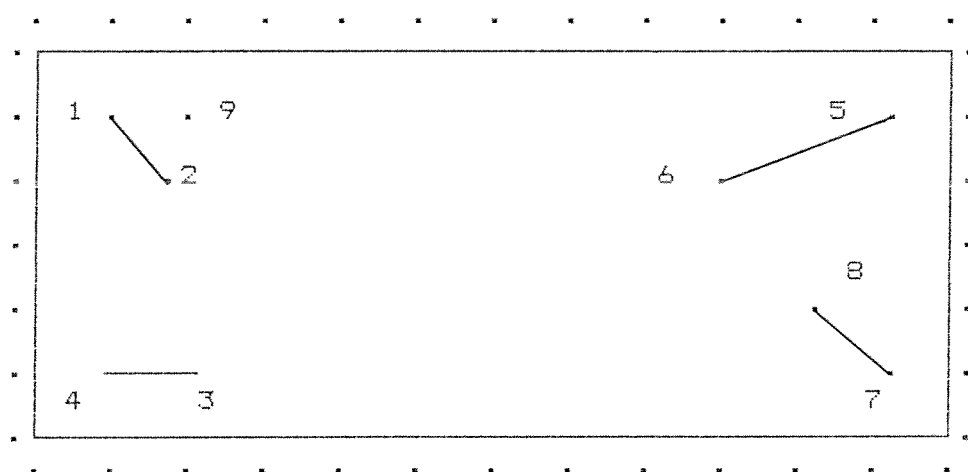


selecting and programming origins

The following example shows the way of using the function ORIGIN'S SELECTION, limited to origins selected per default :

```
G10 X100 Y100 ; point 1 , origin 0 (default)
G00 X200 Y200 ; point 2 , origin 0 (spreaded)
G10 X200 Y100 O1 ; point 3 , origin 1
G00 X100 ; point 4 , origin 1 (spreaded)
G10 X100 Y100 O2 ; point 5 , origin 2
G00 X300 Y200 ; point 6 , origin 2 (spreaded)
G10 X100 Y100 O3 ; point 7 , origin 3
G00 X200 Y200 ; point 8 , origin 3 (spreaded)
G00 X200 Y100 O0 T2 ; point 9 , origin 0
```

Panel's lenght = 1200
Panel's height = 600

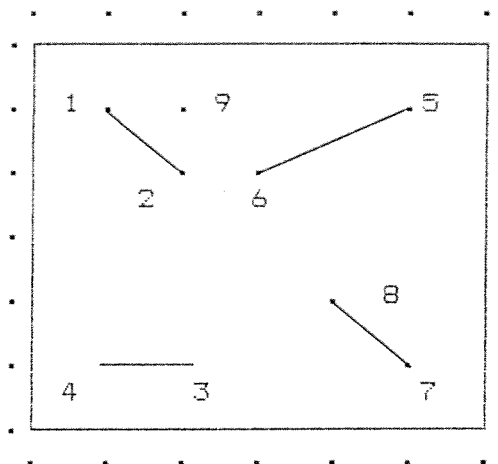


The main advantage of command O and its immediate use is in recovering a program when processing panels with different sizes : in fact, the processings are specified referred to the panel's Origins : if they change, the program will be adapted to the new origins.

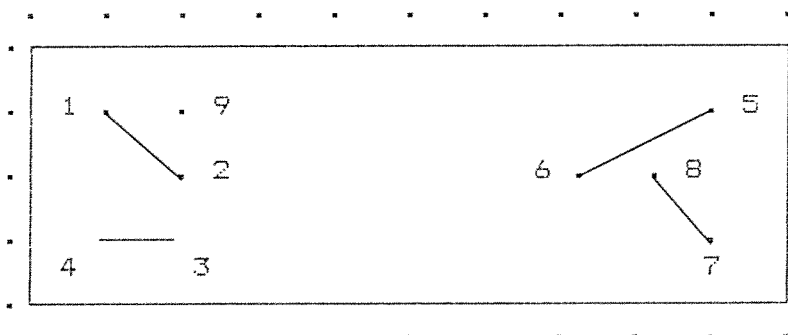
selecting and programming origins

The two following pictures show the same program with different panel's dimensions.

Panel's lenght = 600
Panel's height = 600



Panel's lenght = 1000
Panel's height = 400



_____ selecting and programming origins _____

Here, it is proposed an example of how to use the function Origin in operations of fitting :

```
G20X100Y100Z10T1T2T3T4T5  
G0002
```

The execution of the fitting can vary according with the panel's lenght, adapting on total number of holes and -therefore- of ref.points.

The value of start fitting is determined on Origin 0, and it is therefore in $X=100$. The value of end fitting is determined on Origin 2 : $X=LP0 - 100$, where $LP0$ is the panel's lenght.

The same reasoning can be applied to operations of grooving, in x or in y :

```
G16X-5Y100Z10           ;grooving in x  
G0002
```

```
G17X100Y-5Z10          ;grooving in y  
G0001
```

The examples following show in general the way of using the functions to select and program an Origin.

EXAMPLE n.1

(Picture 8.1)

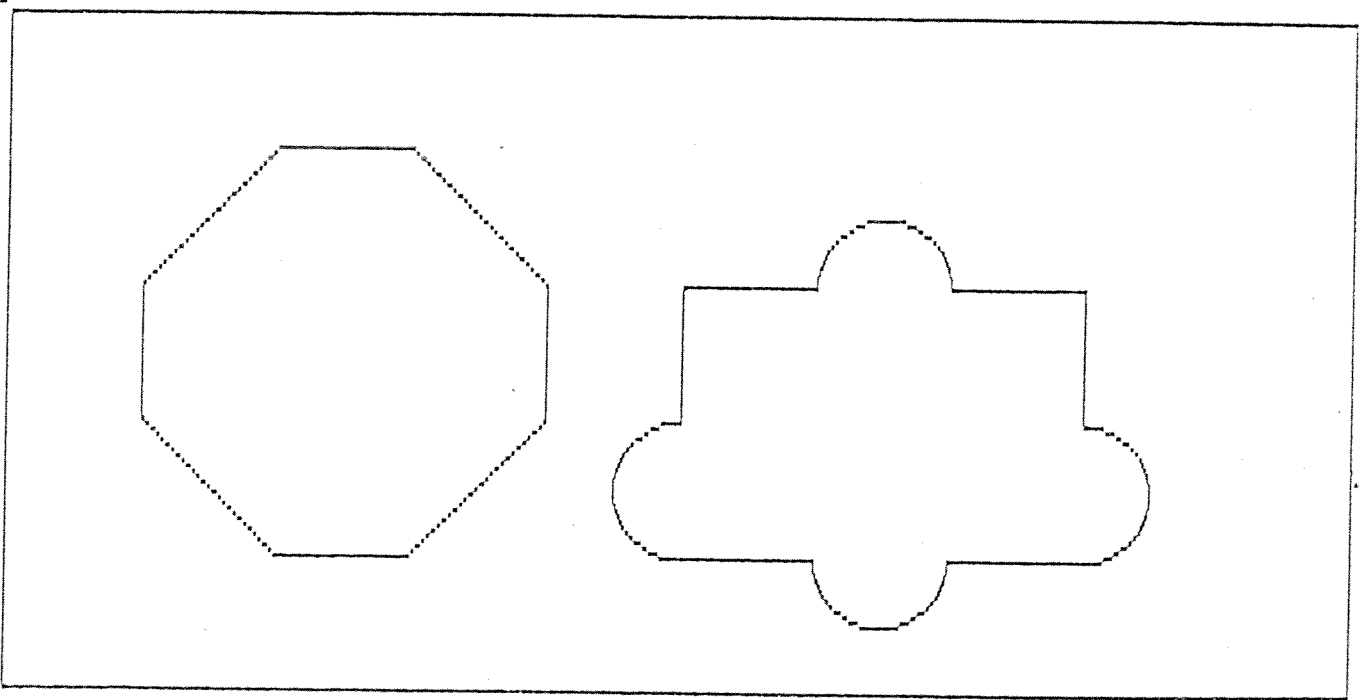
```

1. G10 X100 Y200 Z5
2. O(4) ;progr. 04 (reference origin : 00)
3. X100 Y100 O4/1
4. X200
5. X300 Y0
6. O(5) ;progr. 05 (reference origin :04)
7. Y100 X0 O5/2
8. X100 Y200
9. X200
10. X300 Y100
11. G00 Y0 X0 O4
12. O(6) X100 Y0 O5 ;progr. 06 (reference origin :05)
13. G10 X0 Y0 O6
14. Y100
15. G13 Y200 I0 J150
16. O(8) ;progr. 08 (reference origin :06)
17. G10 X100 Y0 O8/1
18. G13 X200 I150 J0
19. G10 X300
20. O(7) ;progr. 07 (reference origin :08)
21. G13 X0 Y100 I0 J50 O7/3
22. G10 Y200
23. X100
24. G13 X200 I150 J200
25. G00 X0 Y0 O6
    
```

In the example :

- origin 04 -> reference origin 00
 -> co-ordinates X100 Y200
 it is therefore positioned in $x=100$ $y=200$;
- origin 05 -> reference origin 04
 -> co-ordinates X300 Y0
 it is therefore positioned in $x=100+300=400$ $y=200+0=200$;
- origin 06 -> reference origin 05
 -> co-ordinates X100 Y0
 it is therefore positioned in $x=400+100=500$ $y=200+0=200$;
- origin 08 -> reference origin 06
 -> co-ordinates X0 Y200
 it is therefore positioned in $x=500+0=500$ $y=200+200=400$;
- origin 07 -> reference origin 08
 -> co-ordinates X300 Y0
 it is therefore positioned in $x=500+300=800$ $y=400+0=400$.

Picture 8.1

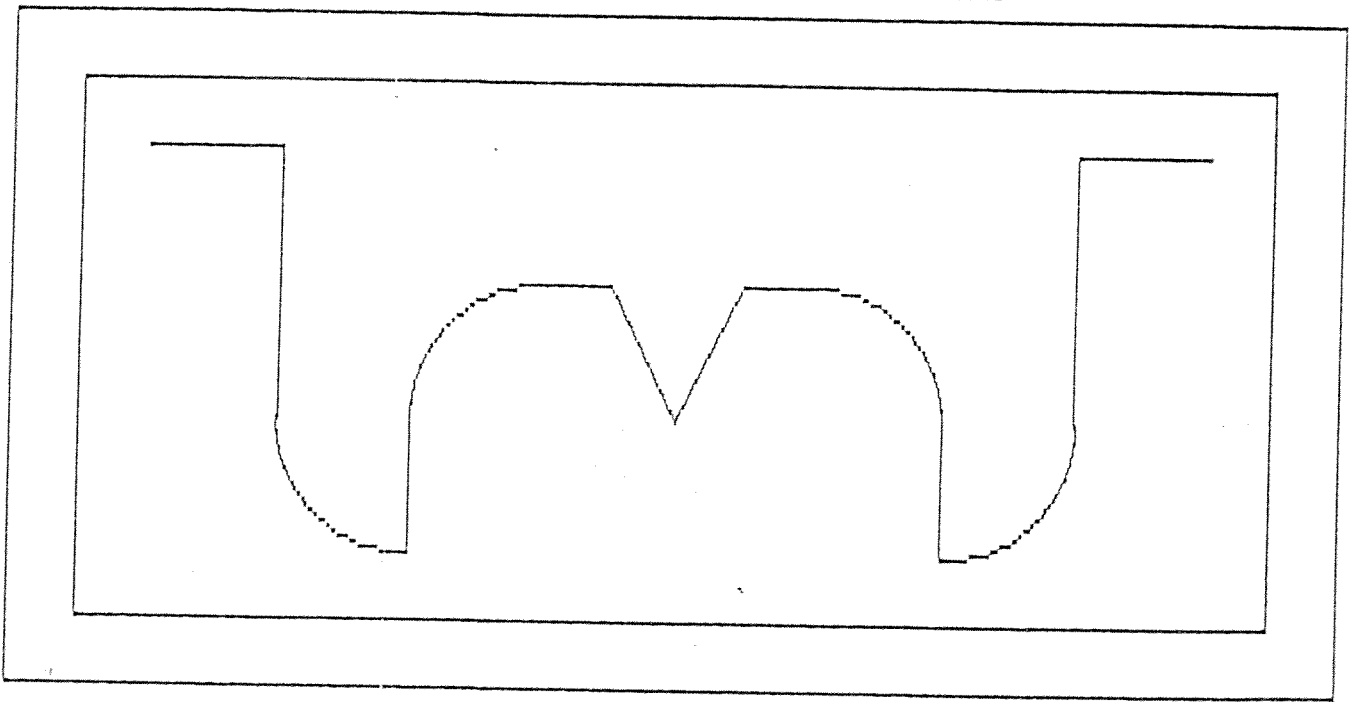


EXAMPLE n.2

(Picture 8.2)

1. O(4) X50 Y50 O0
2. O(5) O2
3. O(6) O1
4. O(7) O3
5. G10 X0 Y0 O4
6. O5/2
7. O7/3
8. O6/1
9. O4 G00
10. G10 X100 Y100 Z5 O0
11. X200
12. Y300
13. G13 X300 Y400 I300 J300
14. O(8)
15. G10 X0 Y100 O8/1
16. G12 X100 Y200 I100 J100
17. G10 X150
18. X200 Y100
19. X250 Y200
20. X300
21. G12 X400 Y100 I300 J100
22. G10Y0
23. G13 X200 Y300 I300 J300 O2
24. G10 X200 Y100
25. G00 X100

Picture 8.2



REMARKS CONCERNING THE EXECUTION IN OPTION "D"

The program's layout in operations with the function Origin is of basilar importance when using the option "D" to run programs. Here are listed the origins' programming rules in order to exploit at its best the option "D" on programs :

- It is opportune to divide all processings in two groups : one relative to Origins 0 and 1, and the other relative to Origins 2 and 3. This way, when running the program with option "D", it will be performed only one change of reference, with consequent time optimization in panel processing.
- Option "D" allows to run a program transferring the tolerancies of the panel's real lenght, in the middle of the panel itself : the values programmed are infact converted into the axes' real position (X and Y) in relation with the two ref.point in question, and are not reckoned on the panel's theoretical position entered.
However, this is true only if the program is set up, on the two references, in such a way to define two different work areas not interfering with each other, both along axis X.
- According with what said in the two preceding points, it is reccomended to refer consecutive millings to a single Origin : this is infact the only way to obtain processing values independent from the panel's lenght stated.
When the machine runs on two references, as valid reference is assumed the Origin specified on the G10 of tool lowering.

```
Example :  G10 X200 Y200 00
           X500
           X300 Y100 02
           X100 Y300 03
           G00 X400 Y400 01
```

the program text above reported does not determine faults, nor when edited nor when the program runs, with or without option "D" specified : infact, in case of option "D" specified, it is anyway assumed Origin 00 (as specified in the first program line) to locate the execution's ref.point. The processings referred to Origins 02 and 03, in the 3rd and 4th line, are performed according with the panel's lenght stated (theoretical), and not with the real lenght : this cuts off the execution of the program in option "D".

selecting and programming origins

- In a program using programmed Origins (Origins from 04 to 09), are anyway assumed as valid the reference Origins located on the corners (from 00 to 03) when running the option "D".
So : on Origin 04 programmed with reference Origin 00 the processings programmed on 04 are related to Origin 00 ; if a further Origin 05 is programmed, having as reference Origin 04, also the processings programmed on 05 are related to Origin 00.

EXAMPLE

Panel's lenght = 1000
Panel's height = 400

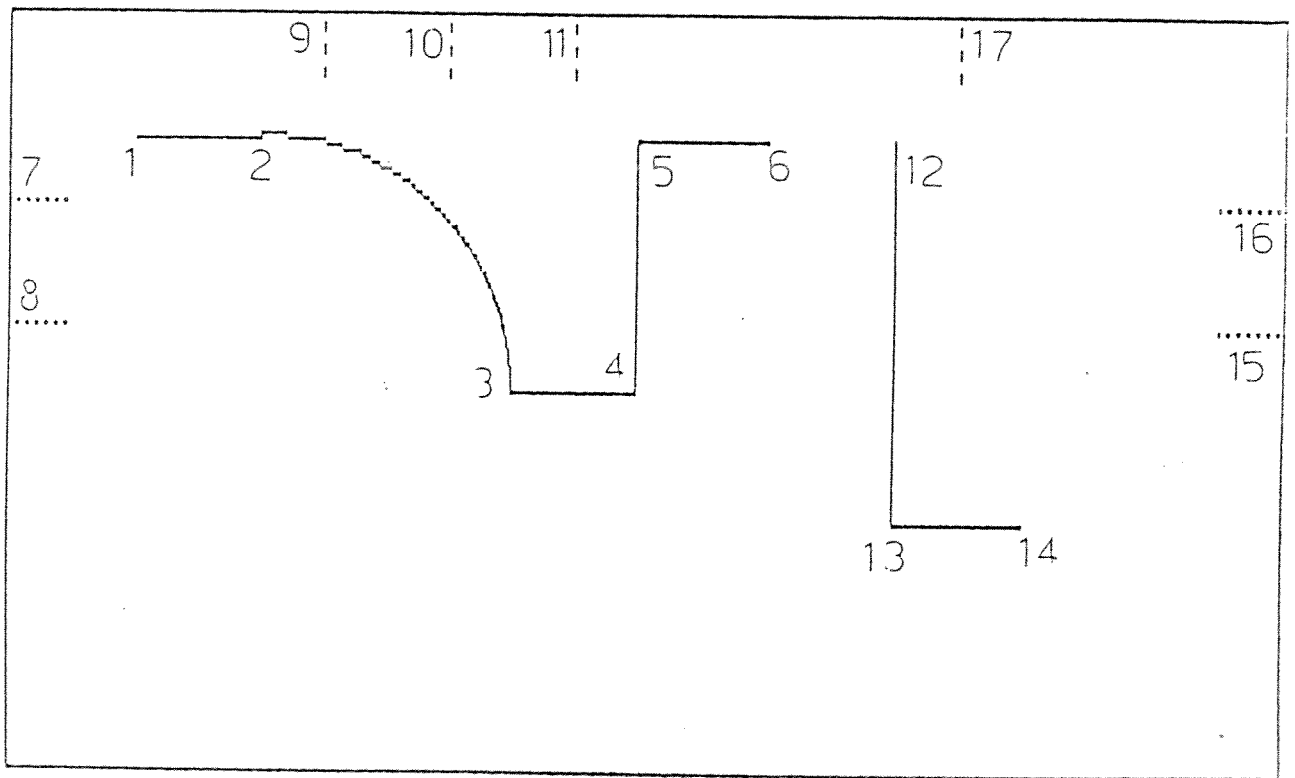
1. G10 X100 Y100 Z10 ; origin 0 per default
2. X200
3. G12 X400 Y300 I200 J300
4. G10 X500
5. Y100
6. G00 X600
7. G06 X3 Y150 Z10
8. G06 Y250
9. G08 X250 Y3
10. G08 X350
11. G08 X450
12. G10 X300 Y100 Z10 O2 ; origin 2
13. Y400
14. G00 X200
15. G07 X3 Y250 Z10
16. G07 Y150
17. G08 X250

The program has been typed so to divide the processings on the panel in these two areas :

- area related to Origins 0 and 1 : from X<0 to X=600.
Every operation related to Origins 0 and 1 is performed within this area.
- area related to Origins 2 and 3 : beyond the absolute value X=600.
Every operation related with Origin 2 and 3 is performed within this area.

These two areas are defined as separate : this allow to find, between these two areas, the part of the panel in which to discharge what has been left over in the relation between the panel's real and theoretical lenght :
this area lays between X=600 and X=700.

Picture



The table reported below defines the execution terms with inquiry of option "D":

per each work field are specified the kind of processing performed (normal or mirrored), the field related (that means the alternative ref.point), the Origins related to the ref.point on the specified field and the ones related to the alternative ref.point.

specified field	kind of processing	related field	Origins on specif. field	Origins on assoc.field
S	normal	A	Ø , 1	2 , 3
T	normal	R	Ø , 1	2 , 3
N	normal	M	Ø , 1	2 , 3
A	mirror	S	Ø , 1	2 , 3
R	mirror	T	Ø , 1	2 , 3
M	mirror	N	Ø , 1	2 , 3

When the program runs, the change of ref.point is signaled in the following way :

1. It is performed the function of end panel on the running field (for instance : resident function FUN ØØ1, if the execution is of type N)
2. It is performed the function of start panel on the new field (in the example : FUN ØØ3, being the field M related with N) ;
3. the operator has to move the panel to the new reference, with the machine waiting on field start (as defined in function "start panel in execution")
4. With "Field Start" the panel processing on the new reference starts again.

_____ mill radius correction _____

MILL RADIUS CORRECTION

Function R in a command G10 of start milling cycle :

G10 Xxxxx.xxxx Yyyyy.yyyy Zzzzz.zzzz [VXnn] [Ttt] [R+-nn]

where R+- nn should be specified in the following way :

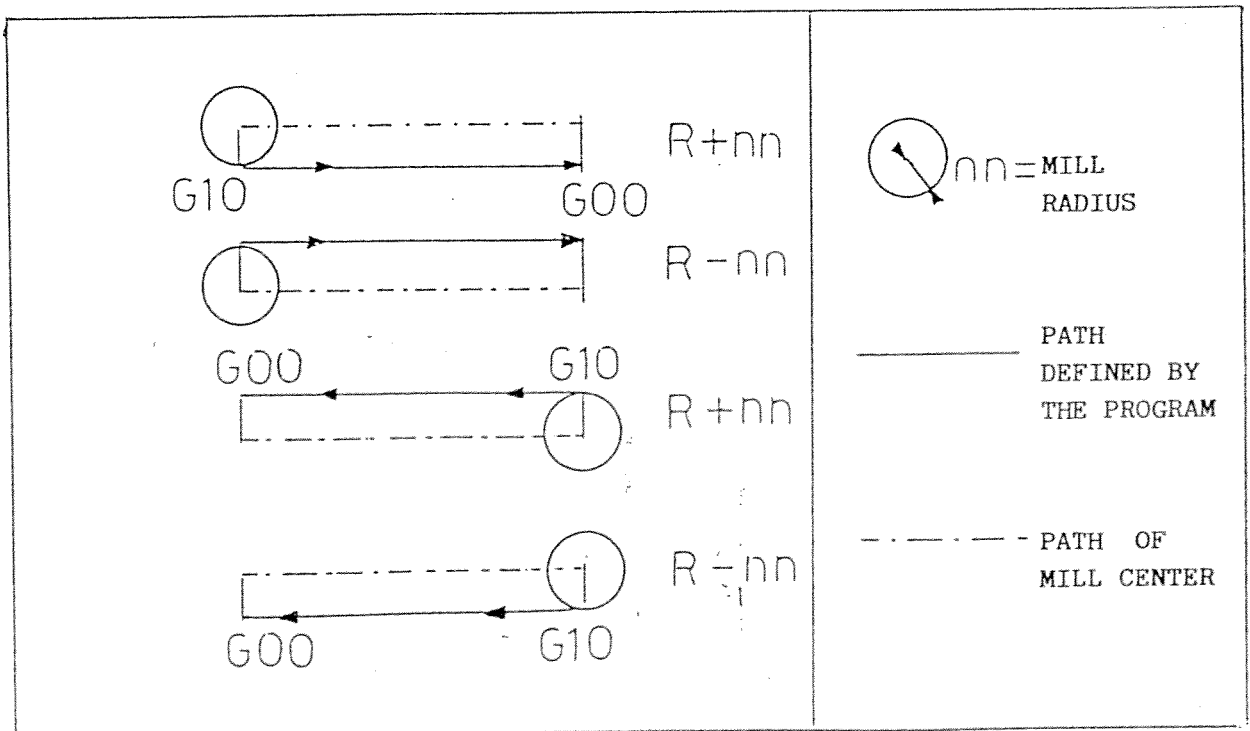
nn specifies the mill radius
nn should never be zero.

if introducing R+ or R- (that is : nn omitted) the mill value is read from the data entered in Parameters. In other words, as correction value is assumed the mill radius entered in the machine parameters concerning the tool (mill) at work.

The sign associated to the mill radius specifies if the correction has to be carried out on the left or the right side of the picture in subject :

- sign "+" : corrects at the left side of the trajectory.
 - sign "-" : corrects at the right side of the trajectory.
- (right side and left side are determined by following the oriented direction of the milling (see figure in picture 9.1))

Picture 9.1



The correction on the milling radius is accepted only in figures interpolated on **plane XY** : other way, it is displayed a fault for "Tool radius correction not allowed".

The tool radius correction acts in normal direction in respect to the contour programmed, composed of segments of straight line and circumference arcs.

The simpler request of mill radius correction concerns a contour in interpolation defined on a single geometric tract (circular or linear).

The pictures to follow depict these elementar cases in determining the correct contour :

- first, it is proposed a linear tract, with correction on positive sign (first drawing) and negative sign (second drawing) The geometric tract resulting when the correction has ended runs parallel to the original tract, at a distance that equals the mill radius.

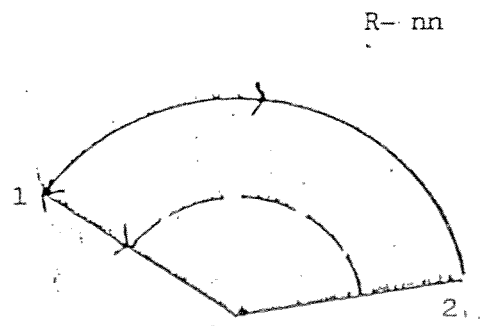
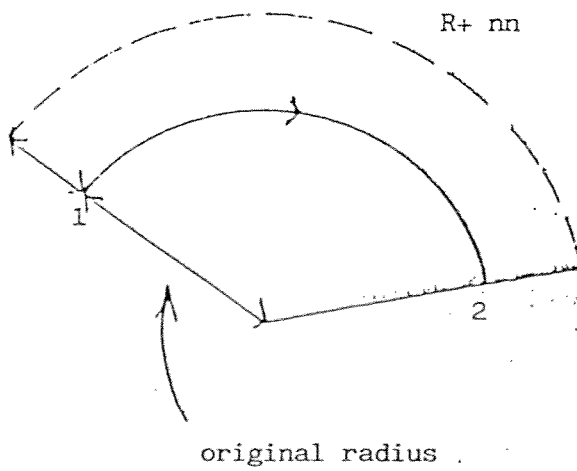
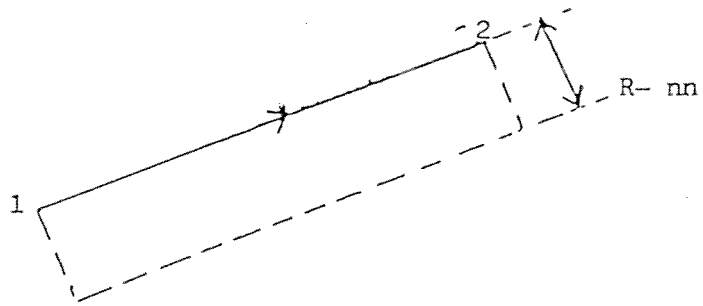
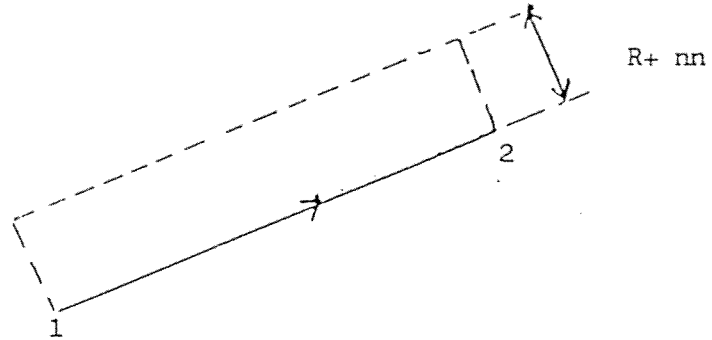
- It is then proposed a circular tract with correction on positive sign (first drawing) and on negative sign (second drawing) of the mill radius. The geometric tract resulting once the correction has ended is concentrical in respect to the original arc, and it is defined on a radius that differs from the original radius for the value of the mill radius.

In the example reported, of tract defined on clockwise rotation, to the correction on positive sign corresponds an increase of the radius value, on the element corrected, in respect to the original radius ; the radius reduces once corrected with negative sign.

The criteria to determine the arc correction result reversed with an arc defined on c.clockwise rotation.

In the cases here considered of contours defined on a single tract, also the contour corrected remains defined on a single tract.

Pictures 9.2 A e B



A geometrical contour however defined on more than a single tract should be regarded as a sum of elementary situations of correction, defined on two consecutive geometrical elements. With reference to this, four possible elementary cases are to be distinguished :

- a) case 1 : straight line - straight line ;
- b) case 2 : straight line - arc ;
- c) case 3 : arc - straight line ;
- d) case 4 : arc - arc .

The picture at the next page shows these four cases in a few examples of correction.

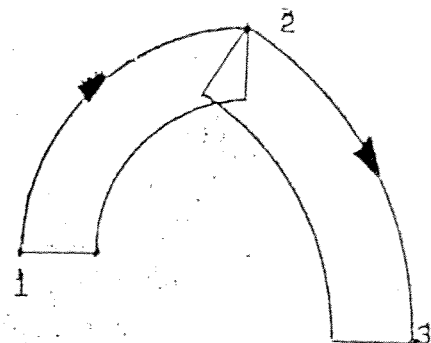
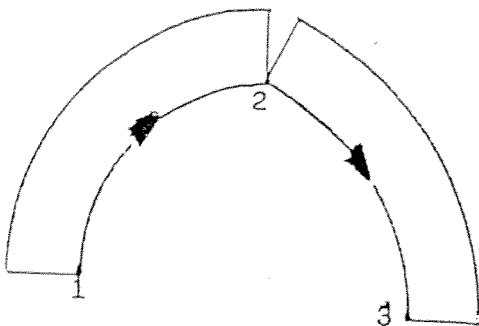
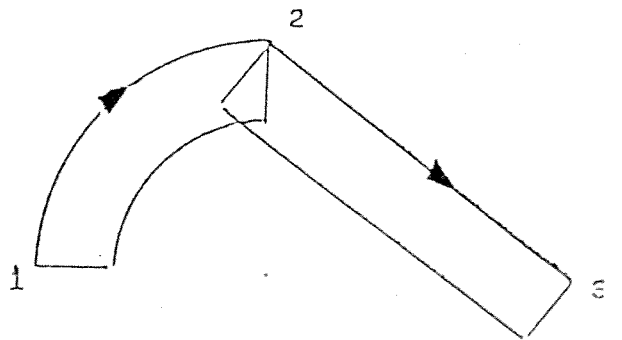
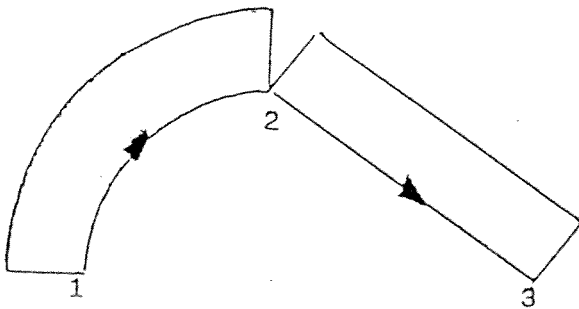
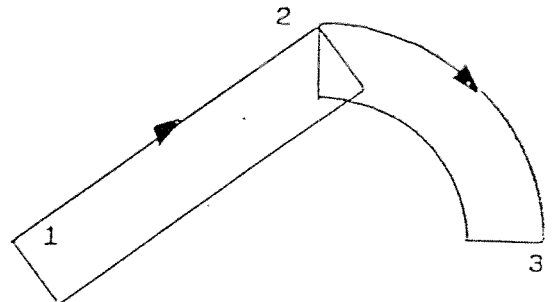
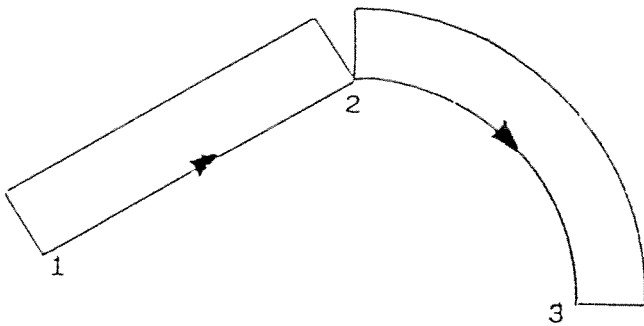
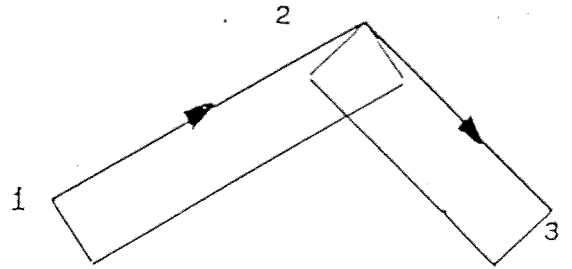
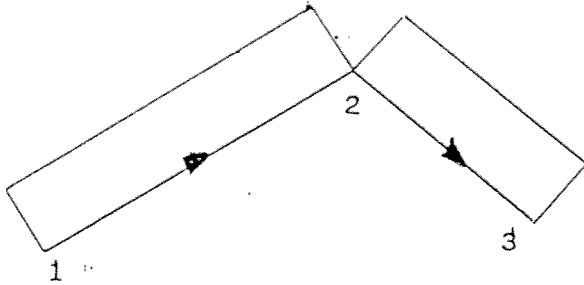
With the arrows (pointers) are marked the original contours. Each contours is assigned with starting point in (1), middle point in (2) and final point in (3).

In the picture, besides the original geometrical tracts, are drawn also the correspondent corrected tracts with indication of the geometrical structure used to determine the correct tracts.

The examples reported are not casual, but defined on two different cases :

- On the left side, the geometrical tracts corrected are detached, without a point of intersection : these are the cases in which the mill radius correction procedure inserts a connecting tract on the contour corrected so to connect the geometric tract corrected. With a correction of mill radius of positive sign, the connection has clockwise rotation direction ; with negative sign the connection has c.clockwise rotation direction.
- On the right side, the geometric tracts corrected intersect : in these cases, the contour corrected remains defined on two geometric tracts and the corrected contour middle point is assigned on the point of intersection of the two tracts corrected.

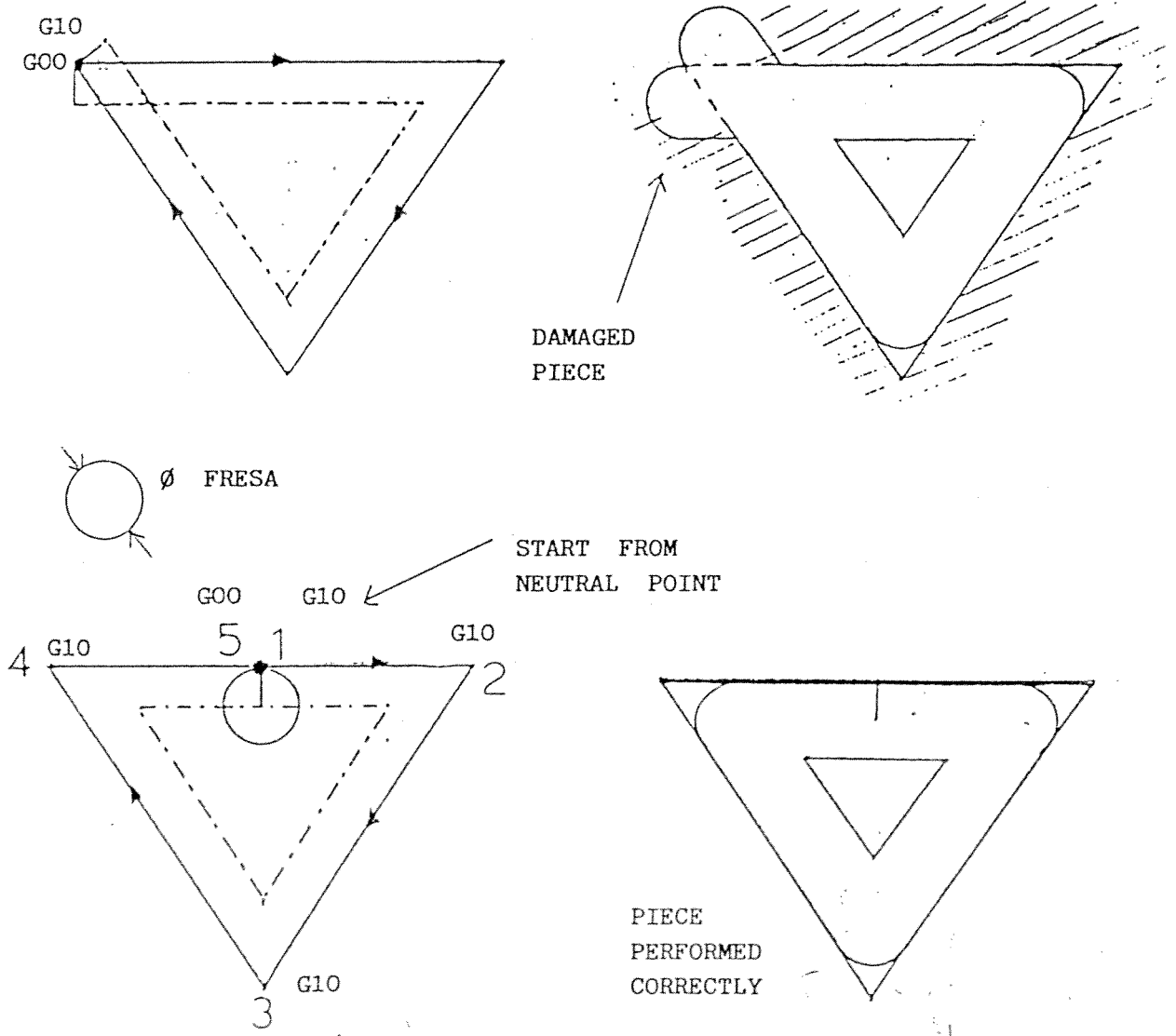
Picture : contours defined on two geometric tracts.



Mill radius correction on closed figures :

The program does not automatically recognize if a figure ends at its origin. When a figure has coincident starting/end points, it is important not to define as starting/end point a point in a corner, in order to avoid damaging the piece right on that point. A neutral point should be assigned as starting/end point.

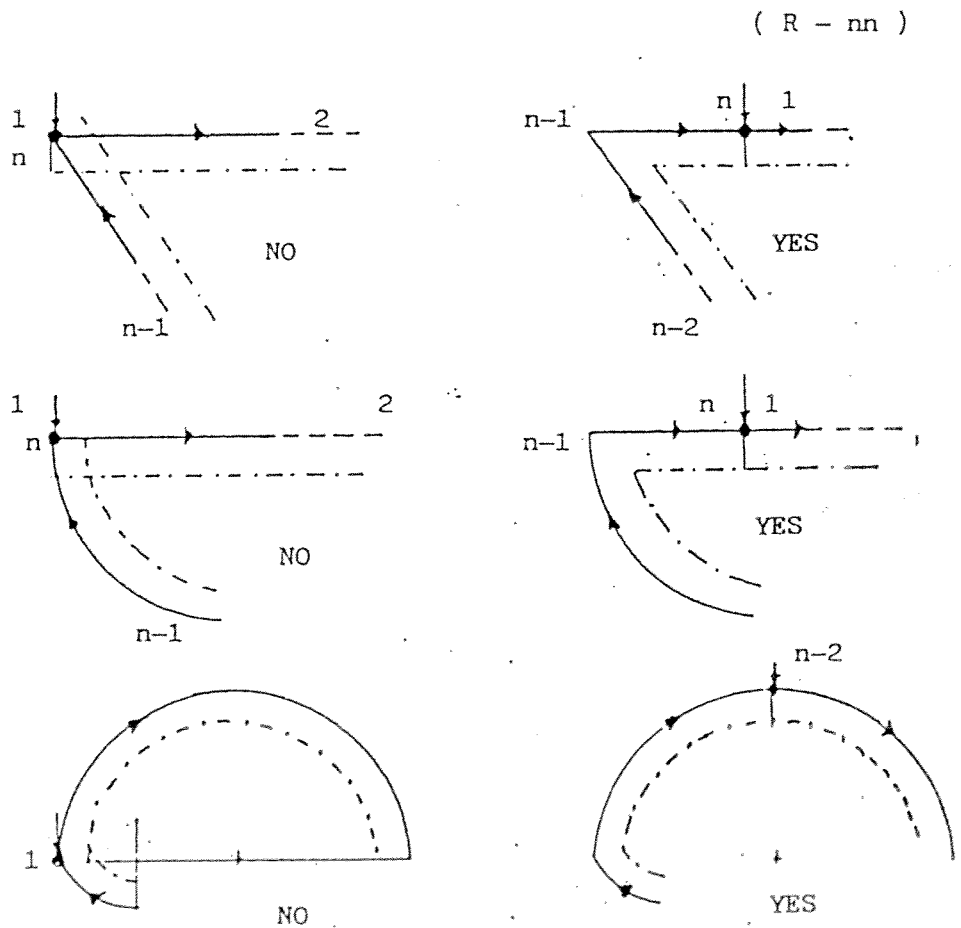
Picture 9.3 A, B



In the previous example can be noticed the two different ways to perform a triangle corrected from the inner side. In the first case - picture 9.3A - the mill descent take place in a corner of the figure (point of tangent variation) overlapping the piece when it lowers and raise. In this case is important to shift the starting/end point to a "neutral" point (point with tangent continuity on the point of starting and arrival (end)), as shown in picture 9.3B

In picture 9.4 are shown three examples in which, by placing the first point in a neutral point, the figure is performed correctly.

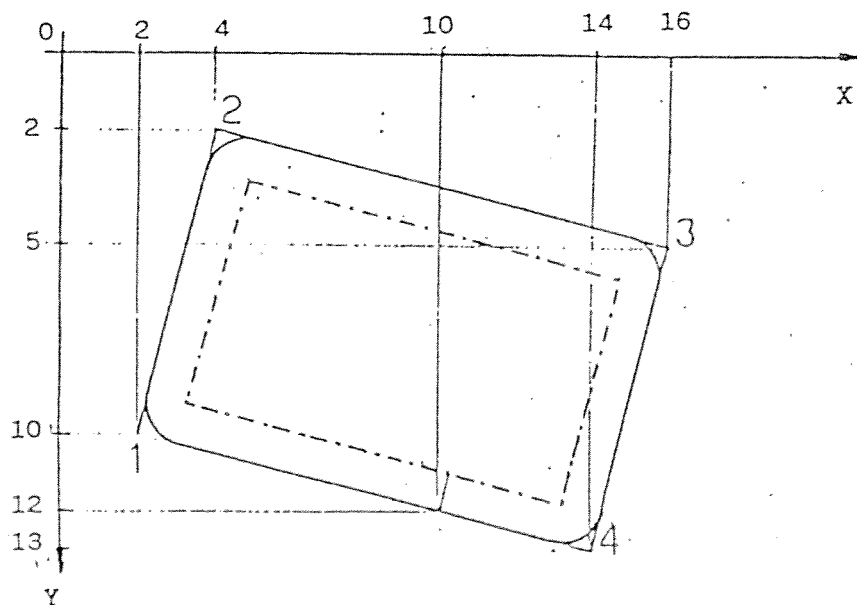
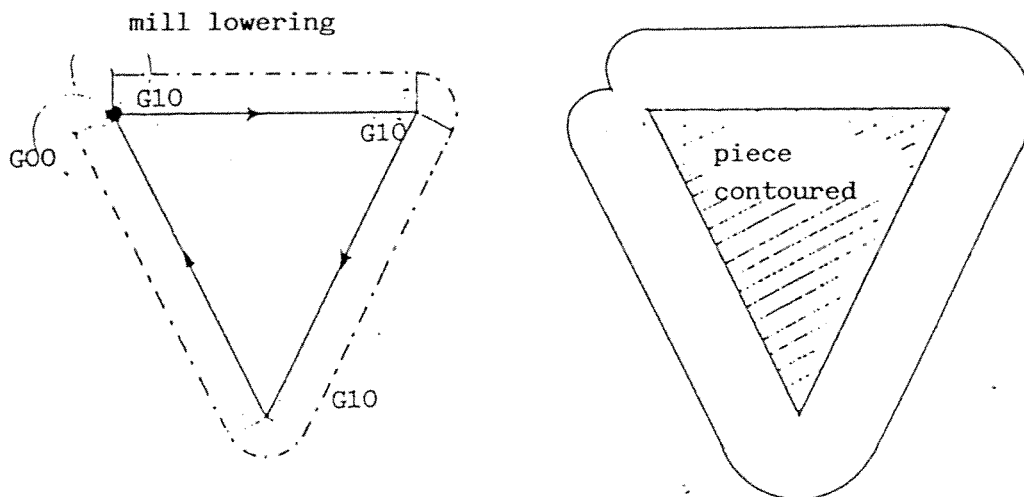
Picture 9.4



In outside contouring often it is not really relevant a connection between the start/end point, as the part to use is the inner part, and therefore the position of the lower/raise point makes no difference. (picture 9.5)

In picture 9.6 is possible to notice how in an oblique rectangle corrected from the inner side it has been necessary to add one more point A to avoid overlapping in the start/end point; point A is located on a linear tract so that it does not have tangent variation and the piece is performed correctly.

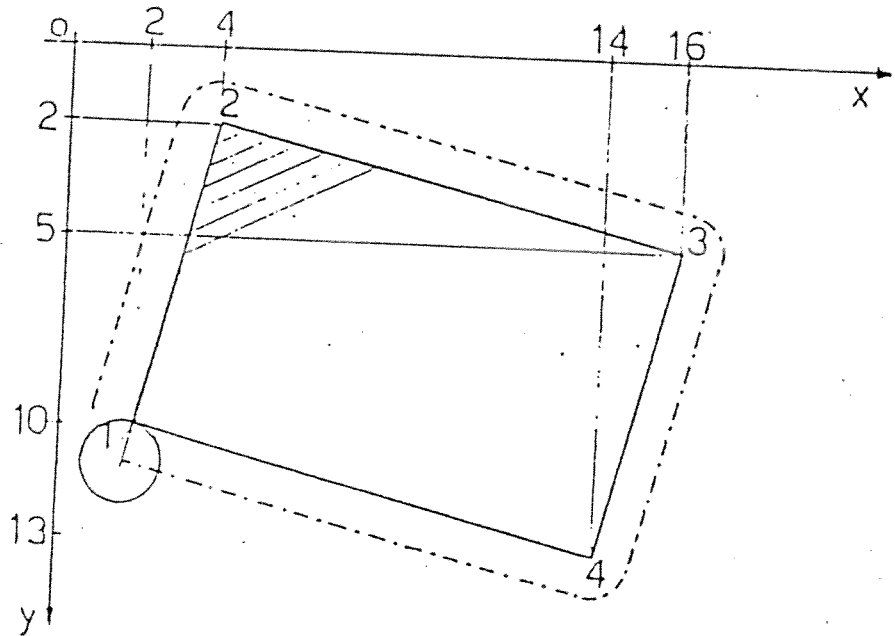
Pictures 9.5 e 9.6



— mill radius correction

In picture 9.7, the same rectangle as in picture 9.6 is performed with external contouring. In this case there is no need of adding any point : the figure is infact contoured correctly.

Picture 9.7



EXAMPLES

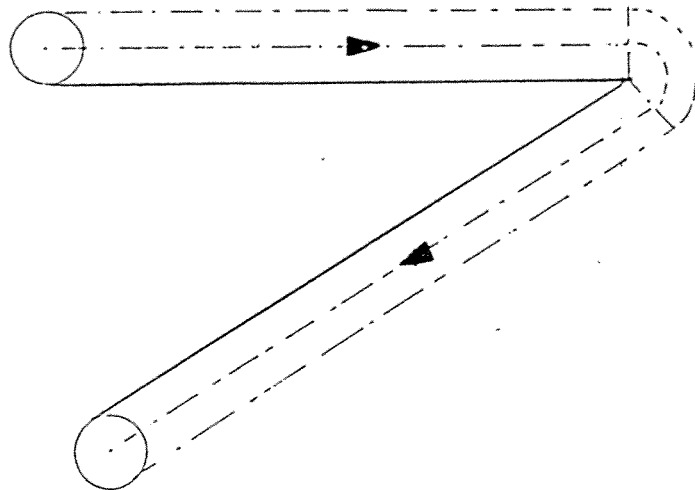
Left hand correction of oriented st.line

G10 R+ X100 Y100 Z10

G10 X400

G10 X100 Y300

:



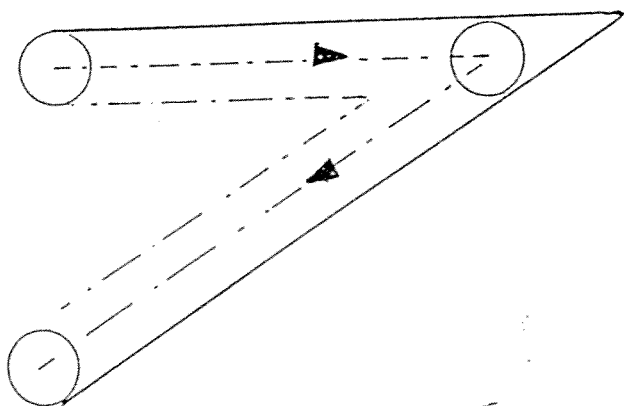
Right hand correction of oriented s.line

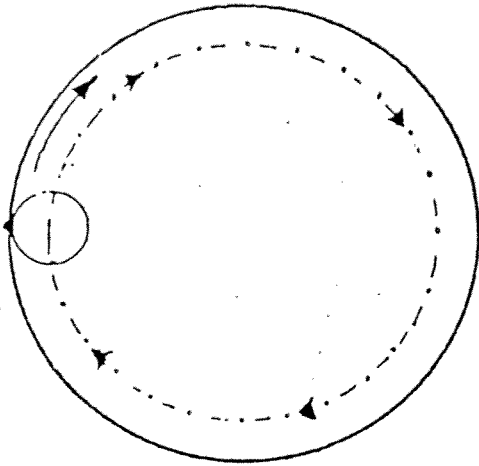
G10 R- X100 Y100 Z10

G10 X400

G10 X100 Y300

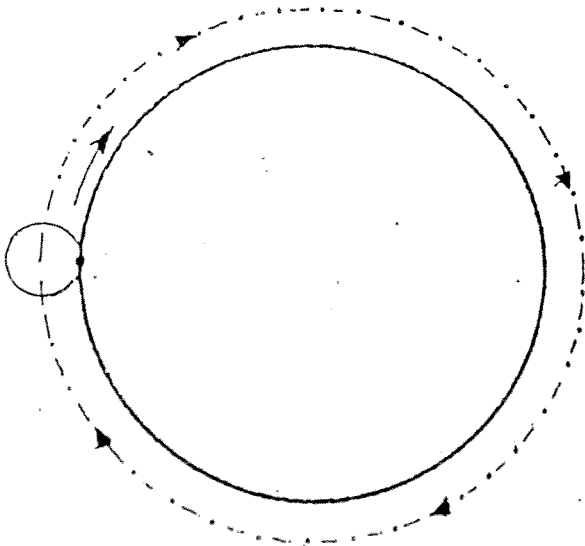
:





Right hand correction of clockwise circle.

```
G10 R- X200 Y200 Z10  
G02 X200 Y200 I300 J200
```



Left hand correction of clockwise circle.

```
G10 R+ X200 Y200 Z10  
G02 X200 Y200 I300 J200
```

As seen, to correct the milling tool radius could mean to add connections to the contour as programmed. On a contour however assigned, the total number of connections programmed can be equal to the total number of blocks programmed -1.

The following example shows the case of programming a closed figure with outer correction and with a special focus on the connections inserted.

The program realizing the figure in picture 9.8 does not close the corrected contour :the point of START/END (marked in the left side of the picture) is, on the contour programmed (inner), corner point and it is offset on the contour corrected (outer) and splitted in two different points.

The contour programmed is marked with continuous line, while the correction is dashed.

The points programmed are numbered from 1 to 11, plus 1 to close the contour, for a total of 12 points.

On each point programmed are marked the perpendicular trajectories to the contour, on the tract coming to and on the one going from the point itself; if the two tracts are tangent, the result is a single perpendicular (example : point 5).

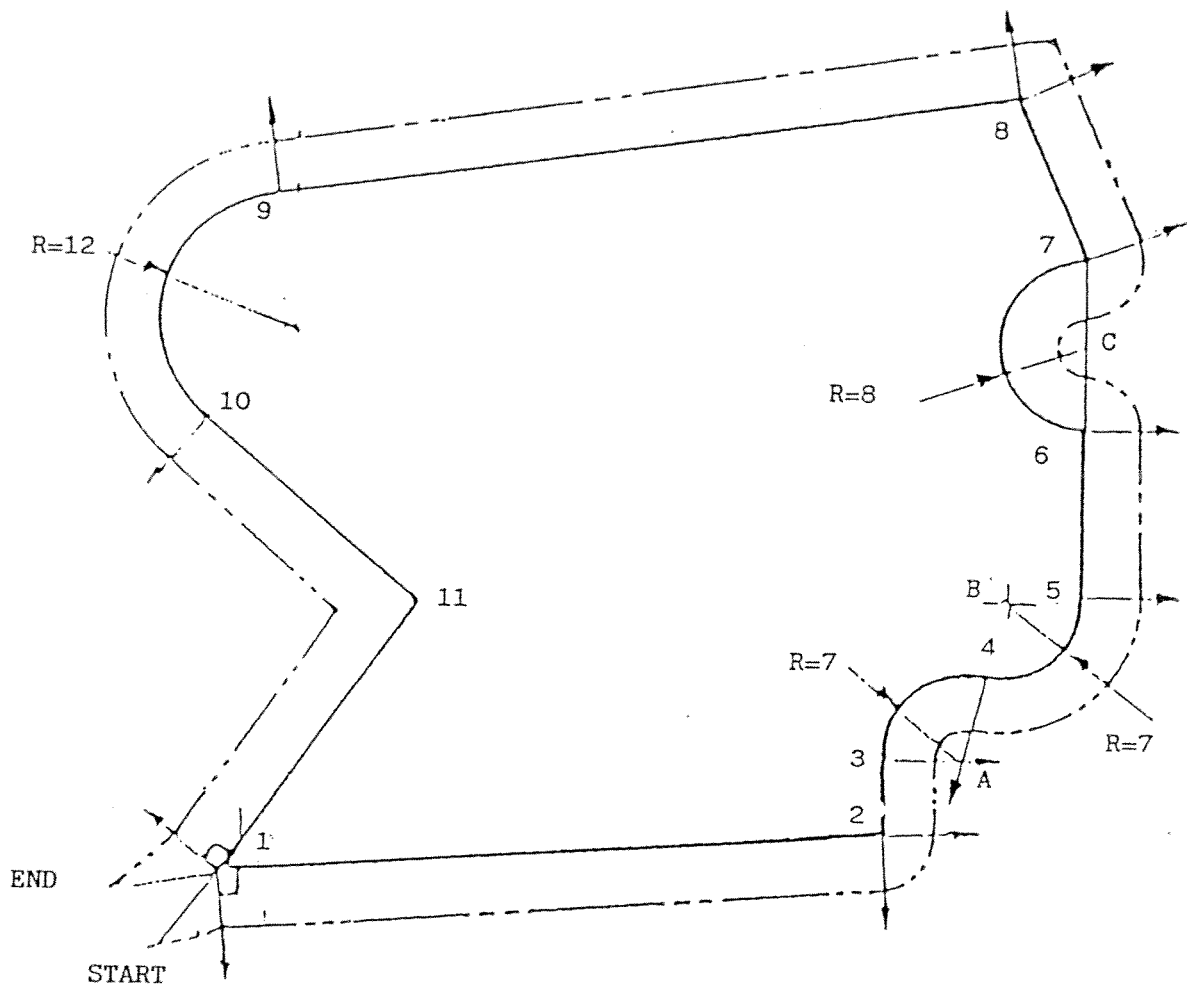
On the tracts of circular trajectory are marked rough values of the processing radius.

On the picture result inserted connections in the points :

- point 2 (connection between two linear segments)
- point 6 (connection between linear segment and arc)
- point 7 (connection between arc and linear segment)
- point 8 (connection between two linear segment).

As already said, the points of begin and end cut line on the corrected contour result distincts and they are localized by the two perpendiculars indicated on point 1, along the two linear tracts of begin and end of the contour programmed.

Picture 9.8



_____ mill radius correction _____

The connections inserted have curve radius equal to the mill radius.

When performing these connections it is applied an interpolation velocity proportional to the value of the mill radius, based on the criterion here exposed :

- It is assumed as reference the "reference interpolation velocity" as defined in "Parameters".
 - To this value is related the performance of arcs with curve radii majors than or equals to 100 mm.
 - Upon varying of the mill radius value to apply, a velocity is produced which :
1. equals to the reference interpolation velocity on the values major than or equals to 100 mm.

2. equals to
$$\frac{(\text{reference velocity}) * (\text{mill radius})}{100}$$

For mill radius values below 100 mm.

3. Upon insertion of a connection, it is imposed the velocity resulting the minor on the line preceding the connection among the velocity as above determined and the one entered in the program.

If the velocity obtained with points 1 and 2 results higher than the max. interpolation velocity (as defined in Parameters) it will be displayed the fault "VELOCITY NOT ALLOWED". The program line in which the error is displayed is the first line of the contour programmed with the mill radius correction required. (line of mill lowering).

To a variation of the value of reference interpolation velocity as entered in Parameters, should correspond a new recording of the programs in which the mill radius has been corrected.
(function STORE in mode Edit)

NOTE

The same remark is valid also when using R+/- : if the mill radius values entered in Parameters change, an adjustment of the programs to the new values takes place only after a new storing. (STORE of EDIT).

2. Correction of contour defined on one point.
The error is displayed as :

"Error 38 : GEOMETRIC ERROR IN MILL RADIUS CORRECTION
(sl -1010)"

^
|
The letters "sl" means that
the line in object is defined
on a straight line.

3. Correction of circular geometric tract, with radius smaller
than 0.5 mm.

The error is displayed as :

"Error 38 : GEOMETRIC ERROR IN MILL RADIUS CORRECTION
(a -2030)"

^
|
The letter "a" means that
the line in subject is
defined on a arc.

4. Correction of two consecutive linear segments, with length of
one segment smaller than the mill dimensions.

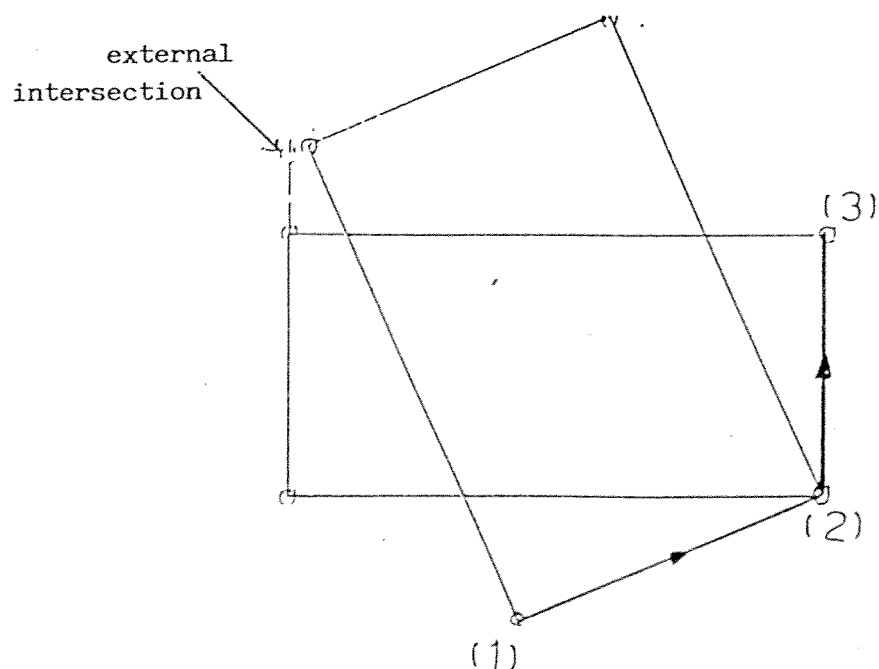
This case is shown in the picture.

The error is displayed in one of the two patterns :

"Error 56 : MILL RADIUS VALUE NOT ALLOWED (rr -3110)"
"Error 56 : MILL RADIUS VALUE NOT ALLOWED (rr -3120)"

according with the geometric area in subject.

PICTURE



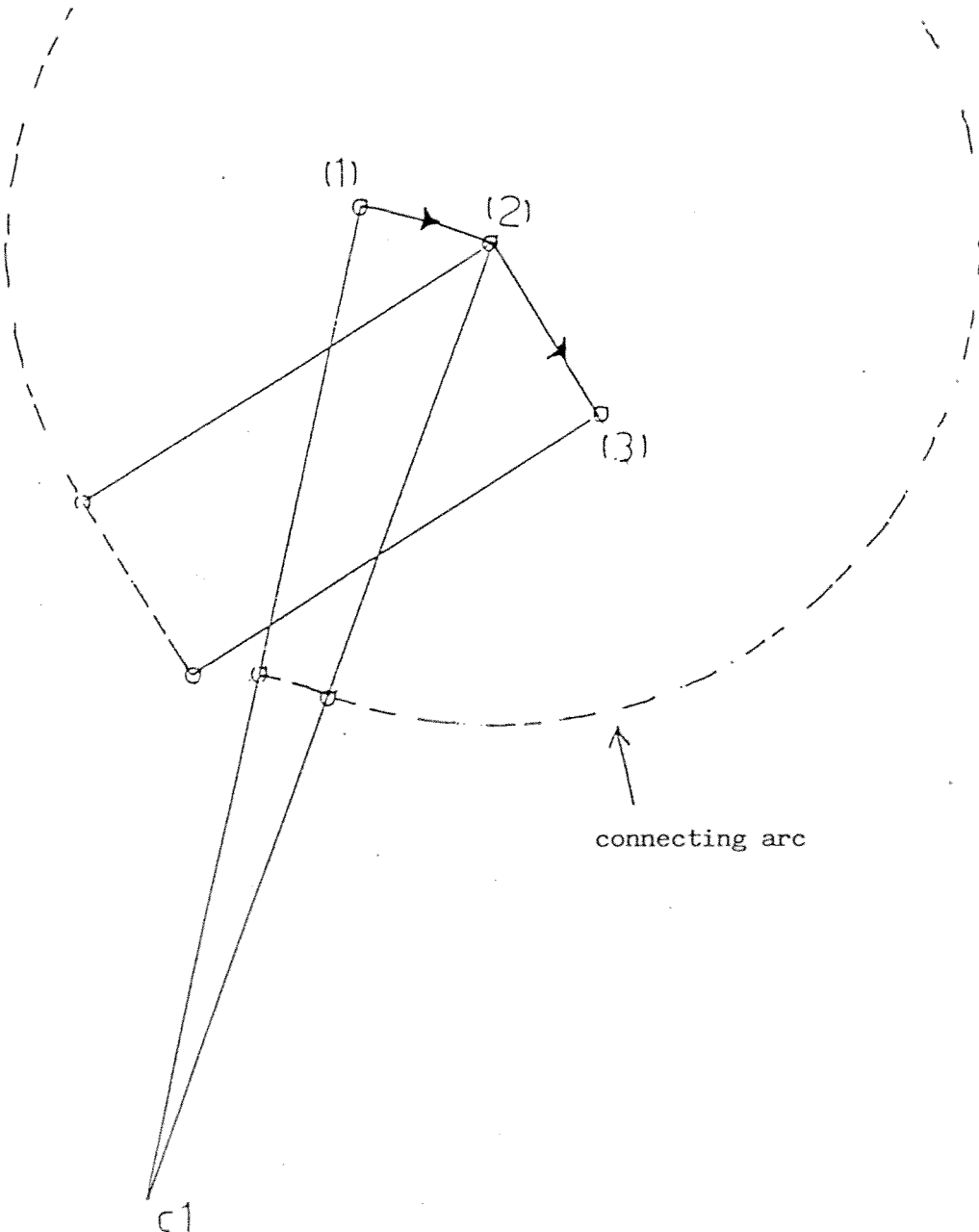
5. Correction between a linear geometric tract and a consecutive circular geometric tract (or vice versa), with solution on the case of connection, if the connection arc as reckoned is not reproducing the original contour. This case is depicted in the picture. The error is displayed as :

"Error 38 : GEOMETRIC ERROR IN MILL RADIUS CORRECTION
(sl-a -4660)"

(in case of straight line - arc)

"Error 38 : GEOMETRIC ERROR IN MILL RADIUS CORRECTION
(a-sl -4660)"

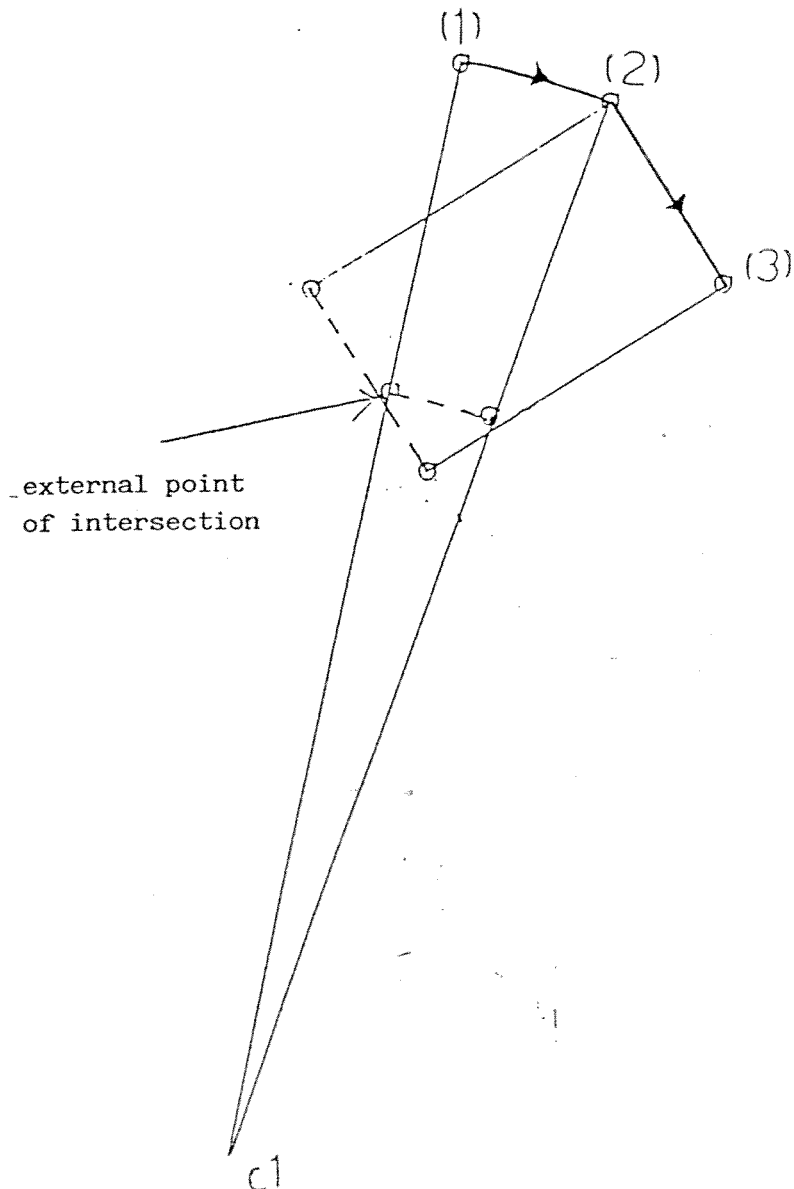
(in case of arc - straight line)



6. Correction between linear geometric tract and consecutive circular geometric tract (or vice versa), with solution on intersection case if the point of intersection lays external to the geometric tracts corrected.
The case is shown in the picture.
The error is displayed as :

"Error 38 : GEOMETRIC ERROR IN MILL RADIUS CORRECTION
(sl-a -5200)"
"Error 38 : GEOMETRIC ERROR IN MILL RADIUS CORRECTION
(sl-a -5205)"
"Error 38 : GEOMETRIC ERROR IN MILL RADIUS CORRECTION
(sl-a -5210)"
^

"sl-a" if straight line - arc
"a-sl" if arc - straight line



ERRORS DISPLAYED IN EDIT OF MILL RADIUS CORRECTION

Error number	Message and description
38	GEOMETRIC ERROR IN MILL RADIUS CORRECTION In the cases above examined.
51	VELOCITY NOT ALLOWED If the velocity entered on a connection is higher than the max. interpolation velocity entered in Parameters.
53	MILL RADIUS CORRECTION NOT ALLOWED (PLANE NOT XY) Function R of mill radius required on a group of millings not entirely defined on plane XY.
56	MILL RADIUS VALUE NOT ALLOWED In the cases above examined.

PERFORMING CONTOURING

The contouring defines a feature exploited when performing millings and depending both from the geometry of the contour as programmed, and from the programming modalities employed on the contour itself.

An instructions sequence of linear and circular interpolations performed without :

- acceleration and deceleration ramp
 - stopping on the connecting points
- defines a processing of contouring.

Consider a generical contour under milling :

```
G1Ø X(1) Y(1) Z(1) VX nn.n Ppp ...  
G.. X(2) Y(2) Z(2) VX nn.n Ppp ...  
.  
.  
GØ. X(n) Y(n) Z(n) VX nn.n Ppp
```

On each program line is assigned :

- Function G of interpolation selection (G1Ø , G12 ,...)
- Functions of values assignment (X, Y, Z, ...)
- Function P of interpolation plane selection (PXY, ..)
- Function VX to set the interpolation speed.

AUTOMATIC RUNNING OF CONTOURING

The control is able to run automatically the contouring when performing a contour milling however assigned.

When exploiting this possibility, consider the following criteria :

1. On each program line it is possible to vary the interpolation speed .
2. Between a milling and the next the continuity of movement is maintained, according with the interpolation speed assigned on each line, if conditions of "geometric continuity" between the millings in object take place. When performing a contouring, the speed max. gap allowed should be defined experimentally.

The condition of "geometric continuity" take place if the tangents in the connection point differs within an angle of max.10 degrees (however positioned in space). In a execution not in contouring, the velocity values programmed on each line keeps unchanged.

The situations geometrically possible to define a connection point on a contour in interpolation are here defined :

CASE 1 : straight line - arc (on a plane)

CASE 2 : arc (on a plane or segment of a helix) - straight line

CASE 3 : arc (on a plane or segment of a helix) - arc (on a plane)

CASE 4 : straight line - straight line

On these cases it is tested the condition of "geometrical continuity", with a confirmation if the two tangents in the connecting point result positioned within an angle, however positioned in the space, of 10 degrees.

The two tangent lines are reckoned on the connecting point :

In case of straight line, the tangent line coincides with the straight line itself, both if it defines the trajectory arriving to, or departing from, the connecting point.

The direction of the tangent straight line coincides with the direction of the milling on the linear segment.

In case of an arc, the tangent straight line coincide with the tangent to the arc in the connecting point. The direction of the tangent straight line follows the direction of the circular milling.

CASE 5 : straight line - arc (on a segment of helix)

CASE 6 : arc (on a plane or a segment of helix) - arc
(on a segment of helix)

These are cases with the connecting point in exit on a segment of helix.

The condition of geometric continuity is tested in a stricter way than in the previous cases. In fact, on the two tangent straight lines are not considered the components along axis Z : they are considered projected onto plane XY. The condition of geometric continuity is therefore tested in this way :

1. the two tangent straight lines, determined in the connecting point, should result distinct within an angle of 10 degrees.
2. The shiftings programmed on axis Z, concerning the two millings in object, should not determine displacements in opposite directions.

Appointing with z1 the displacement assigned to the milling arriving at the connecting point, and with z2 the displacement assigned to the milling departing from the connecting point (z1 and z2 are values increments with sign + or -) :

- the cases : $(z1 * z2) \Rightarrow \emptyset$ [read : if the result of z1 multiplied for z2 is positive or equals zero] suit the condition ;
- in case of result below zero (negative) the condition is not suited.

The geometric checking performed on the cases 5 and 6 results less restrictive than the checking carried over on the previous cases ; This control infact does not consider the displacement required on axis Z.

Programming on function C allows anyway to disable the contouring in connections with segments of helix (departing from the connecting point) when the condition of continuity is found insufficient.

FUNCTION "C" TO HANDLE PROGRAMMED CONTOURING

In programs' layouts it is always possible to qualify or disable the contouring in any connecting point of the contour under milling.

This possibility does not cut off the automatic contouring facility but complete it, as it will be examined here.

Programming mode : function C in a line of milling.

Syntax : Ggg Xnnn.n Ynnn.n Znn.n [Vnn] [R+-nn] [Cc]

where : on function C is accepted the programming of : CØ; C1; C2. Per default it is programmed CØ.

Function C programmed it is spreaded within a pattern of consecutive milling (until different specification), but no spreading takes place between a milling contour and the next.

The three possible modes are :

CØ qualifies automatic contouring from the point - of start milling or connection - in which it is programmed.

If a contour under milling is started without program on function C, this is the mode assumed per default (automatic running).

C1 qualifies contouring execution, from the point in which it is programmed. Once selected, C1 keeps activated until different specification, and only on the contour in object.

C2 disables contouring execution, from the point in which it is programmed. Once selected, C2 keeps activated until different specification, and only on the contour in object.

When a subprogram is loaded, the contouring is performed as it has been specified in the subprogram itself.

The following examples propose milling contours in programs with the contouring automatic running facility (function C is not used)

Example N. 1 (Picture 10.1)

1. G10 X100 Y100 Z10
2. X300
3. G12 X400 Y200 I300 J200
4. G10 X600 Y225
5. G12 X400 Y425 I400 J225
6. G00 X100 Y400
7. G10 X500 Y50
8. X725 Y75
9. G12 X800 Y150 I700 J175
10. X650 Y300 I650 J150
11. G00 X500 Y450

Following, are reported parts of the text "disassembled" (see Appendix E : Auxiliary programs of debug) when the contours under milling are performed. The references to the programs addressing have been deleted, as they were irrelevant in the analysis in course.

```
{ Program start : function "start program" in the work field
  selected ; enter the processing velocities };
{ Begin the first contour in interpolation : initial positioning
  of axes X, Y, Z (as in line 1 of the source program) and
  qualification of tool type J } ;
```

```
INFRE (1)
LINEAR XY RATEIC: 0 RESOLI: 4 RESOLK: 4 RESOLJ: 4
D1: 200 D2: 0 D3: 0 R6: 200
FIFRE (2)
ARC XY CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
QXC: 0 QYC: 100 QXF: 100 QYF: 100 RADIUS: 100
LINEAR XY RATEIC: 0 RESOLI: 4 RESOLK: 4 RESOLJ: 4 (3)
D1: 200 D2: 25 D3: 0 R6: 201.4
INFRE (4)
ARC XY CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
QXC: -200 QYC: 0 QXF: -200 QYF: 200 RADIUS: 200
FIFRE (5)
LINEAR XY RATEIC: 0 RESOLI: 4 RESOLK: 4 RESOLJ: 4
D1: -300 D2: -25 D3: 0 R6: 301
```

```
{ End of the contour under milling : wait arrival on the final
  value of the axes in interpolation (as line 6 of the source
  program); disabling of tool type J } ;
```

{ Begin the second contour in interpolation : initial positioning of axes X, Y, Z (as in line 7 of the source program) and qualification of tool type J } ;

```

INFRE                                                    (6)
LINEAR XY RATEIC: Ø RESOLI: 4 RESOLK: 4 RESOLJ: 4
  D1: 225 D2: 25 D3: Ø RG: 226.2
FIFRE
ARC XY CLOCKWISE RATEIC: Ø RESOLI: 4 RESOLK: 4
  QXC: -25 QYC: 100 QXF: 75 QYF: 75 RADIUS: 103
ARC XY CLOCKWISE RATEIC: Ø RESOLI: 4 RESOLK: 4
  QXC: -150 QYC: Ø QXF: -150 QYF: 150 RADIUS: 150
LINEAR XY RATEIC: Ø RESOLI: 4 RESOLK: 4 RESOLJ: 4
  D1: -150 D2: 150 D3: Ø RG: 212
  
```

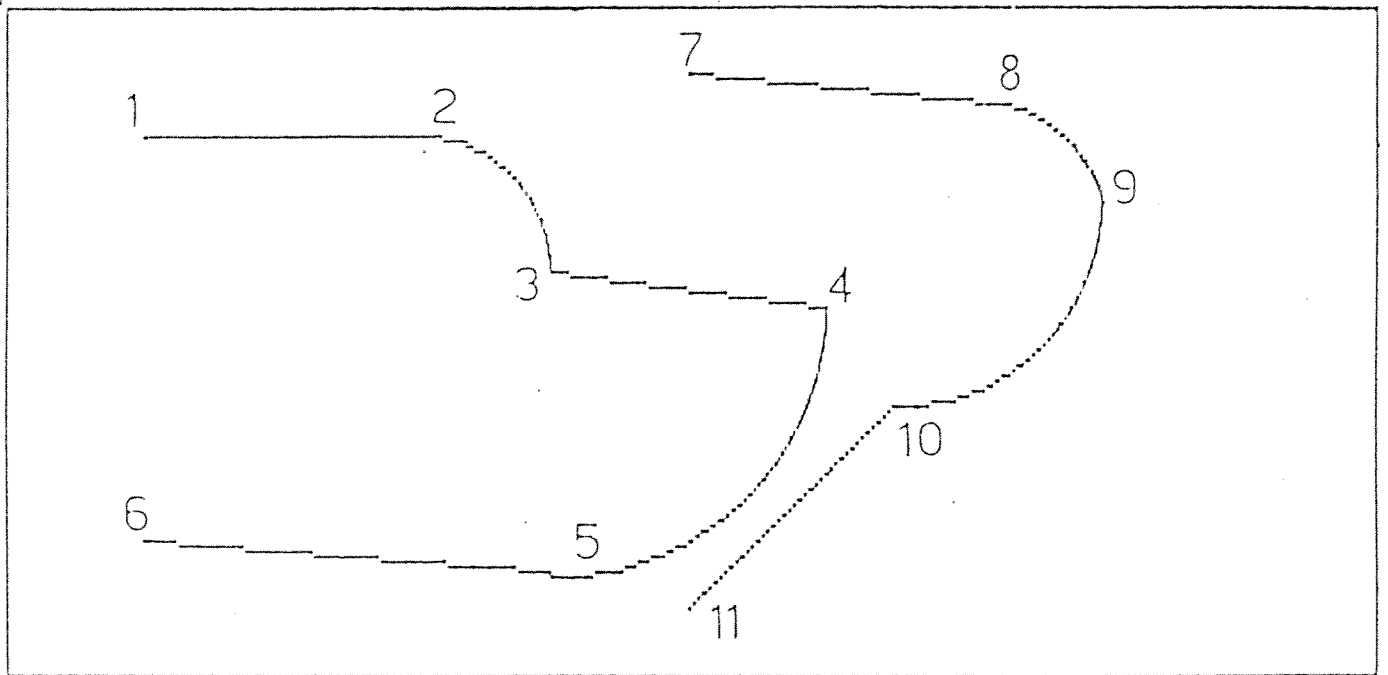
{ End of the contour under milling : wait arrival on the final value of the axes in interpolation (as in line 11 of the source program) and disabling of tool type J } ;

{ End of the program : function of "end program " on the work field selected } .

NOTES

- (1) The instruction INFRE qualifies the contouring when performing the connection in point 2 (connection between line and arc) : the tangent in point 2 in fact coincides with the trajectory of arrival (from 1 and 2) and of departure (from 2 to 3).
- (2) The instruction FIFRE disables the contouring from the arriving point of the interpolation instruction following (in this case it is : ARC XY ...) : in picture 10.1 is clear the corner in point 3. In this point, therefore, the axes arrive with handling of the deceleration/stop ramp. The starting, toward point 4, takes place with the acceleration ramp.
- (3) Instruction LINEAR : performs a linear interpolation from point 3 to point 4, with stop also on point 4 (being performed in FIFRE)
- (4) INFRE recognize geometric continuity on point 5 (connection between arc and line).
- (5) The instruction FIFRE disables however the contouring of an end (final) point of a contour in interpolation.
- (6) In the second contour programmed (from line 7 to line 11 of the source program) it is carried out only the contouring of point 8 (first connecting point of the contour) : connection between line and arc.

PICTURE 10.1



Example N. 2 (Picture 10.2)

1. G10 X100 Y300 Z10
2. Y200
3. G12 X200 Y100 I200 J200
4. G10 X400
5. G12 X500 Y200 I400 J200
6. G10 X475 Y475
7. G13 X625 Y325 I475 J325
8. G12 X725 Y225 I725 J325
9. G13 X825 Y125 I725 J125
10. G00 X500 Y100

Text "disassembled", when running the milling :

{ Start program : function "start program " in the selected work field ; enter the processing velocities } ;
 { Begin the contour in interpolation : initial positioning of axes X, Y, Z (as in line 1 of the source program) and qualification of tool type J } ;

```

INFRE (1)
LINEAR XY RATEIC: 0 RESOLI: 4 RESOLK: 4 RESOLJ: 4
D1: 0 D2: -100 D3: 0 RG: 100
ARC XY CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
QXC: 100 QYC: 0 QXF: 100 QYF: -100 RADIUS: 100
LINEAR XY RATEIC: 0 RESOLI: 4 RESOLK: 4 RESOLJ: 4
D1: 200 D2: 0 D3: 0 RG: 200
ARC XY CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
QXC: 0 QYC: 100 QXF: 100 QYF: 100 RADIUS: 100
FIFRE (2)
LINEAR XY RATEIC: 0 RESOLI: 4 RESOLK: 4 RESOLJ: 4
D1: -25 D2: 275 D3: 0 RG: 276
INFRE (3)
ARC XY C.CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
QXC: 0 QYC: -150 QXF: 150 QYF: -150 RADIUS: 150
ARC XY CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
QXC: 100 QYC: 0 QXF: 100 QYF: -100 RADIUS: 100
FIFRE (4)
ARC XY C.CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
QXC: 0 QYC: -100 QXF: 100 QYF: -100 RADIUS: 100
LINEAR XY RATEIC: 0 RESOLI: 4 RESOLK: 4 RESOLJ: 4
D1: -325 D2: -25 D3: 0 RG: 325.8
    
```

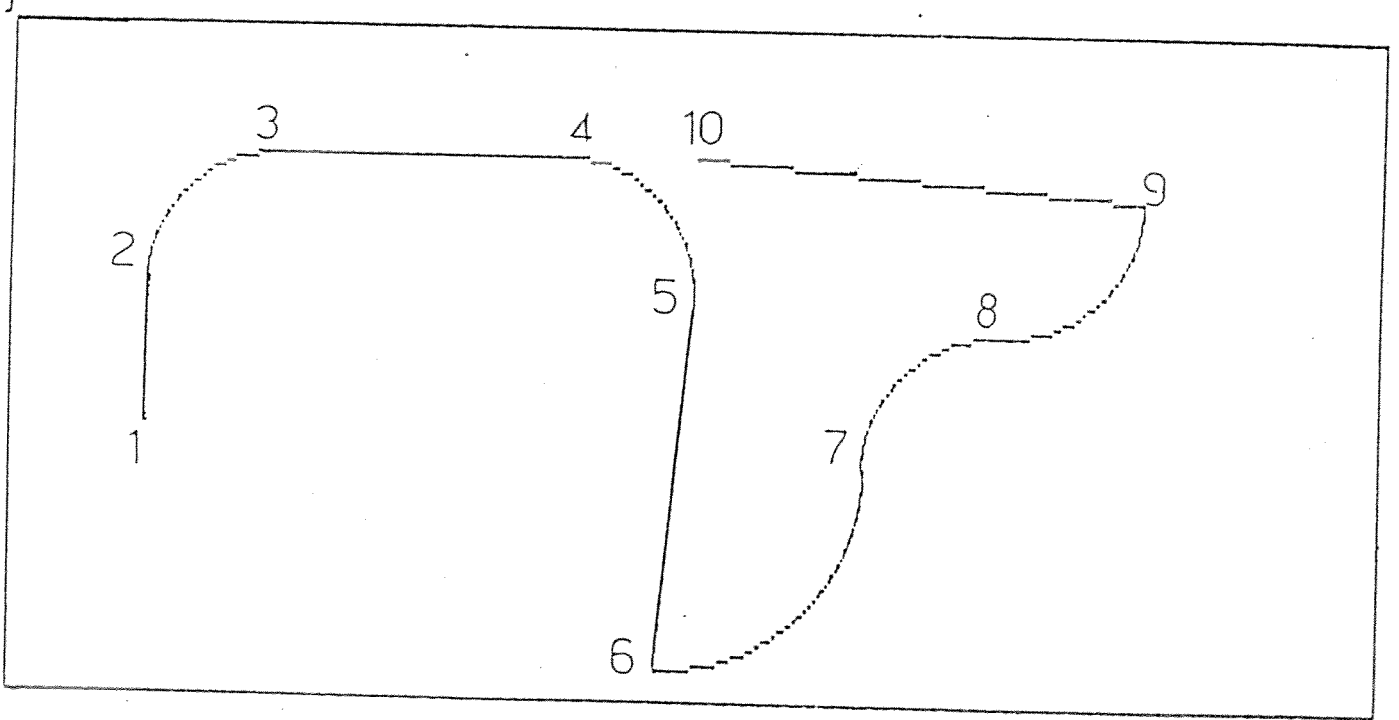
{ End of the contour under milling : wait arrival to end value of the axes in interpolation (as in line 10 of source program); disabling of tool type J } ;

{ End of the program : function "end program" in the work field selected } .

NOTES

- (1) The instruction INFRE qualifies the contouring when performing the connection in point 2 (connection between line and arc). The contouring keeps qualified until the second instruction "ARC", in the connection on point 5.
- (2) In the corner in point 6 the axes under processing are stopped, to start again on the arc toward point 7.
- (3) The instruction INFRE before the instruction ARC qualifies the contouring on the connections in points 7 and 8.
- (4) Disables the contouring of the corner in 9 and till the end of the contour in interpolation.

Picture 10.2



Example N. 3 (Picture 10.3)

1. G10 X0 Y300 Z15 PXZ
2. Z5
3. G12 X5 Z0 I5 K5
4. G10 X300
5. G12 X400 Y400 I300 J400 PXY
6. G10 Y495
7. G12 Y500 Z5 J495 K5 PZY
8. G00 Z15

Unlike the two previous examples, the milling profile is here performed on two different planes.

This makes the "disassembled" text different in respect to the source text, for what concerns the interpolation instructions (linear and circular) used : every interpolation instruction is in fact reported to the processing on the three axes qualified to interpolate : X, Y, Z , in this way :

- In linear interpolations is employed the ins. LINEAR on XYZ ;
- In circular interpolations is employed the ins. HELIX, with circular development on the two moving axes, and helix' axis along the third axis (still) ;
- In the helicoidal interpolations programmed on the source text, it is employed the instruction HELIX, as before, with circular development on plane XY and helix' axis along axis Z.

The conversion of every interpolation on area XYZ is carried over with any contour programmed with variation, in the contour itself, of the interpolation plane ; this is necessary in order to handle the contouring on the programs to be run.

To have this clear makes possible to read the "disassembled text" reported in the next page.

```
{ Start program : function "start program" in the work field
  selected. Enter the processing velocities };
{ Begin the contour in interpolation : initial positioning of
  axes X, Y, Z (as in line 1 of the source program) and
  qualification of tool type J } ;
```

```
INFRE
LINEAR XYZ RATEIC: 0 RESOLI: 4 RESOLK: 4 RESOLJ: 4
  D1: 0 D2: 0 D3: -10 RG: 10
HELIX (XZ) Y CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
  RESOLJ: 4 DY: 0 DCIR: 8 DIAG: 8.2 QXC: 5 QZC: 0
  QXF: 5 QZF: -5 RG: 5
LINEAR XYZ RATEIC: 0 RESOLI: 4 RESOLK: 4 RESOLJ: 4
  D1: 295 D2: 0 D3: 0 RG: 295
HELIX (XY) Z CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
  RESOLJ: 4 DZ: 0 DCIR: 157.2 DIAG: 157.4 QXC: 0 QYC: 100
  QXF: 100 QYF: 100 RG: 100
LINEAR XYZ RATEIC: 0 RESOLI: 4 RESOLK: 4 RESOLJ: 4
  D1: 0 D2: 95 D3: 0 RG: 95
HELIX (YZ) X CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
  RESOLJ: 4 DX: 0 DCIR: 8 DIAG: 8.2 QYC: 0 QZC: 5
  QYF: 5 QZF: 5 RG: 5
```

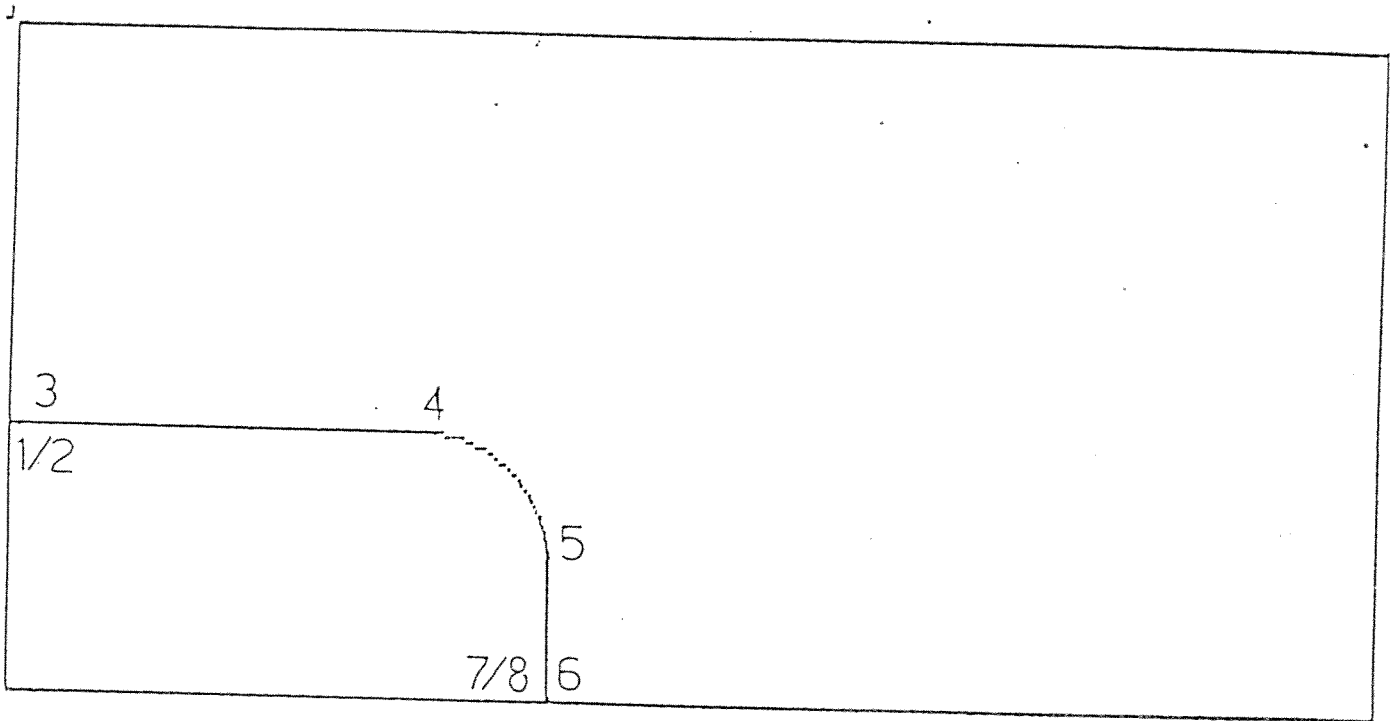
```
FIFRE
LINEAR XYZ RATEIC: 0 RESOLI: 4 RESOLK: 4 RESOLJ: 4
  D1: 0 D2: 0 D3: 10 RG: 10
```

```
{ End of the contour under milling : wait arrival on the final
  value of the axes in interpolation (as in line 8 of the source
  program) ; disabling of tool type J } ;
```

```
{ End of the program : function "end program" in the work field
  selected } .
```

The entire contour programmed has been contoured : the instruction FIFRE is displayed, infact, only before the last instruction of interpolation to disable the contouring once reached the final point.

Picture 10.3



It is here proposed a contour to be milled in different programs of the source text, using function C.

Example N. 4 (Picture 10.4)

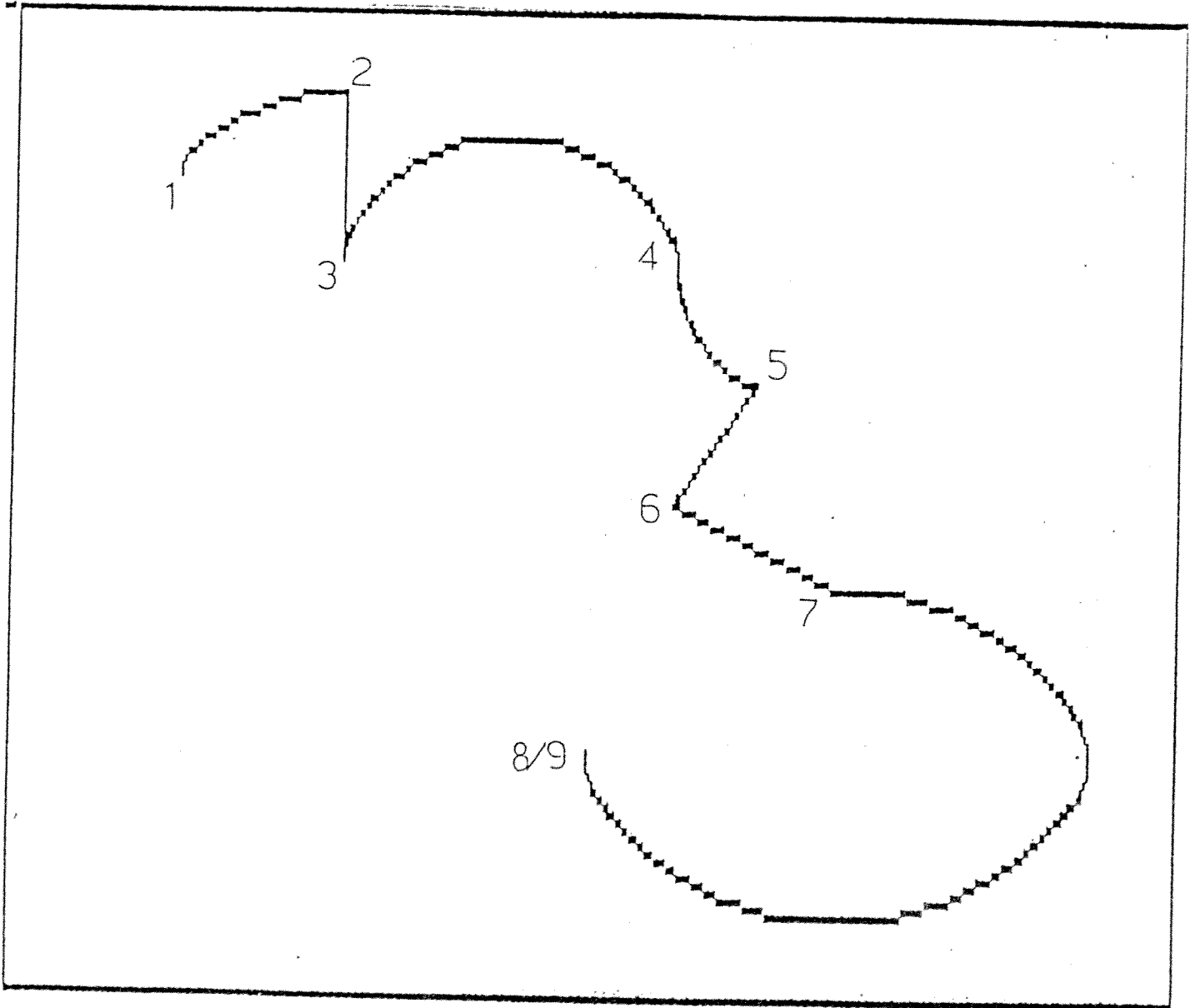
On the contour as in picture 10.4, are proposed three source texts differing for the kind of contouring. On the first text reported, function C has not been used: the contouring is performed automatically.

1. G10 X100 Y100 Z10
2. G14 X200 Y50 I100 J50 A20
3. G10 Y150
4. G14 X400 I100 J75 A30
5. G15 X450 Y225 I50 J75 A15
6. G10 X400 Y300
7. X500 Y350
8. G14/3 X350 Y450 I150 J100 A50
9. G00

On the second text reported, function C appears on line number 5. (C2 : disables the contouring, in this case until the end of the contour programmed) :

1. G10 X100 Y100 Z10
2. G14 X200 Y50 I100 J50 A20
3. G10 Y150
4. G14 X400 I100 J75 A30
5. G15 X450 Y225 I50 J75 A15 C2
6. G10 X400 Y300
7. X500 Y350
8. G14/3 X350 Y450 I150 J100 A50
9. G00

Picture 10.4



On the third text reported : function C is displayed in the lines 4 (C2 : disables contouring), 5 (C0 : qualifies automatic control of contouring), 7 (C1 : obliges the performing of contouring, in this case until the end of the contour programmed) :

1. G10 X100 Y100 Z10
2. G14 X200 Y50 I100 J50 A20
3. G10 Y150
4. G14 X400 I100 J75 A30 C2
5. G15 X450 Y225 I50 J75 A15 C0
6. G10 X400 Y300
7. X500 Y350 C1
8. G14/3 X350 Y450 I150 J100 A50
9. G00

The texts programmed set in evidence also the application of function C on functions that implies the source text expansion on a bigger number of execution lines. In the example : to every function G14 or G15 corresponds the execution of more circular interpolation instructions. Similar situations happens when programming :

For what concerns the contouring : on the added circular interpolations (arcs on oval, simplified connections, connection upon mill radius correction), it is applied function C valid on the correspondent source text instruction.

The "disassembled" texts of the three program versions proposed are shown in the next page.
The functions running the execution of arc of oval are pointed out as groups with indication of the source text line number to which the group refers.

FIRST TEXT PROPOSED

{ Start program : function "start program" on the work field selected. Enter the processing velocities } ;
 { Begin the contour in interpolation : initial positioning of axes X, Y, Z (as in line 1 of source text) and qualification of tool type J } ;

```

INFRE
2  ARC      XY  CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
   QXC: 20 QYC: 0 QXF: 6.8 QYF: -15.2 RADIUS: 20
FIFRE
4  ARC      XY  CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
   QXC: 93.2 QYC: 106.8 QXF: 93.2 QYF: -34.8 RADIUS: 141.6
LINEAR XY  RATEIC: 0 RESOLI: 4 RESOLK: 4 RESOLJ: 4
   D1: 0 D2: 100 D3: 0 RG: 100
INFRE
4  ARC      XY  CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
   QXC: 30 QYC: 0 QXF: 2.6 QYF: -12.6 RADIUS: 30
   ARC      XY  CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
   QXC: 97.4 QYC: 44.4 QXF: 194.6 QYF: 0 RADIUS: 106.8
   ARC      XY  CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
   QXC: -27.2 QYC: 12.6 QXF: 2.6 QYF: 12.6 RADIUS: 29.8
5  ARC      XY  ANTIOR. RATEIC: 0 RESOLI: 4 RESOLK: 4
   QXC: 83.8 QYC: 0 QXF: 42.6 QYF: 73 RADIUS: 83.8
FIFRE
4  ARC      XY  C.CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
   QXC: 7.4 QYC: -13 QXF: 7.4 QYF: 1.8 RADIUS: 14.8
LINEAR XY  RATEIC: 0 RESOLI: 4 RESOLK: 4 RESOLJ: 4
   D1: -50 D2: 75 D3: 0 RG: 90
LINEAR XY  RATEIC: 0 RESOLI: 4 RESOLK: 4 RESOLJ: 4
   D1: 100 D2: 50 D3: 0 RG: 111.8
INFRE
8  ARC      XY  CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
   QXC: 0 QYC: 175 QXF: 140 QYF: 70 RADIUS 175
   ARC      XY  CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
   QXC: -40 QYC: 30 QXF: 0 QYF: 60 RADIUS: 50
   ARC      XY  CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
   QXC: -140 QYC: -105 QXF: -280 QYF: 0 RADIUS: 175
FIFRE
4  ARC      XY  CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
   QXC: 40 QYC: -30 QXF: -10 QYF: -30 RADIUS: 50
    
```

{ End of the contour under milling : wait arrival on end value of the axes in interpolation (as in line 9 of source program). Disabling of tool type J } ;
 { End of the program : function "end program" on the work field selected } .

The automatic running of the contouring sets in evidence the corner points : points 2 , 3 , 5 , 6 , 7 .
The arcs of oval are obviously performed in contouring, having been determined on the condition of geometric continuity in the variation points of the interpolation radius.

SECOND TEXT PROPOSED

{ Start program : function "Start program" in the work field selected. Enter processing velocities };
{ Begin the contour in interpolation : initial positioning of axes X, Y, Z (as in line 1 of source program). ; qualification of tool type J } ;

```

INFRE
2  [ ARC XY CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
    QXC: 20 QYC: 0 QXF: 6.8 QYF: -15.2 RADIUS: 20
  FIFRE
  [ ARC XY CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
    QXC: 93.2 QYC: 106.8 QXF: 93.2 QYF: -34.8 RADIUS: 141.6
  LINEAR XY RATEIC: 0 RESOLI: 4 RESOLK: 4 RESOLJ: 4
    D1: 0 D2: 100 D3: 0 RG: 100
  INFRE
  [ ARC XY CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
    QXC: 30 QYC: 0 QXF: 2.6 QYF: -12.6 RADIUS: 30
  4 [ ARC XY CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
    QXC: 97.4 QYC: 44.4 QXF: 194.6 QYF: 0 RADIUS: 106.8
  [ ARC XY CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
    QXC: -27.2 QYC: 12.6 QXF: 2.6 QYF: 12.6 RADIUS: 29.8
  FIFRE
  [ ARC XY C.CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
  5 [ QXC: 83.8 QYC: 0 QXF: 42.6 QYF: 73 RADIUS: 83.8
  [ ARC XY C.CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
    QXC: 7.4 QYC: -13 QXF: 7.4 QYF: 1.8 RADIUS: 14.8
  LINEAR XY RATEIC: 0 RESOLI: 4 RESOLK: 4 RESOLJ: 4
    D1: -50 D2: 75 D3: 0 RG: 90
  LINEAR XY RATEIC: 0 RESOLI: 4 RESOLK: 4 RESOLJ: 4
    D1: 100 D2: 50 D3: 0 RG: 111.8
  [ ARC XY CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
  [ QXC: 0 QYC: 175 QXF: 140 QYF: 70 RADIUS: 175
  8 [ ARC XY CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
    QXC: -40 QYC: 30 QXF: 0 QYF: 60 RADIUS: 50
  [ ARC XY CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
    QXC: -140 QYC: -105 QXF: -280 QYF: 0 RADIUS: 175
  [ ARC XY CLOCKWISE RATEIC: 0 RESOLI: 4 RESOLK: 4
    QXC: 40 QYC: -30 QXF: -10 QYF: -30 RADIUS: 50

```

{ End of contour under milling : wait arrival to the final value of the axes in interpolation (as in line 9 of the source program); disabling of tool type J } ;
{ End of the program : function "end program" in the work field selected } .

Automatic running of contouring keeps qualified till the connection in point 4.

Function C2 programmed on line 5 of the source text disables contouring from line 5 of the source text until the end of the contour programmed.

Note how the selection on C2 keeps operative on the whole arc of oval programmed in line 5.

THIRD TEXT PROPOSED

{ Start program : function "start program" in the work field selected ; enter processing velocities };

{ Begin the contour in interpolation : initial positioning of axes X, Y, Z (as in line 1 of the source text) ; qualification of tool type J }

```

INFRE
[ ARC XY CLOCKWISE RATEIC: Ø RESOLI: 4 RESOLK: 4
  QXC: 2Ø QYC: Ø QXF: 6.8 QYF: -15.2 RADIUS: 2Ø
2 FIFRE
[ ARC XY CLOCKWISE RATEIC: Ø RESOLI: 4 RESOLK: 4
  QXC: 93.2 QYC: 1Ø6.8 QXF: 93.2 QYF: -34.8 RADIUS: 141.6
  LINEAR XY RATEIC: Ø RESOLI: 4 RESOLK: 4 RESOLJ: 4
  D1: Ø D2: 1ØØ D3: Ø R6: 1ØØ
[ ARC XY CLOCKWISE RATEIC: Ø RESOLI: 4 RESOLK: 4
  QXC: 3Ø QYC: Ø QXF: 2.6 QYF: -12.6 RADIUS: 3Ø
4 ARC XY CLOCKWISE RATEIC: Ø RESOLI: 4 RESOLK: 4
  QXC: 97.4 QYC: 44.4 QXF: 194.6 QYF: Ø RADIUS: 1Ø6.8
[ ARC XY CLOCKWISE RATEIC: Ø RESOLI: 4 RESOLK: 4
  QXC: -27.2 QYC: 12.6 QXF: 2.6 QYF: 12.6 RADIUS: 29.8
  INFRE
[ ARC XY C.CLOCKWISE RATEIC: Ø RESOLI: 4 RESOLK: 4
  QXC: 83.8 QYC: Ø QXF: 42.6 QYF: 73 RADIUS: 83.8
5 FIFRE
[ ARC XY C.CLOCKWISE RATEIC: Ø RESOLI: 4 RESOLK: 4
  QXC: 7.4 QYC: -13 QXF: 7.4 QYF: 1.8 RADIUS: 14.8
  LINEAR XY RATEIC: Ø RESOLI: 4 RESOLK: 4 RESOLJ: 4
  D1: -5Ø D2: 75 D3: Ø R6: 9Ø
  INFRE
  LINEAR XY RATEIC: Ø RESOLI: 4 RESOLK: 4 RESOLJ: 4
  D1: 1ØØ D2: 5Ø D3: Ø R6: 111.8
[ ARC XY CLOCKWISE RATEIC: Ø RESOLI: 4 RESOLK: 4
  QXC: Ø QYC: 175 QXF: 14Ø QYF: 7Ø RADIUS: 175
  ARC XY CLOCKWISE RATEIC: Ø RESOLI: 4 RESOLK: 4
  QXC: -4Ø QYC: 3Ø QXF: Ø QYF: 6Ø RADIUS: 5Ø
8 ARC XY CLOCKWISE RATEIC: Ø RESOLI: 4 RESOLK: 4
  QXC: -14Ø QYC: -1Ø5 QXF: -28Ø QYF: Ø RADIUS: 175
  FIFRE
[ ARC XY CLOCKWISE RATEIC: Ø RESOLI: 4 RESOLK: 4
  QXC: 4Ø QYC: -3Ø QXF: -1Ø QYF: -3Ø RADIUS: 5Ø

```

_____ contouring _____

{ End of the contour under milling : wait arrival to final value of axes in interpolation (as line 9 of the source text); disabling of tool type J } ;

{ End of the program : function "end program" on the work field selected } .

Automatic running of contouring keeps qualified till the connection in point 3.

Function C2 programmed in line 4 of the source text disables the contouring of the arc of oval programmed in that same line.

Function C0 in line 5 qualifies the Automatic running of contouring according with the geometric contour programmed.

Function C1 in line 7 imposes the contouring until the end of the contour.

PROGRAMMING PARAMETERS

In the stage of program Edit it is possible to exploit the facility of parametric values programming. That means that, instead of numbers to appoint the required values, it is possible to enter formulas containing the panel's sizes.

Calling L, H, and T the dimensions respectively along axes X, Y, Z the values relative to each axis can contain the correspondent dimensional parameter :

X and I = function of L (panel's length)
Y and J = function of H (panel's height)
Z and K = function of T (panel's thickness).

The expressions allowed are linear combinations of the kind of, for instance for X :

$$X(\pm a * L / b \pm c * L / d)$$

where :

a , b , c , d whole or decimal numbers, of max. shape [4.2] ;
* multiplication sign
/ division sign
+ sum sign
- subtraction sign
() round brackets should enclose the expression. The character "(" should be typed immediately after the value function (in this case X) without adding spaces.

No space should be digitated within the round brackets. In case of expressions entered wrong, it will be displayed the error :
" TEXT ENTERED WRONG"

This programming method allows to bond the points on the edge of the panel, at its center or to run operations along fixed fractions of any of the panel's dimensions.

No control is performed concerning the "correctness" of a program in parametric pattern. So, for instance, on a program of a arc through the radius, a program concerning the field I value (here related to the radius) in parametric pattern is anyway accepted, even if it does not represent a legitimate use of it.

Valid expressions of parametrical programming are here listed as examples :

X(50)
Y(3*H/4)
Z(S-5)
X(L/2+50) .

In this example it is clear that there is no need of the dimension parameter related to the programmed value (for instance, letter L on value of X). Also expression entirely numeric are valid :

X(700+100)
X(1000-100)
X(900/5) .

At this purpose, an interesting application concerns a program having [inches] as measure unit, with expressions containing fractions of [inches].

For instance :

X(5+7/8) could indicate 5 inches + (7/8) of an inch
x(5/4) could indicate (5/4) of an inch.

Some use of parametric pattern programming is proposed in the pages to follow.

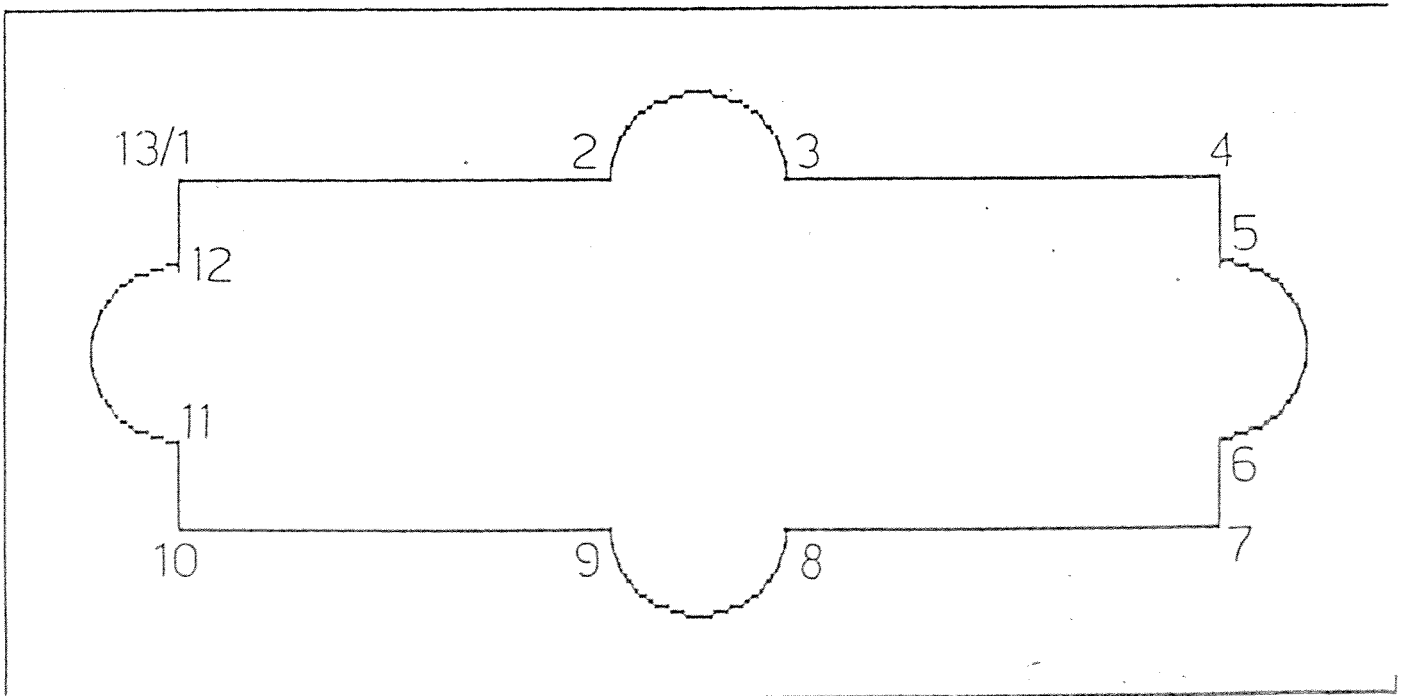
EXAMPLE n. 1

(Picture 11.1)

1. G10 X100 Y100 Z10
2. X(L/2-50)
3. G12 X(L/2+50) I(L/2) J100
4. G10 X(L-100)
5. Y(H/2-50)
6. G12 Y(H/2+50) I(L-100) J(H/2)
7. G10 Y(H-100)
8. X(L/2+50)
9. G12 X(L/2-50) I(L/2) J(H-100)
10. G10 X100
11. Y(H/2+50)
12. G12 Y(H/2-50) I100 J(H/2)
13. G00 Y100

The example shows the interdependence of the work values X and Y at the panel's edges and center.

Picture 11.1



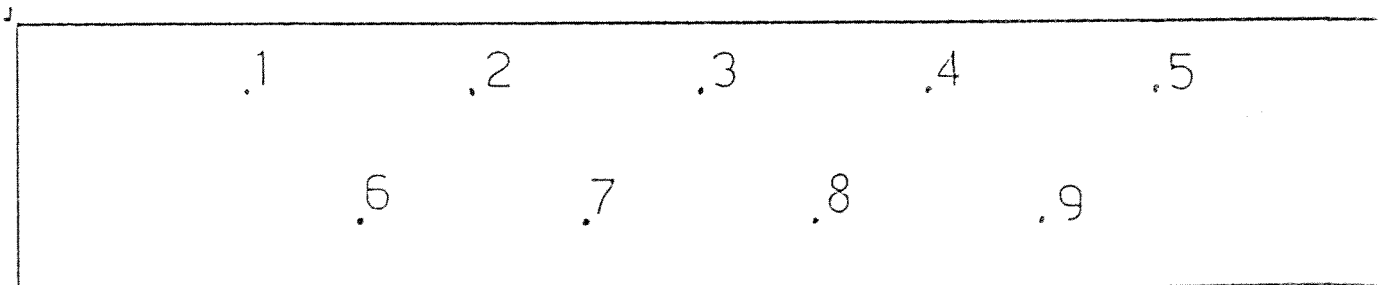
EXAMPLE n. 2

(Picture 11.2)

1. G00 X(L/6) Y(H/4) Z5 T1
2. X(2*L/6)
3. X(3*L/6)
4. X(4*L/6)
5. X(5*L/6)
6. X(3*L/12) Y(H-H/4) Z(S-5)
7. X(5*L/12)
8. X(7*L/12)
9. X(9*L/12)

In the example are performed operations (single borings) along fixed fractions of panel's length and height ; the depth of the two lines of holes varies according with panel's thickness.

Picture 11.2



PROGRAMMING SUBPROGRAMS

Functions G05 and S on the program line in object.

Syntax : G05 Xxxx.x Yyyy.y Saaaa [Dddd] [Rrr.r] [Oo] [Ttt]

where:

- G05 it is the function to load a subprogram ;
- aaaaa it's the name of the program loaded as subprogram, identified with a max. of 8 characters. Field "aaaaa" is specified on command "S". In every respect the program specified on function "S" is a normal processing program.
- Xxxx.x it is the value X at which the program locates the first point specified in the subprogram. If not specified, the value X is taken from the previous program line.
If specified, the value X is spreaded itself on the next program line.
- Yyyy.y it is the value Y in which the program locates the first point specified in the subprogram.
It follows the same rules as value X.
- Dddd It is an optional field to assign the subprogram rotation code. The code is given on the parameter "ddd" :
- | | |
|-----|--|
| 0 | no rotation |
| 90 | 90 degrees clockwise rotation |
| 180 | 180 degrees clockwise rotation |
| 270 | 270 degrees clockwise rotation |
| X | mirrors the subprogram along axis X
(rotation around Y) |
| Y | mirrors the subprogram along axis Y
(rotation around X) . |
- If omitted it is assumed code D0.
- Rrr.r It is an optional field to assign the value of mill radius correction, to apply to every contour programmed in the subprogram. If field R is existing, in the subprogram should not be specified any mill radius : other way, it will be realized an error for mill radii multiple assignment. With a radius equal to "+/-", the mill radius value is taken from Parameters and used with the sign specified. No spreading on the parameter R is carried out when it is appointed to load the subprogram. If field R is omitted, are assumed as

valid the mill radii eventually contained in the subprogram. The syntax of the parameter "rr.r" has been defined on the paragraph "mill radius correction".

Oo It is an optional field to assign the Origin (starting point) to which refers the point of the co-ordinates Xxxxx.x Yyyyy.y as above. The Origins specified within the subprogram are, instead, normally applied before the "placement" of the subprogram itself. If specified, the Origin is spreaded to the next program line. If field O is omitted, it is assumed the Origin existing in the program. The syntax of the parameter "O" it is defined in the paragraph "Origins programming". It is possible to enter only one among the default Origins, from O0 to O3.

Ttt It specifies the working tool in every milling operation existing in the subprogram. If this parameter is not specified, the subprogram is loaded with the tools specified when the subprogram itself was stored.

The values of Z of the subprogram are not altered.

If the program loaded with function S begins with a milling cycle, with mill radius correction applied, the first working point relocation on values Xxxx.x Yyyy.y above defined it is applied to the contour already corrected.

Instead, if the correction is entered on the line loading the subprogram (G05 ...)., or if the program does not begin with a milling corrected on Rrr.r, on the values X and Y specified is located the first point as programmed in the subprogram.

A program can load only the subprograms realized with the same measure unit ([mm] o [inch]), other way it will be displayed a error for "DIFFERENT MEASURE UNIT".

The function to load subprogramd is equivalent to an "automatic merge" function to define parameters (Origins, Rotations, etc) of remarkable practical usefulness. Upon command "STORE" (Edit mode), the subprogram is read and inserted, opportunely elaborated, in the performable text of the program to be stored.

If the subprogram has been changed in a second moment no automatic updating is performed in the programs using it. It is therefore necessary to examine in Edit the programs involved and perform a new storing.

The example following outlines an idea of simple applications of a subprogram loaded.

name : PIPPO

```
G10 X50 Y50 ;
X100 ; The program PIPPO is represented
G12 X150 I125 J50 ; in picture 12.1
G10 Y100 ; PIPPO is loaded on the programs
X50 ; following.
G00 Y50 ;
```

name : ES1

```
G05 X50 Y50 SPIPPO ;
X300 SPIPPO ; The program ES1 is
X200 Y300 SPIPPO ; depicted in picture 12.2
X150 Y150 SPIPPO O3 ;
```

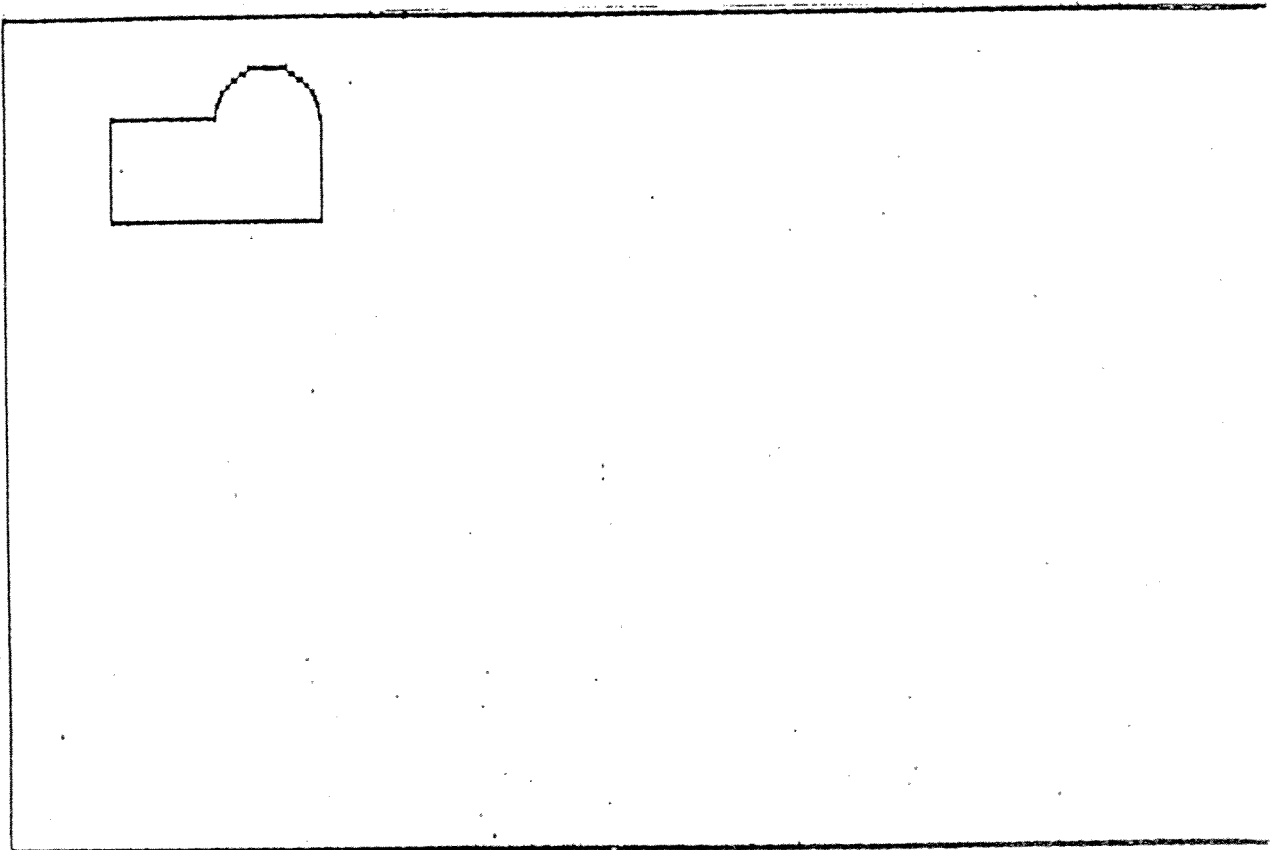
name : ES2

```
G05 X50 Y50 SPIPPO D0 ;
X350 SPIPPO D90 ; The program ES2 is
X350 Y350 SPIPPO D180 ; depicted in picture 12.3.
X100 SPIPPO D270 ; In the first line, D0 could also
; not be specified.
```

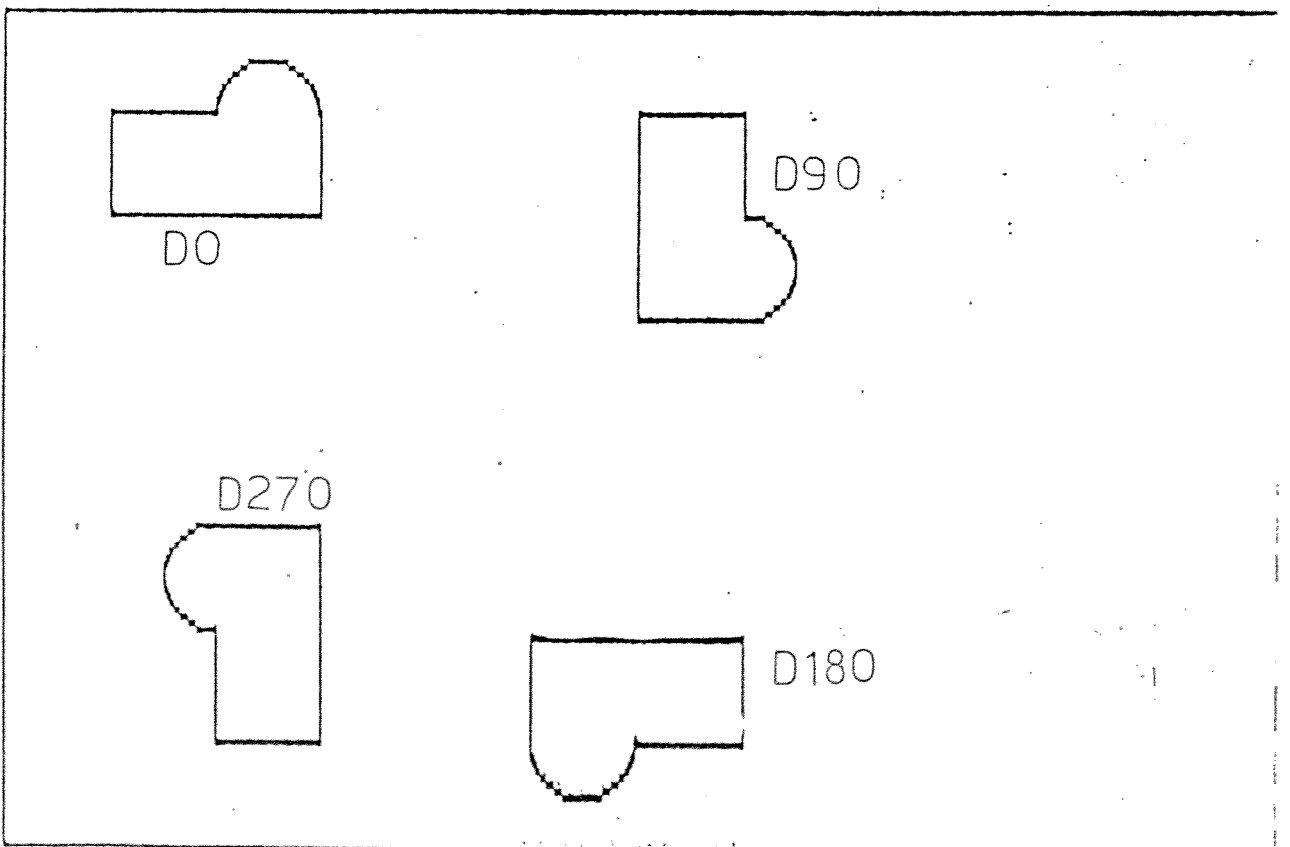
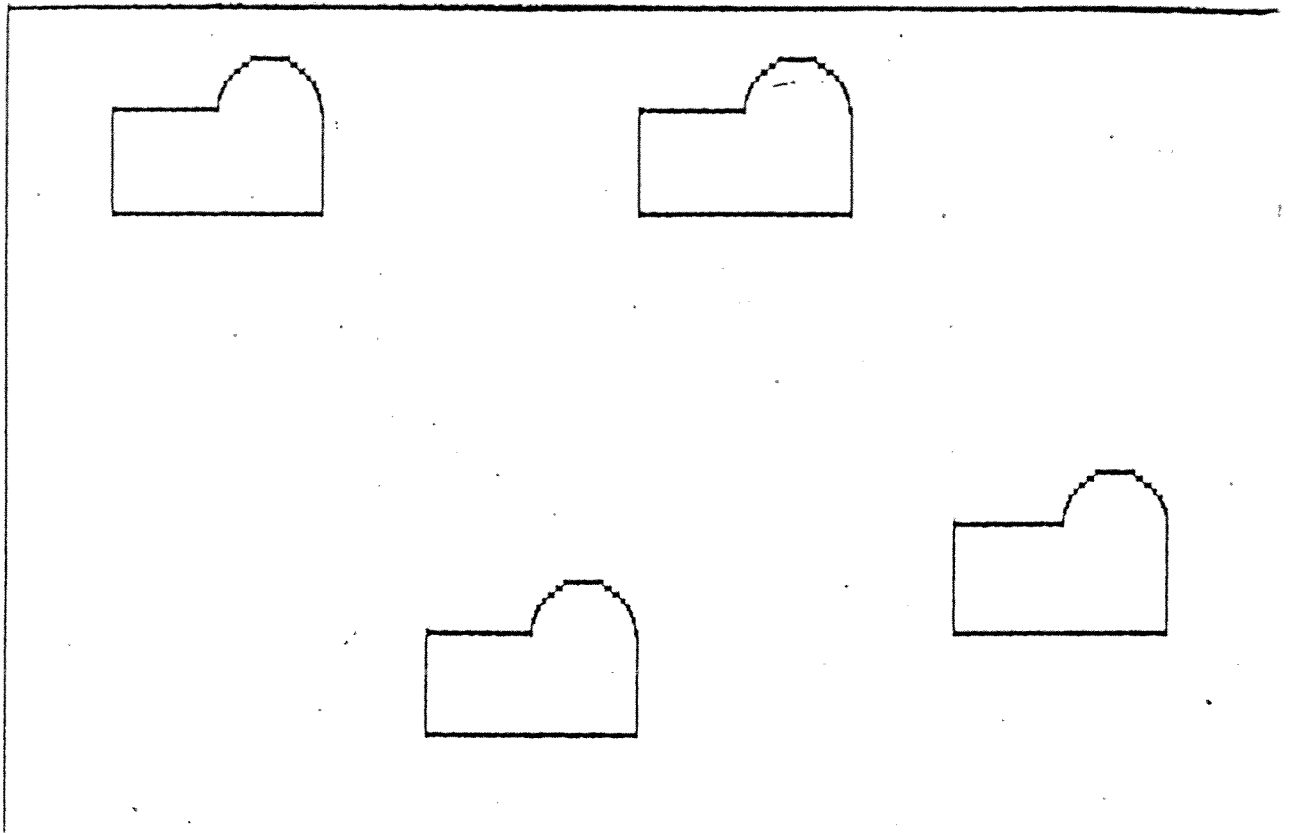
name : ES3

```
G05 X50 Y50 SPIPPO D0 ;
X350 SPIPPO DX ; The program ES3 is,
X50 Y300 SPIPPO ; depicted in picture 12.4.
Y200 SPIPPO DY ;
;
```

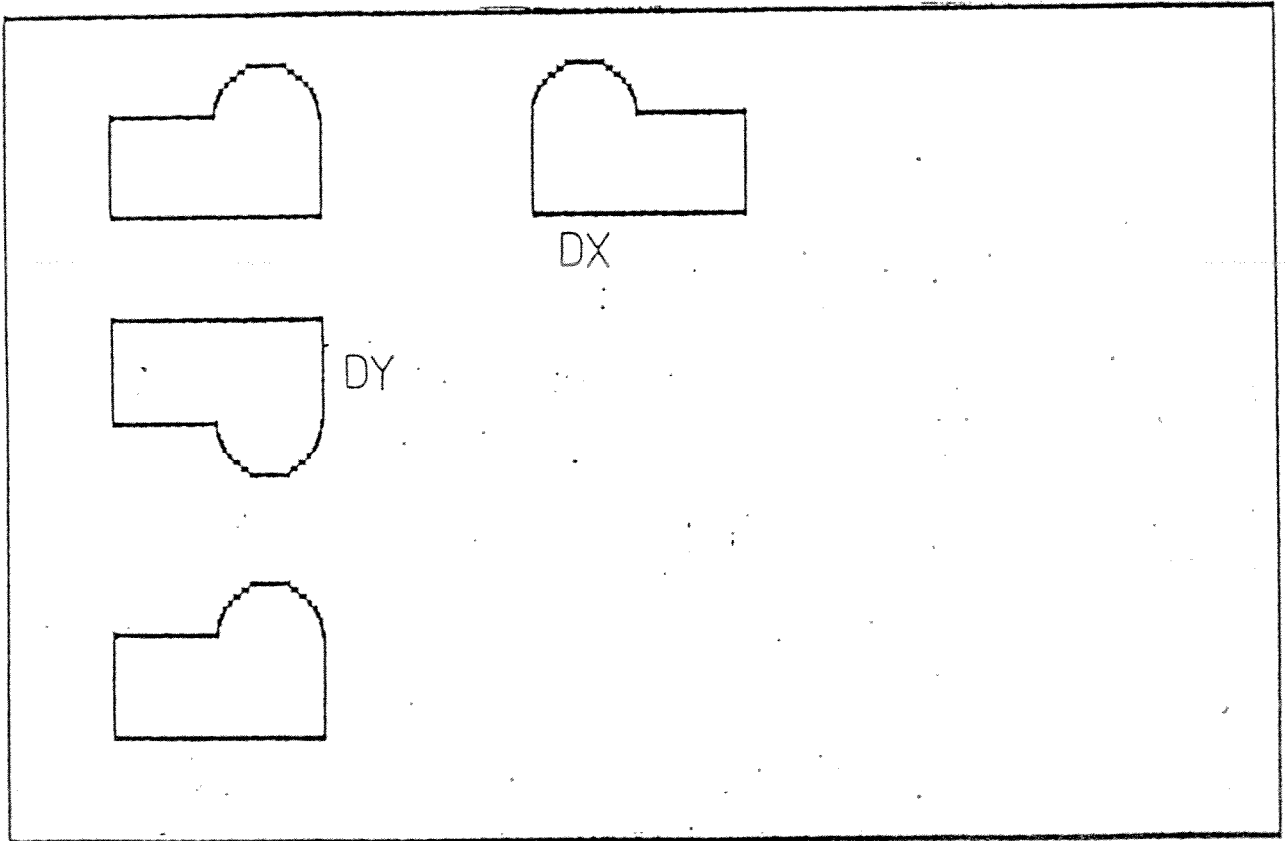
Picture 12.1



Pictures 12.2, 12.3



Picture 12.4



As evident in the examples previously reported, the processings specified in the subprogram are positioned on the programmed panel through the functions X , Y , D , O .

In the examples shown up to now, all processings keep the reciprocal positions specified in the subprogram's original text . That is because just simple operations of translation and rotation have been performed, up to now, on the whole text.

A particular attention should instead be paid when performing operations such as :

horizontal borings
borings on fitting heads
blades in X
blades in Y.

Besides, also the following should be of particular interest :

function P of interpolation plane definition
functions G of circular interpolation

NOTE

It is however opportune to proceed to a few remarks :

When creating a program with the idea of using it as a subprogram, it is advisable to adopt the following criteria :

1. Create the program on Origin "Ø"
2. Do not mix operations such as horizontal borings, or blades use or fitting with operations of millings and vertical boring.
3. Run possibly only vertical borings and millings, preferably excluding others operation, also fitting.
4. Perform mill radius corrections only when loading the subprogram.

function D	function	result
D90	G06	change it into G08 tool : changes into R
	G07	change it into G09 tool : changes into S
	G08	change it into G07 tool : changes into Q
	G09	change it into G06 tool : changes into P
	G16 (**) G17 G20	no rotation is applied : values XY follow the subprogram's first point traslation.
	PXZ	becomes PYZ
	PYZ	becomes PXZ ; G02 / G12 become G03 / G13 ; G03 / G13 become G02 / G12 ;

(**)

The operations of grooving and fitting are treated in a special way : the program's first point traslation keeps respected, but the direction (X or Y) and the advancement direction do not change. For instance, a fitting programmed on increasing values X is however mantained as such.

function D	function	result
D180	G06	change it into G07 tool : changes into Q
	G07	change it into G06 tool : changes into P
	G08	change it into G09 tool : changes into S
	G09	change it into G08 tool : changes into R
	G16 G17 G20	no rotation is applied ; the values XY follow the subprogram first point traslation.
	function P	keeps the same
	G02 /G12	change only on FXZ or PYZ : G02 becomes G03 ; G12 becomes G13 ;
	G03 /G13	change only on FXZ or PYZ : G03 becomes G02 ; G13 becomes G12 ;
D270	G06	change it into G09 tool : changes into S
	G07	change it into G08 tool : changes into R
	G08	change it into G06 tool : changes into P
	G09	change it into G07 tool : changes into Q
	G16 G17 G20	no rotation is applied ; the values XY follow the subprogram's first point traslation.
	PXZ	becomes PYZ ; G02 / G12 become G03 / G13 ; G03 / G13 become G02 / G12 ;
	PYZ	become PXZ

function D	function	result
DX	G06	change it into G07 tool : changes into Q
	G07	change it into G06 tool : changes into P
	G08 G09	unchanged
	G16 G17 G20	no rotation is applied ; the values XY follow the subprogram 's first point traslation.
	G02 /G12	change only on PXY, FXZ or PXYZ : G02 becomes G03 ; G12 becomes G13 ;
	G03 /G13	change only on PXY, FXZ or PXYZ : G03 becomes G02 ; G13 becomes G12 ;
DY	G06 G07	unchanged
	G08	change it into G09 tool : changes into S
	G09	change it into G08 tool : changes into R
	G16 G17 G20	no rotation is applied ; the values XY follow the subprogram's first point traslation.
	G02 /G12	change only on PXY, FYZ or PXYZ : G02 becomes G03 ; G12 becomes G13 ;
	G03 /G13	change only on PXY, FYZ or PXYZ : G03 becomes G02 ; G13 becomes G12 ;

Upon rotations D180, DX, DY, the operations of vertical boring do not change typology (G06/G07 do not change into G08/G09, and vice versa). In these cases, upon function G change (ex. G06 changed into G07) every working tool specified in the original function (ex. G06) it is changed into the tool related to it in mirrored operations (see Head configuration modes, in Parameters). If the mirrored tool is not configured in an acceptable way, it will be displayed the error "WRONG ROTATION WITH MORE HORIZONTALS ON THE SAME LINE". On the rotations D180, DX and DY, it is possible to specify more borers contemporaneously.

In the cases of rotation on D90 and D270, instead, operations in G06/G07 are changed into operations in G08/G09, and vice versa. In these cases it is not possible to enter, in the subprogram, horizontal boring functions with more tool working at the same time, other way it will be displayed again the error:

"WRONG ROTATION WITH MORE HORIZONTALS ON THE SAME LINE".

When a function G is changed, like G06 changed into G08, the working tool is appointed automatically (tool of the kind required with higher i.d. number). If no tool is found of the required kind, it will be displayed the error "WRONG TOOL".

The following example set in evidence some of the rules just given

name : TSC

```
G00 X100 Y100 T2
Y125
Y150
Y175
Y200
G06 Y200 X10 Z10 T18
Y150
Y100
Y50
G08 X50 Y10 Z10
X75
```

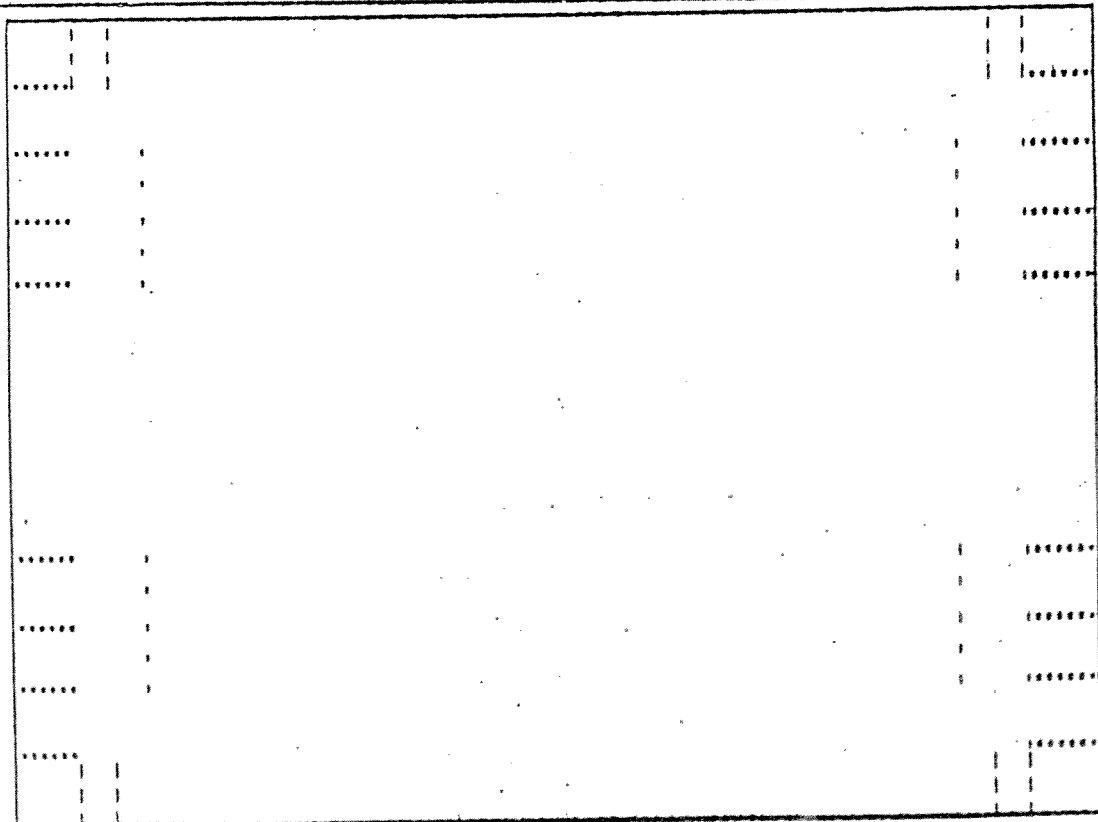
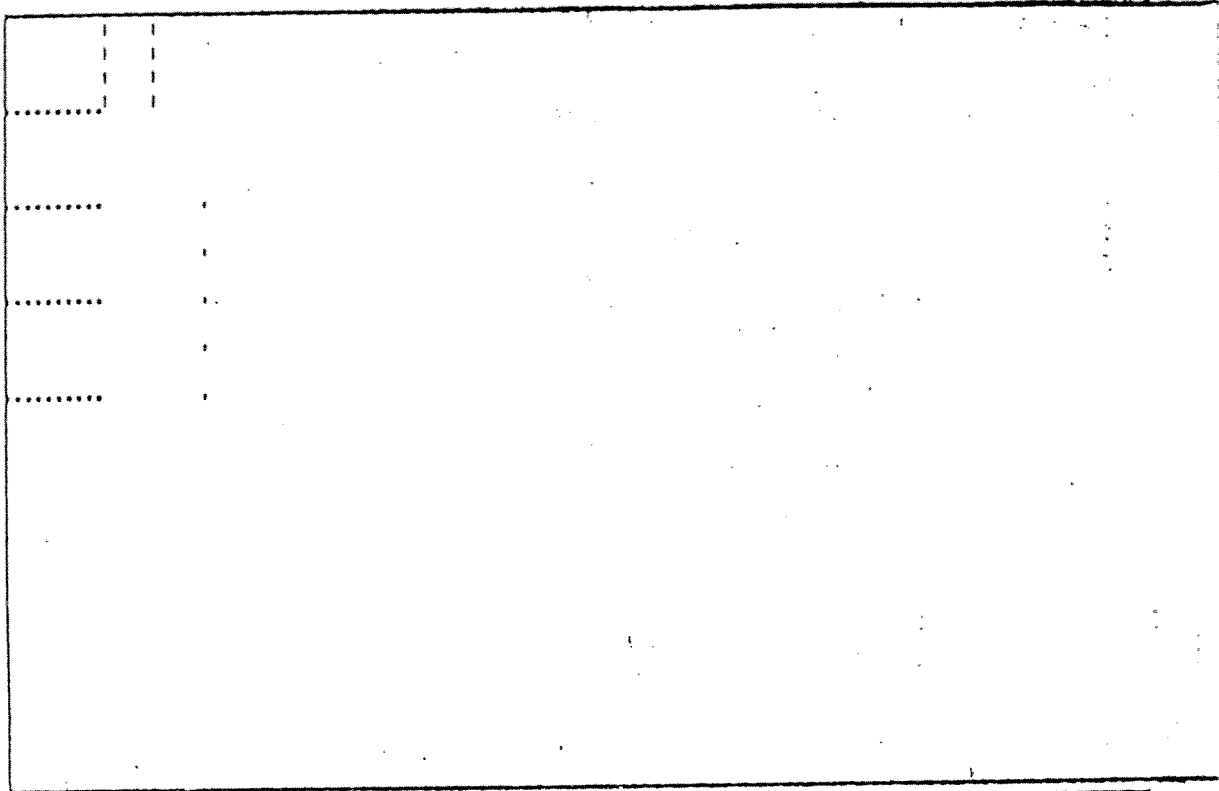
The program TSC is depicted in picture 12.5.

name : CALLTSC

```
G05 X100 Y100 STSC
G05 X100 Y100 STSC DX 02
G05 X100 Y100 STSC D180 03
G05 X100 Y100 STSC DY 01
```

The program CALLTSC is depicted in picture 12.6.

Pictures 12.5 , 12.6



The following examples show how to use the functions Origin when loading a subprogram. The subprogram in object, besides, contains interpolations on different planes and connections.

name : SUB1

G10 X-2 Y200 Z15 PXZ
G12 Z5 A10
G00 X295
G10 X300 Y195 PZY
G13 Y-2 A10
G00 Z15

The program SUB1 is depicted in picture 12.7.

name : UNO

G05 X-2 Y200 SSUB1 00
G05 DX SSUB1 02
G05 D180 SSUB1 03
G05 DY SSUB1 01

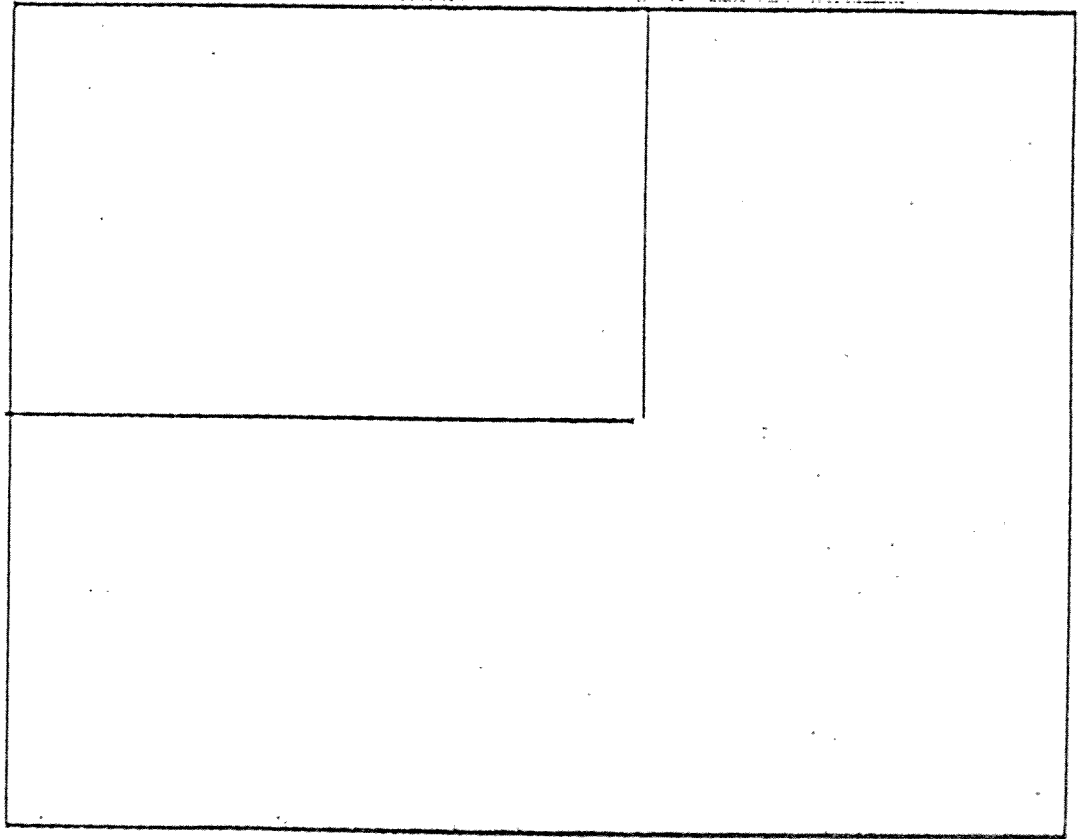
The program UNO is depicted in picture 12.8.

name : DUE

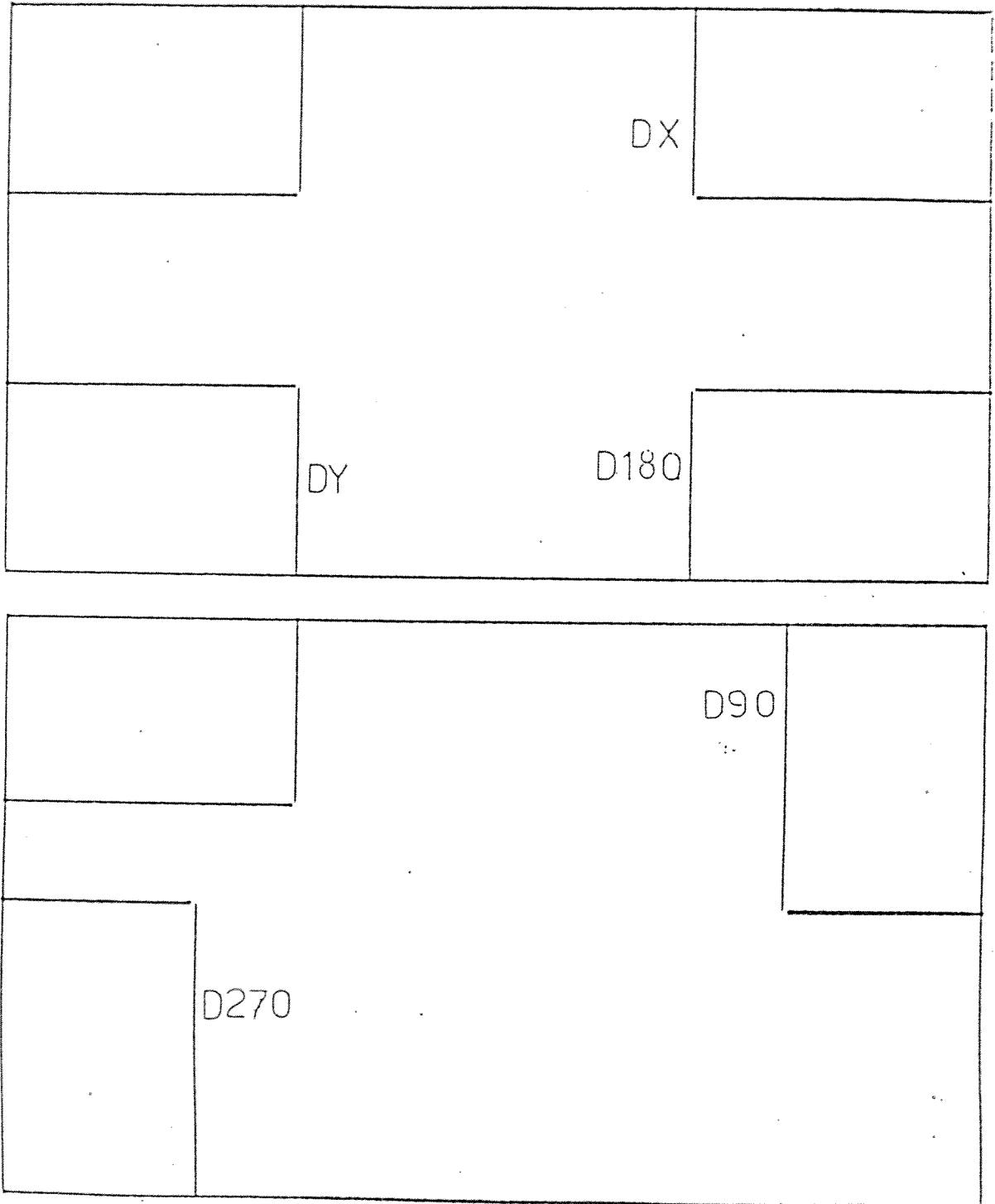
G05 X-2 Y200 SSUB1
G05 X200 Y-2 SSUB1 D90 02
G05 D270 SSUB1 01

The program DUE is depicted in picture 12.9.

Picture 12.7



Pictures 12.8 , 12.9



----- programming subprograms -----

In the programs following is proposed again the program of picture 12.1 with the addition of a fitting function.

name : PIPPO1

```
G10 X50 Y50 ;  
X100 ; The program PIPPO1 is  
G12 X150 I125 J50 ; depicted in picture 12.10.  
G10 Y100 ; PIPPO1 is loaded in the  
X50 ; program following.  
G00 Y50 ;  
G20 X50 Y150  
G00 X320
```

name : ES01

```
1. G05 X50 Y50 SPIPPO1 ;  
2. G05 O2 DX SPIPPO1 ; The program ES01 is  
3. G05 O3 D180 SPIPPO1 ; depicted in picture 12.11  
4. G05 O1 X100 Y200 D270 SPIPPO1 ;  
-----
```

In ES01, the performing of the fitting operations do not change the direction imposed in the original text PIPPO1 : in other words, with X increasing. In the example ES01 : The values Start/End fitting result exchanged on the loading lines in DX and in D180. The relocation in X and Y follows, so to respect the original reciprocal position with the first point programmed in the subprogram. As usual, upon reckoning of the ref.points to run the G20, the execution on the points of start fitting is however carried out as programmed, with eventual approximation (for the values X and number of holes, as seen) on the end point : therefore, on each of the four fittings, the approximation carried on is always on the right side point.

For completeness it is here reported a program text equivalent, under the point of execution, to the program ES01 previously proposed.

```

G10 X50 Y50
X100
G12 X150 I125 J50
G10 Y100
X50
G00 Y50
G20 X50 Y150
G00 X320
G10 X950 Y50
X900
G13 X850 I875 J50
G10 Y100
X950
G00 Y50
G20 X680 Y150
G00 X950
G10 X950 Y450
X900
G12 X850 I875 J450
G10 Y400
X950
G00 Y450
G20 X680 Y350
G00 X950
G10 X100 Y300
Y250
G12 Y200 I100 J225
G10 X150 Y200
Y300
G00 X100
G20 X100 Y400
G00 X370
    
```

This block is equivalent to a program line :

1. G05 X50 Y50 SPIPP01

equivalent to program line :

2. G05 02 DX SPIPP01

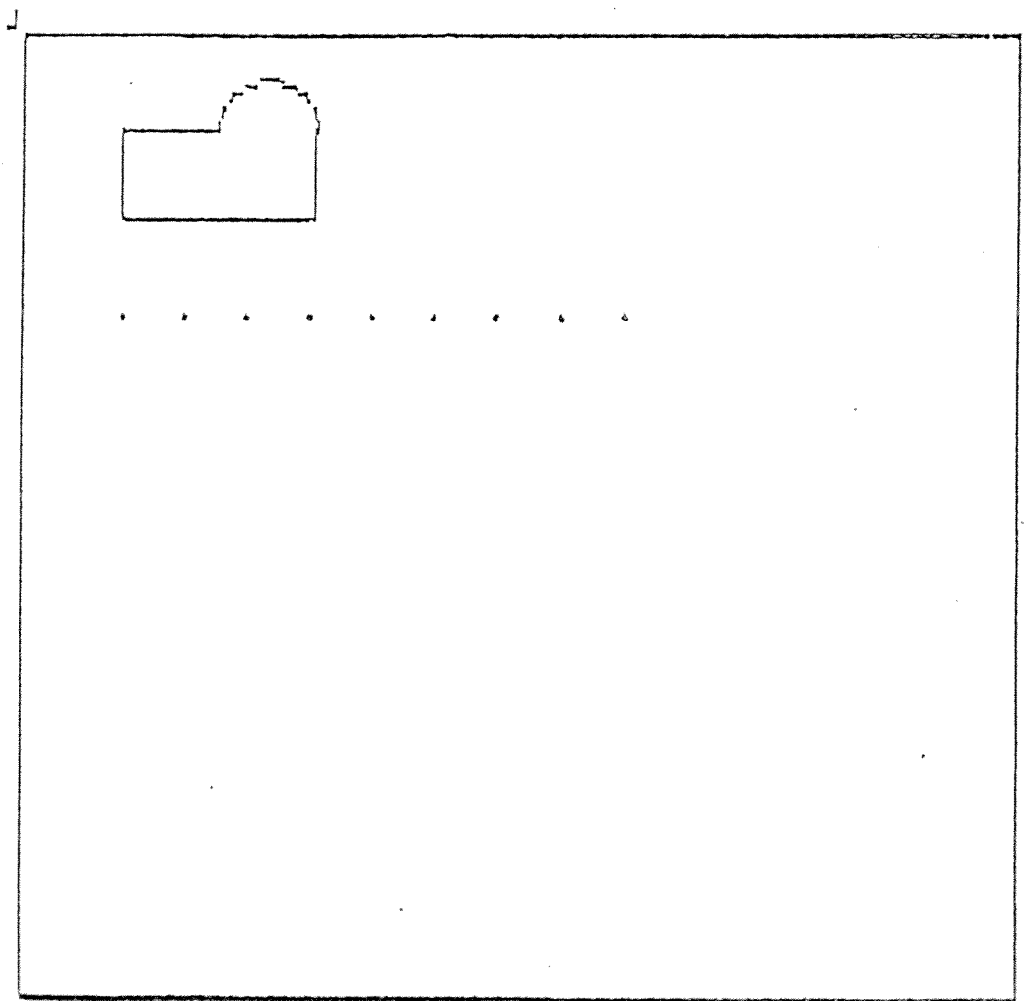
equivalent to program line :

G05 03 D180 SPIPP01

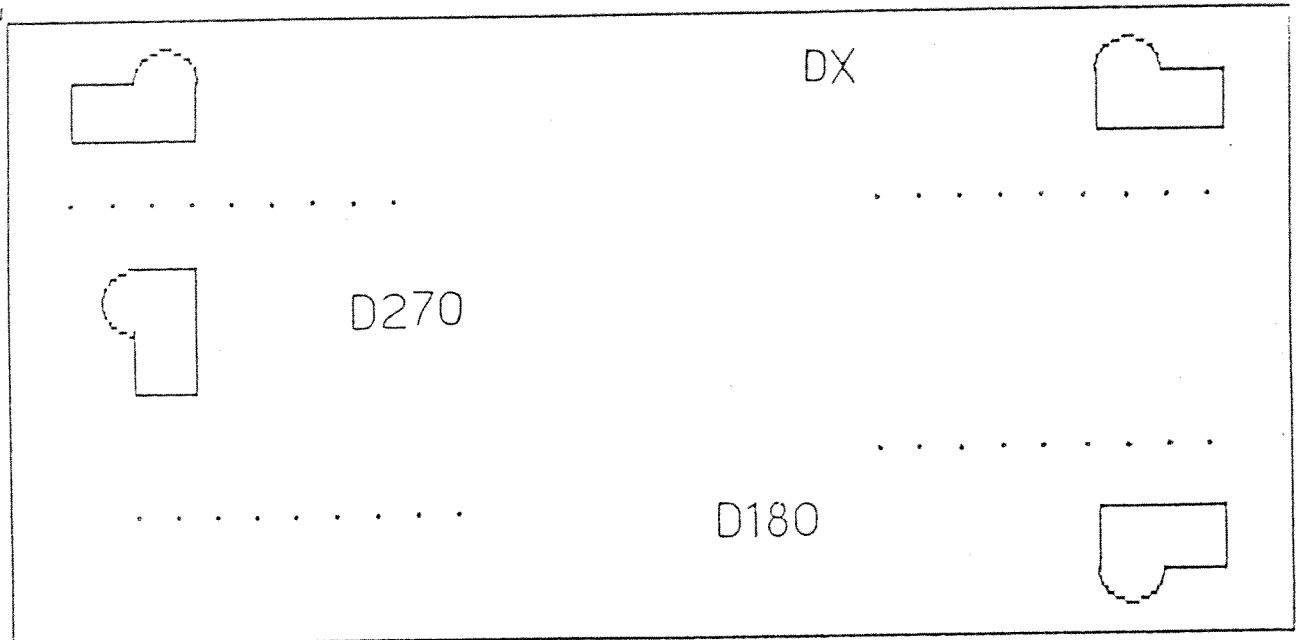
equivalent to program line :

4. G05 01 X100 Y200 D270 SPIPP01

Picture 12.10



Picture 12.11



ERRORS DISPLAYED WHEN EDITING

Error number	Message and description
23	<p>SUBPROGRAM NAME NOT VALID If the name related to function "S" has more than 8 digits (characters) The space after the name has not been entered The name is not specified.</p>
24	<p>SUBPROGRAM NOT FOUND If the subprogram has not been found in the programs directory.</p>
22	<p>ROTATION CODE NOT VALID If function "D" of rotation does not result defined in a correct way.</p>
39	<p>ERROR IN PROGRAM'S ORIGIN If on function G05 is specified an origin with a number above 3 (example : 06)</p>
42	<p>TYPE OF MILLING NOT POSSIBLE If function "S" has been assigned on a program line containing a function "G" different from "G05".</p>
47	<p>WRONG ROTATION WITH MORE HORIZONTALS IN THE SAME LINE . Upon inquiry of rotations D90 or D270, with horizontal multiple borings (with more tools). . Upon inquiry of rotations D180 or DX o DY, with mirrored tools in horizontal borings not entered when the head has been configured.</p>
44	<p>WRONG TOOL Upon inquiry of rotation D90 o D270, if no tool necessary to perform horizontal borings has been found.</p>
52	<p>MULTIPLE DEFINITION OF MILL RADIUS If function R is defined on the line loading the subprogram, and the subprogram contains the same function already programmed.</p>

Error number	Message and description
55	WRONG TOOL ON G05 Upon working tool specification in millings : . if more tools are specified. . if the tool specified is not defined as type "J".
67	DIFFERENT MEASURE UNITS If the program loaded is defined on a linear measure unit different from the one running.