



Albatros

3.2.3

Controllo Numerico



Tecnologie e Prodotti per l'Automazione

La presente documentazione è di proprietà della TPA S.r.l.
Ne è vietata la duplicazione non autorizzata.
La società si riserva il diritto di modificarne il contenuto in
qualsiasi momento.

Indice

1	Introduzione	1
1.1	Come utilizzare la guida	1
1.2	Le finestre di lavoro	1
2	Composizione del sistema	2
2.1	Diritti di accesso al sistema	2
2.2	Supporto multilingue	2
2.3	Architettura tipica di un sistema	3
2.4	Organizzazione e configurazione logica	4
2.5	I Dispositivi	5
3	Quadro sinottico	7
3.1	Uso del Quadro Sinottico	7
3.2	Come operare sul Quadro Sinottico	7
3.3	Come agire sui Dispositivi	7
3.4	Movimento Assi in Manuale	7
4	Parametri Tecnologici e Utensili	9
4.1	La finestra Parametri Tecnologici	9
4.2	La finestra Parametri Utensili	10
5	Diagnostica	12
5.1	La finestra Diagnostica	12
5.2	Composizione della finestra Diagnostica	12
5.3	Rappresentazione dei Dispositivi	12
5.4	Interagire con i Dispositivi	13
5.5	Elenco dei tasti per navigare all'interno di una struttura ad albero	13
5.6	Correttori di linearità	14
5.7	Plancia di taratura assi	14
6	Errori e Segnalazioni	17
6.1	Introduzione	17
6.2	Errori di sistema	18
6.2.1	Errori generati dalla gestione assi	18
	1 NomeAsse: collegamento encoder errato	18
	2 NomeAsse: movimento non finito	18
	3 NomeAsse: servoerror	18
	4 NomeAsse: oltre limite positivo	18
	5 NomeAsse: oltre limite negativo	19

	10 NomeAsse: l'esecuzione Real-Time è più veloce della costruzione del profilo	19
6.2.2	Errori generati dalla gestione I/O remoti	19
	2049 Ricevitore numero: configurazione errata	19
	2050 Ricevitore numero: scollegato	19
	2051 Ricevitore numero: ricollegato	19
	2052 Ricevitore numero: errore di riletture uscita non collegata numero NumeroUscita	19
	2054 Ricevitore numero: tipo errato	19
	2055 Ricevitore numero: inizializzato	20
	2056 Ricevitore numero: errore alimentazione +24 Vcc	20
	2057 Errore alimentazione GreenBUS	20
	2058 Ricevitore numero: errore in riletture TipoDispositivo NomeDispositivo	20
	2059 Fallito il test della dual port memory del trasmettitore	20
	2060 Errore nell'inizializzazione del trasmettitore	20
	2061 Errore durante la trasmissione del firmware al trasmettitore	21
	2062 Errore durante la trasmissione della configurazione al trasmettitore	21
	2063 Errore durante la trasmissione della configurazione al ricevitore	21
	2064 Ricevitore numero: versione firmware non compatibile	21
	2065 Ricevitore numero: errore in una comunicazione asincrona	21
	2066 Ricevitore numero: errore generico	21
	2067 Ricevitore numero: errore durante la trasmissione della configurazione	21
	2068 Ricevitore numero: errore interno n. numeroerrore	22
	2069 Ricevitore numero: errore alimentazione +24 Vcc banco numero	22
6.2.3	Errori generati dalla gestione MECHATROLINK-II	22
	2308 Scheda NumeroScheda: inizializzazione fallita dovuta a un'impostazione non corretta di un parametro di configurazione	22
	2341 Scheda NumeroScheda: il numero di servoazionamenti supera il massimo consentito	22
	2342 Scheda NumeroScheda: l'indirizzo hardware del servoazionamento Servo supera il valore massimo consentito	22
	2349 Scheda NumeroScheda: il servoazionamento Servo non è collegato	23
6.2.4	Errori generati dalla gestione CanBUS	23
	2761 Nodo numero: scollegato	23
	2762 Nodo numero: ricollegato	23
	2763 Errore di mancata trasmissione	23
	2764 Nodo numero: errore di mancata ricezione	23
	2765 Nodo numero: inizializzato	24
	2766 Condizione di guasto sull'interfaccia CAN	24
	2767 Errore perdita di stato CANopen	24
	2768 Nodo numero: errore mancata ricezione PDO	24
	2769 Nodo numero: errore ricezione di un nodo non configurato	24
	2770 Nodo numero: configurazione errata	24
	2771 Nodo numero: errore sulla comunicazione SDO	24
	2772 Timeout sul ciclo CAN di interrogazione dei nodi	24
	3073 Nodo numero: errore Emergency n. NumeroErrore	25
	3074 Nodo numero: errore CAN generico n. NumeroErrore	25
	3088 Scheda CAN numero: nodo NumeroNodo: errore sulla comunicazione SDO n. NumeroErrore - descrizione	25
6.2.5	Errori generati dalla gestione bus EtherCAT	25
	3329 Errore nell'inizializzazione del socket di comunicazione	25
	3330 Errore durante la scansione della rete EtherCAT	25
	3331 Errore nella configurazione della mailbox di trasmissione	25
	3332 Errore nella configurazione della mailbox di ricezione	26
	3333 Scheda EtherCAT numero: errore nel tipo delle espansioni del nodo NumeroNodo	26
	3334 Errore in fase di configurazione dei PDO	26
	3335 Nodo NumeroNodo in allarme (NumeroErrore)	26
	3336 Scheda EtherCAT numero: il numero di espansioni del nodo NumeroNodo è errato	27
	3337 Scheda EtherCAT: nodo NumeroNodo scollegato	27

	3338 Scheda EtherCAT: nodo NumeroNodo ricollegato	28
	3340 Scheda EtherCAT: il nodo NumeroNodo non ha risposto alla richiesta (Codice)	28
	3341 Scheda EtherCAT: il nodo NumeroNodo non esiste	28
	3342 Cavo scollegato	28
	3343 Scheda EtherCAT numero: nodo NumeroNodo non passa nello stato SAFE-OPERATIONAL (Codice)	28
	3344 Scheda EtherCAT numero: nodo NumeroNodo non passa nello stato OPERATIONAL (Codice)	28
	3345 Scheda EtherCAT: comunicazione instabile	29
	4400 Troppi assi attivi in FASTREAD (Funzione:NomeFunzione linea:NumeroLinea)	29
6.2.6	Errori generati dall'inizializzazione	29
	769 Configurazione software errata	29
	770 Numero di IRQ configurato errato	29
	772 Errore nella lettura dell'area di memoria tamponata durante l'inizializzazione	29
	773 Raggiunto il numero massimo di assi configurabili	29
	774 Il Real-time assi non è entrato in esecuzione	30
	775 Tempo insufficiente per l'esecuzione GPL	30
	776 Tempo eccessivo di esecuzione del Real-Time	30
	777 Watchdog scaduto	30
	778 Il codice di Main del firmware è bloccato	30
	1025 Scheda NumeroScheda: non risponde al comando	30
	1026 Scheda NumeroScheda: errore nell'invio del firmware alla scheda assi	30
	1028 Scheda NumeroScheda: firmware non presente	31
	1029 Scheda NumeroScheda: Main bloccato	31
	1031 Scheda NumeroScheda: errore in inizializzazione	31
	1032 Scheda NumeroScheda: fallito il test della dual port memory	31
	1033 Scheda NumeroScheda: il codice di boot del firmware non è in esecuzione	31
	1035 Scheda NumeroScheda: non presente	31
	1037 Scheda NumeroScheda: fallita l'apertura della dual port memory	32
	1039 Scheda NumeroScheda: watchdog scaduto	32
	1040 Scheda NumeroScheda: errore alimentazione +24 Vcc	32
	1047 Scheda NumeroScheda: configurazione software non permessa	32
	1052 Scheda NumeroScheda: è in esecuzione il codice di boot	32
	1053 Scheda NumeroScheda: watchdog assi scaduto	32
	1055 Watchdog scaduto sulla scheda NumeroScheda	32
	1056 Scheda NumeroScheda: errore di alimentazione interfaccia CAN	32
	1057 Scheda NumeroScheda: errore interno numero NumeroErrore	33
6.2.7	Errori generati dalla gestione della memoria	33
	1281 Errore nell'allocazione di memoria nell'area di heap	33
	1286 Errore nella gestione dello heap	33
	1287 Troppe disallocazioni di memoria dallo heap	33
	1289 Errore nella creazione delle variabili globali	33
	1290 Errore nella dimensione delle variabili non volatili	33
	1291 Errore nella dimensione delle variabili sola lettura	33
6.2.8	Errori generati dai fault	34
	1559 Trace di breakpoint	34
	1569 Codice operativo del microprocessore non valido	34
	1586 Valore INTEGER diviso per zero	34
	1600 Overflow nel risultato di un'operazione in virgola mobile	34
	1601 Underflow nel risultato di un'operazione in virgola mobile	34
	1602 Argomento non valido per un'operazione in virgola mobile	34
	1603 Valore in virgola mobile diviso per zero	35
	1604 Risultato inesatto di un'operazione in virgola mobile	35
	1605 È stato usato un valore in virgola mobile errato	35
	1728 Si è cercato di accedere a un indirizzo non valido	35
	1735 Eccezione generica	35

1736	Dati non allineati	35
1801	Allarme temperatura	35
1802	Allarme ventola	36
1803	La frequenza della CPU è instabile	36
6.2.9	Errori generati da istruzioni GPL	36
4097	Il dispositivo TipoDispositivo NomeDispositivo non è configurato	36
4098	La variabile globale NomeVariabile è inesistente	36
4099	Funzione NomeFunzione non trovata	36
4101	Gestione incongruente dell'asse NomeAsse	36
4105	Istruzione non eseguibile sull'asse NomeAsse	37
4106	Il remoto riferito all'asse passo-passo NomeAsse non è collegato	37
4107	Istruzione SYSOK con argomenti errati	37
4108	L'asse NomeAsse: quota finale oltre i limiti software	37
4110	Velocità errata	37
4111	Accelerazione asse NomeAsse negativa	37
4112	Decelerazione asse NomeAsse negativa	37
4114	Asse NomeAsse: azzeramento su ingresso veloce non effettuato	38
4115	Asse NomeAsse: tacca di zero non trovata	38
4353	Codice operativo istruzione sconosciuta (Funzione:NomeFunzione linea:NumeroLinea)	38
4354	Operazione matematica errata (Funzione:NomeFunzione linea:NumeroLinea)	38
4355	Indirizzo errato di matrice o vettore (Funzione:NomeFunzione linea:NumeroLinea)	38
4356	Istruzione RET non chiamata da CALL (Funzione:NomeFunzione linea:NumeroLinea)	39
4357	Variabile locale inesistente (Funzione:NomeFunzione linea:NumeroLinea)	39
4358	Etichetta di salto inesistente (Funzione:NomeFunzione linea:NumeroLinea)	39
4359	Argomento macro errato (Funzione:NomeFunzione linea:NumeroLinea)	39
4360	Errore in allocazione di memoria durante l'esecuzione (Funzione:NomeFunzione linea:NumeroLinea)	39
4361	Troppi task attivi (Funzione:NomeFunzione linea:NumeroLinea)	40
4362	Formato matrice errato (Funzione:NomeFunzione linea:NumeroLinea)	40
4363	Troppe istruzioni ONINPUT attive (Funzione:NomeFunzione linea:NumeroLinea)	40
4364	Asse già impegnato in un riferimento locale (Funzione:NomeFunzione linea:NumeroLinea)	40
4365	Istruzione ONINPUT attivata sullo stesso input (Funzione:NomeFunzione linea:NumeroLinea)	40
4366	Troppe istruzioni ONFLAG attive (Funzione:NomeFunzione linea:NumeroLinea)	41
4367	Istruzione ONFLAG attivata sullo stesso FLAG (Funzione:NomeFunzione linea:NumeroLinea)	41
4368	Tentativo di scrittura di una variabile di tipo ReadOnly (Funzione:NomeFunzione linea:NumeroLinea)	41
4369	Troppi assi master attivi (Funzione:NomeFunzione linea:NumeroLinea)	41
4370	Troppi assi slave attivi (Funzione:NomeFunzione linea:NumeroLinea)	41
4372	Uso errato di una istruzione (Funzione:NomeFunzione linea:NumeroLinea)	41
4373	Impossibile eseguire la lettura del feed rate (Funzione:NomeFunzione linea:NumeroLinea)	42
4374	Troppe istruzioni di tipo IPC in esecuzione (Funzione:NomeFunzione linea:NumeroLinea)	42
4375	FASTREAD eseguita su assi di schede differenti (Funzione:NomeFunzione linea:NumeroLinea)	42
4378	Istruzione non abilitata (Funzione:NomeFunzione linea:NumeroLinea)	42
4379	L'istruzione non può essere utilizzata in funzioni lanciate da interrupt (Funzione:NomeFunzione linea:NumeroLinea)	42
4380	Troppe richieste di scrittura nell'area di memoria tamponata (Funzione:NomeFunzione linea:NumeroLinea)	43
4381	Non è possibile usare una linea seriale non ancora aperta (Funzione:NomeFunzione linea:NumeroLinea)	43
4382	Non è possibile aprire una linea seriale già aperta (Funzione:NomeFunzione linea:NumeroLinea)	43

4383 Si è cercato di aprire troppi processi ausiliari (Funzione:NomeFunzione linea:NumeroLinea)	43
4384 Il processo ausiliario non sta girando (Funzione:NomeFunzione linea:NumeroLinea)	43
4385 Si è cercato di aprire un processo ausiliario da un altro task (Funzione:NomeFunzione linea:NumeroLinea)	43
4391 Errore durante l'attivazione del SYSOK (Funzione:NomeFunzione linea:NumeroLinea)	44
4394 Troppi errori di ciclo (Funzione:NomeFunzione linea:NumeroLinea)	44
4395 Troppi messaggi (Funzione:NomeFunzione linea:NumeroLinea)	44
4397 Stack overflow (Funzione:NomeFunzione linea:NumeroLinea)	44
4398 Stack underflow (Funzione:NomeFunzione linea:NumeroLinea)	44
4399 Parametro fuori range (Funzione:NomeFunzione linea:NumeroLinea)	44
4865 Manca la definizione della macchina per l'interpolazione (G216 o G217)	44
4866 Manca la definizione degli indici della configurazione macchina in presa (M6)	45
6.2.10 Errori generati dal driver delle comunicazioni CNCTPA	45
16385 Modulo scollegato	45
16386 Modulo collegato	45
16387 Modulo ricollegato	45
16388 Modulo inizializzato	45
16389 Il modulo ha interrotto il collegamento	45
16641 Il firmware del controllo non risponde ai comandi	45
16642 TpaSock non risponde ai comandi	46
16643 Il Sistema Operativo non permette di usare RTX	46
16645 Errore nell'invio del codice firmware	46
16646 Non è stato possibile rimettere in esecuzione il codice firmware	46
16897 Non è stato installato RTX	46
16898 L'utente non ha i diritti di amministratore	46
16899 La dimensione della RAM del modulo è errata	47
16900 L'indirizzo IP del modulo è errato	47
16901 Il modulo è già collegato ad un altro impianto	47
16902 Il modulo non è configurato	47
16903 Le impostazioni del firewall impediscono la comunicazione	47
16904 Scheda di rete non presente o disabilitata	47
16905 Manca il codice firmware del controllo	48
16906 Versione RTX incompatibile con il codice firmware del controllo	48
16907 Versione del sistema operativo incompatibile con il codice di firmware del controllo	48
17153 TipoScheda: manca il codice firmware del trasmettitore GreenBUS	48
17154 TipoScheda: manca la parte di codice firmware del trasmettitore GreenBUS	48
17155 TipoScheda: errore nell'invio del codice di bootstrap del trasmettitore GreenBUS	48
17156 TipoScheda: errore nell'invio del codice di Main del trasmettitore GreenBUS	48
17157 TipoScheda: manca il codice di bootstrap	49
17158 TipoScheda: manca il codice di Main	49
17159 TipoScheda: errore nell'invio del codice di bootstrap	49
17160 TipoScheda: errore nell'invio del codice di Main	49
17409 Non è stato possibile inviare l'eseguibile ausiliario	49
17410 Non è stato possibile mettere in esecuzione l'eseguibile ausiliario	49
17667 NomeDLL: non è stato possibile mettere in esecuzione il codice firmware	50
17668 NomeDLL: non è stato possibile ottenere il puntatore alla RAM condivisa	50
17921 Non è stato possibile inviare NODETPA	50
17922 NODETPA non è ripartito	50
17923 NODETPA non è in esecuzione	50
18177 NODETPA ha cercato di accedere ad un indirizzo non valido	50
6.3 Segnalazioni generiche	51
6.3.1 Albatros inizia l'esecuzione	51

6.3.2	Albatros termina l'esecuzione	51
6.3.3	Il computer entra nella modalità di sospensione	51
6.3.4	Il computer esce dalla modalità di sospensione	51
6.3.5	Spegnimento del computer	51
6.3.6	Attuale livello di accesso	51
6.3.7	Aggiornamento software dei moduli	51
6.3.8	Invio della configurazione ai moduli	52
7	Configurazione del Sistema	53
7.1	Introduzione	53
7.2	Configurazione dei Dispositivi	53
7.2.1	Introduzione	53
7.2.2	Dispositivo generico	53
7.2.3	Uscita digitale	54
7.2.4	Ingresso analogico	54
7.2.5	Asse	54
	Dati di Base	54
	Parametri di Movimento	54
	Parametri di Interpolazione	55
	Altri Parametri	55
	Parametri Riferimento	55
	Livelli di Accesso	56
	Concatenamento Assi	56
	Correttori di linearità	56
7.3	Configurazione Logica	56
7.3.1	Configurazione di impianto	56
7.3.2	Configurazione dei gruppi	57
7.4	Configurazione Fisica	59
7.4.1	Configurazione di sistema	59
7.4.2	Configurazione Hardware	59
	Configurazioni predefinite	60
	Configurare un nodo di un bus TPA	60
	Configurare un nodo di un bus CAN	62
	La scheda di controllo del bus	62
	Il nodo CAN	63
	Inserire un nuovo nodo	63
	Configurare un nodo	63
	Caratteristiche della gestione EtherCAT in Albatros	63
	Premessa	63
	Configurazione dell'hardware EtherCAT	63
	Descrizione di un PDO	64
	Modifica di un PDO di un azionamento	65
	PDO aggiuntivi	66
	Acquisizione automatica nodi EtherCAT	66
7.4.3	Configurazione Virtuale-fisico	67
7.4.4	Mappe di cablaggio	68
7.5	Elenco dei tasti per navigare all'interno di una struttura ad albero	68
8	Strumenti di sviluppo	70
8.1	Editor GPL	70
8.1.1	Funzionalità editor GPL	70
	Usò delle espressioni regolari	71
8.1.2	Inserisci messaggio	73

8.1.3	Crittografia	73
8.1.4	Elenco dei tasti di scelta rapida disponibili	74
8.2	Librerie	76
8.3	Debug	76
8.3.1	Il debugger	76
8.3.2	Task in esecuzione	77
8.3.3	Tutti i task	77
8.3.4	Chiamate a funzione	77
8.3.5	Punti di interruzione	78
8.3.6	Contenuto di variabile	78
8.3.7	Elenco dei tasti di scelta rapida disponibili	79
8.4	Inizializzazione del controllo	79
8.4.1	Connessioni di rete	79
8.4.2	Diagnostica hardware	80
	Topologia rete EtherCAT	80
	Visualizzazione e modifica di oggetti nei nodi	80
8.5	Test	81
8.5.1	Memorizza variabile globale	81
8.5.2	Esegui funzione	81
8.5.3	Importazione messaggi	82
8.5.4	Nota utente nel file del report degli allarmi	83
8.6	Strumenti	83
8.6.1	Personalizza...	83
8.7	Browser	85
8.7.1	Il browser	85
8.7.2	Cerca identificatore...	85
8.7.3	Elenco dei tasti di scelta rapida disponibili	86
9	Programmi accessori	87
9.1	XConfMerge: programma per il merge di file di configurazione	87
9.2	XParMerge: programma per il merge di due file di parametrica	88
10	Linguaggio GPL	89
10.1	Concetti base	89
10.1.1	Introduzione al linguaggio GPL	89
10.1.2	Convenzioni e terminologia	89
10.1.3	Le variabili	91
	Tipi di dato	91
	Conversione dati	93
	Dichiarazione e visibilità delle variabili	93
	Modificatori	94
	Assegnazione di RANGE	95
	Diritti di Lettura / Scrittura	95
10.1.4	Le costanti	95
	Costanti predefinite con valore preassegnato	96
	Costanti predefinite con valore preassegnato in avvio di Albatros	97
10.1.5	Parole chiave	97
10.1.6	Le funzioni	99
10.1.7	I parametri di tipo dispositivo	101
10.1.8	Il Multitasking	101
10.1.9	Le Comunicazioni	102

10.1.10	Variabili da usare nella programmazione	103
10.1.11	Gli Assi	103
10.1.12	Correttori di linearità	106
10.1.13	Gestione dei messaggi in lingua	106
10.1.14	Gestione degli errori di sistema	106
10.2	Funzioni speciali	107
10.2.1	Personalizzazione movimento assi	107
10.2.2	Funzioni standard di movimentazione e taratura	109
10.2.3	Funzione OnUIEnd#	113
10.2.4	Funzione OnUIPlugged#	113
10.2.5	Funzione OnUIUnPlugged#	113
10.3	Istruzioni	113
10.3.1	Convenzioni	113
10.3.2	Tipologia delle istruzioni del linguaggio GPL	113
10.3.3	Input/Output	120
	GETFEED	120
	INPANALOG	120
	INPFLAGPORT	120
	INPPORT	120
	MULTIINPPORT	120
	MULTIOUTPORT	121
	MULTIRESETFLAG	121
	MULTIRESETOUT	121
	MULTISETFLAG	122
	MULTISETOUT	122
	MULTIWAITFLAG	122
	MULTIWAITINPUT	122
	OUTANALOG	123
	OUTFLAGPORT	123
	OUTPORT	123
	RESETFLAG	123
	RESETOUT	124
	SETFLAG	124
	SETOUT	124
	WAITFLAG	124
	WAITINPUT	125
	WAITPERSISTINPUT	125
10.3.4	Assi	126
	CHAIN	126
	CIRCABS	126
	CIRCINC	127
	CIRCLE	128
	COORDIN	129
	DISABLECORRECTION	130
	EMERGENCYSTOP	130
	ENABLECORRECTION	131
	ENDMOV	131
	FASTREAD	131
	FREE	132
	HELICABS	132
	HELICINC	133
	JERKCONTROL	133
	JERKSMOOTH	134
	LINEARABS	134
	LINEARINC	135
	MOVABS	135

MOVINC	136
MULTIABS	136
MULTIINC	137
NORMAL	138
RESRIFLOC	138
SETINDEXINTERP	138
SETLABELINTERP	138
SETPFLY	139
SETPFLYCHAINSTRAT	139
SETPZERO	140
SETPZEROCHAINSTRAT	140
SETQUOTE	140
SETQUOTECHAINSTRAT	141
SETRIFLOC	141
SETTOLERANCE	141
START	143
STARTINTERP	143
STOP	143
SWITCHENC	144
WAITACC	144
WAITCOLL	144
WAITDEC	145
WAITREG	145
WAITSTILL	146
WAITTARGET	146
WAITWIN	146
Parametri Asse	147
Lettura/Scrittura	147
DEVICEID	147
GETAXIS	147
Movimento Punto-Punto	152
SETACC	152
SETDEC	152
SETDERIV	153
SETFEED	153
SETFEEDF	153
SETFEEDFA	153
SETINTEG	154
SETMULTIFEED	154
SETPROP	154
SETSLOPE	154
SETVEL	155
Movimento Interpolato	155
LOOKAHEAD	155
SETACCI	155
SETACCLIMIT	156
SETACCSTRATEGY	156
SETAXPARTYPE	156
SETCONTORNATURE	156
SETDECI	157
SETDERIVI	157
SETFEEDFAI	157
SETFEEDI	158
SETFEEDFI	158
SETINTEGI	158
SETPROPI	158
SETSLOPEI	159

	SETSLOWPARAM	159
	SETVELI	160
	SETVELILIMIT	160
	Movimento Coordinato	160
	SETFEEDCOORD	160
	SETOFFSET	161
	Movimento Concatenato	162
	RATIO	162
	SETDYNRATIO	162
	Parametri Generici	162
	DYNLIMIT	162
	ENABLESTARTCONTROL	163
	NOTCHFILTER	163
	RESLIMNEG	164
	RESLIMPOS	164
	SETADJUST	164
	SETBACKLASH	165
	SETBIGWINFACTOR	166
	SETDEADBAND	167
	SETENCLIMIT	167
	SETINDEXEN	167
	SETINTEGTIME	168
	SETIRMPP	168
	SETLIMNEG	168
	SETLIMPOS	168
	SETMAXER	169
	SETMAXERNEG	169
	SETMAXERPOS	169
	SETMAXERTYPE	170
	SETPHASESINV	171
	SETREFINV	171
	SETRESOLUTION	172
10.3.5	Contatori	172
	DECOUNTER	172
	INCOUNTER	172
	SETCOUNTER	172
10.3.6	Timer	172
	HOLDTIMER	172
	SETTIMER	173
	STARTTIMER	173
10.3.7	Variabili, Vettori e Matrici	173
	CLEAR	173
	FIND	174
	FINDB	174
	LASTELEM	174
	LOCAL	174
	MOVEMAT	175
	PARAM	175
	SETVAL	176
	SORT	176
10.3.8	Stringhe	177
	ADDSTRING	177
	CONTROLCHAR	177
	LEFT	177
	LEN	177
	MID	178
	RIGHT	178

	SEARCH	178
	SETSTRING	178
	STR	179
	VAL	179
10.3.9	Comunicazioni	179
	CLEARRECEIVE	179
	COMCLEARRXBUFFER	179
	COMCLOSE	180
	COMGETERROR	180
	COMGETRXCOUNT	180
	COMOPEN	180
	COMREAD	181
	COMREADSTRING	181
	COMWRITE	182
	COMWRITESTRING	182
	RECEIVE	182
	SEND	187
	SENDIPC	192
	WAITIPC	193
	WAITRECEIVE	193
10.3.10	Matematica	193
	ABS	193
	ADD	194
	AND	194
	ARCCOS	194
	ARCSIN	195
	ARCTAN	195
	COS	195
	DIV	195
	EXP	196
	EXPR	196
	LOG	197
	LOGDEC	197
	MOD	198
	MUL	198
	NOT	198
	OR	199
	RANDOM	199
	RESETBIT	199
	ROUND	200
	SETBIT	200
	SHIFTL	201
	SHIFTR	202
	SIN	204
	SQR	204
	SUB	204
	TAN	205
	TRUNC	205
	XOR	205
10.3.11	Multitasking	206
	ENDMAIL	206
	ENDREALTIMETASK	206
	ENDTASK	206
	GETPRIORITYLEVEL	206
	GETREALTIME	207
	GETREALTIMECOUNT	207
	HOLDTASK	207

RESUMETASK	207
SENDMAIL	207
SETPRIORITYLEVEL	208
STARTREALTIMETASK	208
STARTTASK	209
STOPTASK	209
WAITMAIL	209
WAITTASK	209
10.3.12 Gestione di Flusso	210
CALL	210
DELONFLAG	210
DELONINPUT	210
FCALL	210
FOR/NEXT	211
FRET	211
GOTO	212
IF/IFVALUE/IF-THEN-ELSE	212
IFACC	213
IFAND	213
IFBIT	214
IFBLACKBOX	214
IFCHANGELEVEL	215
IFCOUNTER	215
IFDEC	216
IFDIR	216
IFERRAN	216
IFERROR	217
IFFLAG	217
IFINPUT	218
IFMESSAGE	219
IFOR	219
IFOUTPUT	220
IFQUOTER	220
IFQUOTET	221
IFRECEIVED	221
IFREG	222
IFSAME	222
IFSTILL	222
IFSTR	222
IFTARGET	223
IFTASKHOLD	223
IFTASKRUN	224
IFTIMER	224
IFVEL	224
IFWIN	225
IFXOR	225
ONERRSYS	226
ONFLAG	227
ONINPUT	227
REPEAT/ENDREP	227
RET	228
SELECT	228
TESTIPC	229
TESTMAIL	229
10.3.13 Varie	230
CLEARERRORS	230
CLEARMESSAGES	230

	DEFMSG	230
	DELAY	231
	DELERROR	232
	DELMESSAGE	232
	ERROR	232
	IFDEF/ELSEDEF/ENDDEF	234
	MESSAGE	236
	SYSFAULT	238
	SYSOK	238
	TYPEOF	238
	WATCHDOG	238
10.3.14	MECHATROLINK-II	239
	MECCOMMAND	239
	MECGETPARAM	240
	MECGETSTATUS	240
	MECSETPARAM	242
10.3.15	Bus di campo standard	243
	AXCONTROL	243
	AXSTATUS	244
	CNBYDEVICE	245
	READDICTIONARY	246
	WRITEDICTIONARY	246
10.3.16	EtherCAT	246
	ACTIVATEMODE	246
	ECATGETREGISTER	247
	ECATSETREGISTER	247
	GETPDO	247
	SETEOE	247
	SETPDO	248
10.3.17	Schede TMSbus con gestione CAN	248
	GETCNSTATE	248
	GETSDOERROR	248
	GETMNSTATE	249
	RECEIVEPDO	249
	SENDPDO	249
	SETNMTSTATE	249
10.3.18	Simulazione	250
	DISABLE	250
	DISABLEFORCEDINPUT	250
	ENABLE	250
	ENABLEFORCEDINPUT	250
	RESETFORCEDINPUT	251
	SETFORCEDANALOG	251
	SETFORCEDINPUT	251
	SETFORCEDPORT	251
10.3.19	BlackBox	251
	ENDBLACKBOX	252
	PAUSEBLACKBOX	252
	STARTBLACKBOX	252
10.3.20	ISO	253
	ISOG0	253
	ISOG1	253
	ISOG9	254
	ISOG90	254
	ISOG91	254
	ISOG93	255
	ISOG94	255

	ISOG216	255
	ISOG217	256
	ISOM2	256
	ISOM6	256
	ISOSETPARAM	257
	KINEMATICEXPR	258
10.3.21	Istruzioni non più disponibili	259
10.3.22	Istruzioni non utilizzabili su evento	260
10.4	Esempi	261
10.4.1	Azzeramento su Interrupt	261
10.4.2	Server di movimentazione assi	262
10.4.3	Ciclo di Main con gestione errori	264
10.4.4	Operazioni sulle stringhe	265
10.4.5	Esecuzione Sequenziale / Parallela	266
10.4.6	Routine di Azzeramento	266
10.4.7	Movimenti ISO	267

1 Introduzione

1.1 Come utilizzare la guida

La presente guida descrive le funzionalità del controllo numerico Albatros.

La guida è stata impostata in modo da condurre l'operatore nella comprensione del sistema e del suo utilizzo.

I punti salienti di ciascuna sezione della guida sono:

- le finestre e gli strumenti di Albatros.
- la descrizione dell'Architettura tipica di un sistema Albatros.
- come visualizzare i dispositivi e operare sugli stessi con funzionalità di manuale e diagnostica, servendosi del Quadro sinottico.
- come visualizzare e modificare i Parametri Tecnologici e Geometrici ed i Parametri Utensili
- come visualizzare i Dispositivi e agire sugli stessi con funzionalità di manuale e diagnostica

Al fine di evitare un eccessivo appesantimento della presente guida, si rimanda il lettore alla consultazione dei manuali del Sistema Operativo Windows, per un completo apprendimento dei meccanismi d'uso del *mouse*, dei *menu*, delle *toolbar* e di tutte le funzionalità operative classiche di Windows.

1.2 Le finestre di lavoro

Le finestre di lavoro sono di vario tipo, dipendono dal tipo di operazione che si intende svolgere e possono essere tenute aperte contemporaneamente.

Le finestre sono divise nei seguenti tipi:

Finestra	Descrizione
<i>Principale</i>	finestra principale di Albatros. Consente il richiamo delle funzioni ed è il contenitore delle altre finestre il cui contenuto dipende dall'applicazione specifica che esse rappresentano.
Sinottico	contiene una rappresentazione grafica della macchina o di parti di essa, consentendo di operare sulle stesse.
Parametri Tecnologici	consente la visualizzazione e la modifica dei parametri tecnologici.
Parametri Utensili	consente la visualizzazione e la modifica dei parametri utensili.
Diagnostica	consente la visualizzazione dello stato dei dispositivi e, nei casi dove è consentito, anche di operare sugli stessi.
Errori di sistema	finestra con la lista degli ultimi errori di sistema verificatisi. È possibile visualizzare anche gli errori di ciclo ed i messaggi.
Configurazione di sistema	consente la visualizzazione e la modifica dei dispositivi logici e fisici della macchina


2 Composizione del sistema

2.1 Diritti di accesso al sistema

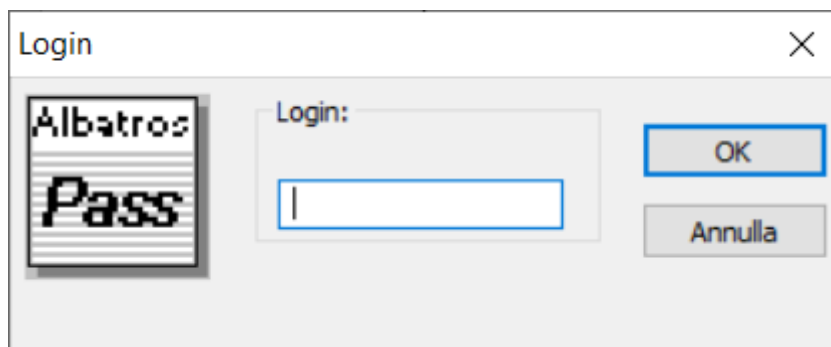
Albatros prevede quattro livelli di accesso al sistema:

- Utente: è il livello con le maggiori restrizioni di accesso. Non è possibile modificare nessuna delle impostazioni relative ai dispositivi. Questo livello viene utilizzato per eseguire le lavorazioni e le normali operazioni di macchina. Quando il sistema viene avviato si attiva automaticamente questo livello di accesso.
- Assistenza: è il livello utilizzato per effettuare la manutenzione ordinaria alla macchina. L'operatore dovrebbe poter modificare alcuni parametri di configurazione ma non apportare modifiche alla struttura della macchina.
- Costruttore: è il livello utilizzato per configurare gli impianti e le macchine. A questo livello si possono apportare quasi tutte le modifiche possibili. Viene utilizzato dagli sviluppatori.
- Tpa: è il massimo livello di accesso al sistema. Serve per proteggere l'accesso ad alcune impostazioni particolarmente delicate la cui modifica richiede una conoscenza approfondita di Albatros. Viene utilizzato molto raramente. La password di accesso a questo livello, deve essere richiesta direttamente alla TPA.

Per accedere al sistema con un livello di accesso superiore a quello Utente o per riportare a livello Utente il sistema dopo aver apportato delle modifiche con un livello superiore, si deve inserire la password corrispondente.

Per richiamare la finestra in cui si inserisce la password si deve utilizzare la combinazione di tasti **Ctrl + *(asterisco)**. In alternativa sulla destra della **Barra delle applicazioni** di Windows è presente l'icona . Cliccando con il tasto destro del mouse sull'icona è possibile visualizzare un menu in cui compare la voce **Change pass level**.

La finestra che si apre si presenta così:



Finestra di login

A questo punto va inserita la password e premuto il pulsante **[OK]** per conferma. In sostituzione ai caratteri che compongono la password vengono visualizzati dei caratteri "*", in modo che nessuno possa leggere la password appena digitata.

Una volta inserita la password si è già al livello di accesso corrispondente. Per avere una conferma del livello di accesso si può selezionare la voce **Informazioni su Albatros** dal menu **?**.

Quando la password inserita non è corretta viene visualizzato l'errore "Attenzione! Password errata!".

2.2 Supporto multilingue

Albatros supporta la visualizzazione del testo in diverse lingue.

Cambio lingua

Il cambio lingua può essere effettuato a qualsiasi [livello di accesso](#) al sistema. Per cambiare la selezione della

lingua si deve utilizzare la combinazione di tasti **Ctrl + /** oppure cliccare sull'icona  dalla "**barra delle applicazioni**" di Windows

Nella finestra che si apre selezionare la lingua desiderata e fare clic sul pulsante **[OK]**.

Il cambio della lingua non viene eseguito immediatamente, ma al successivo riavvio di Albatros.

2.3 Architettura tipica di un sistema

Poiché molti aspetti, nelle rappresentazioni grafiche e nella struttura dei dati di base di Macchina, dipendono fortemente dalla tipologia della Macchina, oltre ad alcune informazioni di carattere generale, la presente Guida fornisce una descrizione della composizione di un sistema tipico a titolo esemplificativo.

Le indicazioni reali, gli schemi e le pagine grafiche del sistema vero e proprio dipendono ovviamente dalla specifica applicazione, e sono pertanto predisposte a cura del Costruttore della Macchina Utensile.

Il controllo numerico Albatros è costituito da un PC supervisore, che espone l'interfaccia Operatore-Macchina, e da un numero variabile di Moduli (da 1 a 16), per il pilotaggio e il controllo di tutte le risorse operative della Macchina Utensile o dell'Impianto.

Si possono quindi avere due tipologie di impianto:

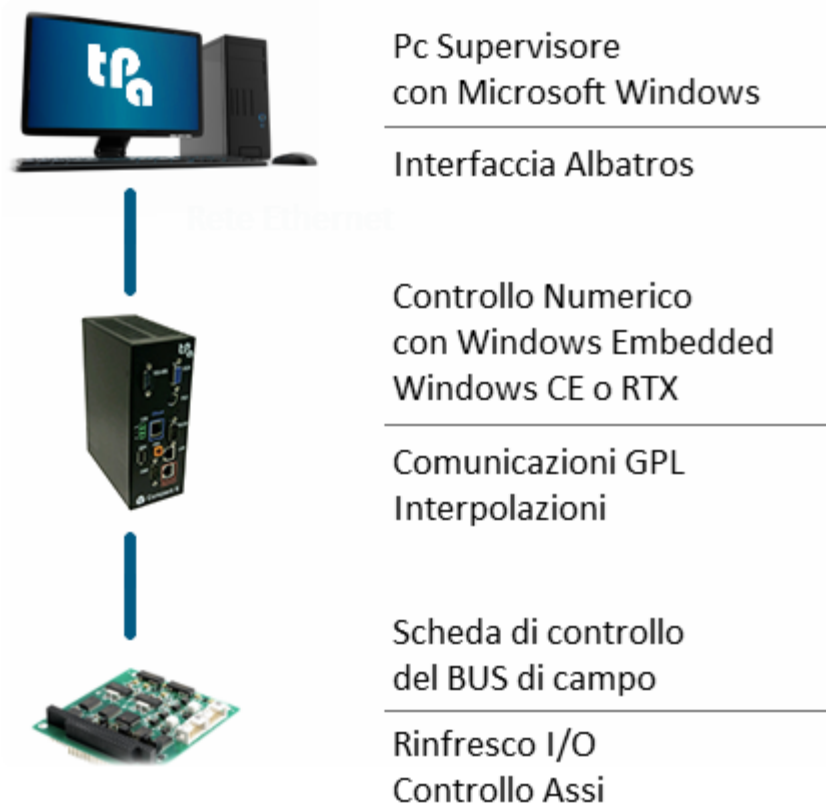
Monomodulo costituita da un solo modulo collegato direttamente al bus del PC.

Multimodulo costituita da un minimo di 1 a un massimo di 16 moduli e prevista solitamente per applicazioni su Impianti o Linee di più macchine; l'unità PC in questo caso è fisicamente separata dai Moduli, che possono essere alloggiati in punti diversi della Linea o dell'Impianto.

In entrambe le architetture, i moduli sono costituiti da una o più schede assi per il controllo diretto degli Assi di Macchina e la gestione logica dei dispositivi di Ingresso/Uscita.

Nella versione monomodulo le schede assi sono installate direttamente sul PC Supervisore, nella versione multimodulo sono installate in un PC industriale (con o senza video e tastiera) collegato al PC Supervisore via rete ethernet. Nella seguente figura è riportato uno schema di collegamento tra il PC Supervisore ed un modulo remoto. Sono inoltre riportate le principali attività svolte dai vari componenti.

SISTEMA DI COLLEGAMENTO DI UN MODULO REMOTO



Dispositivi remoti intelligenti pilotano dispositivi di I/O e assi (remoto TRS-AX) direttamente a bordo macchina. Questi dispositivi provvedono alla lettura delle linee di Ingresso Digitali (ON / OFF) o Analogiche ed al rinfresco delle linee di Uscita Digitali o Analogiche e sono connessi ai Moduli tramite GreenBUS (bus seriale RS485 - 1 Mbaud) bus CAN e Ethercat.

Il funzionamento di Albatros in macchina è protetto dalla presenza di una chiave hardware usb, configurata da TPA.

2.4 Organizzazione e configurazione logica

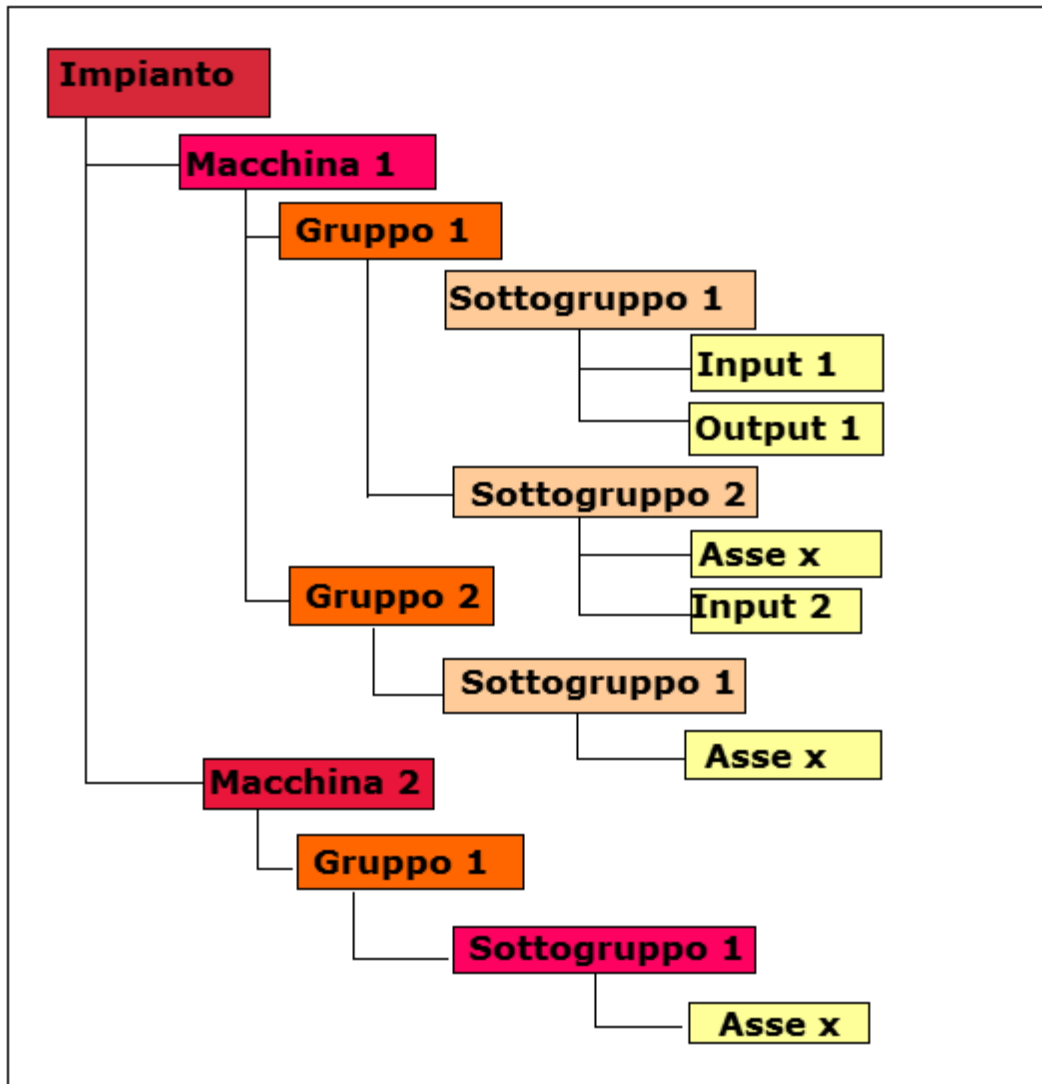
Nel sistema Albatros la struttura descrittiva dell'impianto o della singola macchina utensile è organizzata in un archivio tecnologico con una struttura gerarchica.

Questa impostazione riflette l'esigenza di conservare, a livello di dati di configurazione e di modalità di accesso, l'eventuale struttura modulare delle macchine, classificandola in termini di aggregazione dinamica di diversi moduli, aggregati e dispositivi, inseribili od escludibili in funzione dei diversi possibili allestimenti.

Seguendo questa struttura logica, nel caso più generale e complesso, abbiamo il livello gerarchico superiore costituito da:

- 1. Impianto** non è altro che un insieme di macchine. È inteso come l'insieme degli elementi operativi gestiti dal Controllo Numerico. L'impianto è sempre presente, anche nel caso di una singola macchina, e non deve essere definito esplicitamente.
- 2. Macchina** viene vista da un punto di vista "logico" come un insieme di dispositivi (assi, timer etc.) e di cicliche di controllo, ovvero di codice scritto in GPL che implementa gli algoritmi di controllo della macchina stessa. Generalmente in una macchina è presente un elevato numero di dispositivi che vengono organizzati in gruppi
- 3. Gruppi** sono dei "contenitori" che permettono di organizzare i componenti di una macchina secondo dei criteri logici. Ad esempio si può definire un gruppo "assi" che contiene tutti gli assi della macchina, gli switch di fine corsa, le cicliche che eseguono l'azzeramento degli assi, etc.
- 4. Sottogruppi** rappresentano una ulteriore specializzazione di un gruppo. Ad esempio il gruppo "assi" potrebbe essere suddiviso in "assi-digitali" e "assi-passo-passo".
- 5. Dispositivi** costituiscono il livello più basso della gerarchia. Sono una rappresentazione logica dei componenti elettrici e meccanici della macchina e sono indipendenti dall'hardware sottostante.

Nella figura seguente viene schematizzata la struttura di un ipotetico impianto composto da due macchine:



Esempio di organizzazione gerarchica di un impianto.

NOTA: i Gruppi possono anche non essere suddivisi ulteriormente in Sottogruppi ed essere formati direttamente da Dispositivi.



Per accedere ad alcune operatività come Diagnostica, Configurazione di sistema, Parametriche nel caso di un impianto con più macchine è necessario selezionare la macchina di cui si vogliono visualizzare i dati.






2.5 I Dispositivi

I dispositivi si possono raggruppare in due categorie: dispositivi fisici e dispositivi logici. Nel sistema tutti i dispositivi sono individuati per mezzo di un nome che ne identifica l'utilizzo.

Dispositivi fisici













Per dispositivi fisici si intendono tutti quegli elementi, che agiscono sulle parti elettriche o pneumatiche della macchina o che ne rilevano lo stato, questi sono:

Simbolo	Dispositivo	Funzionalità
	<i>Ingresso digitale</i>	rileva lo stato "acceso" o "spento" di un dispositivo. Ad esempio, l'interruttore di sicurezza di un portellone.
	<i>Uscita digitale</i>	attiva o disattiva un dispositivo, ponendolo quindi nello stato "acceso" o "spento". Può essere utilizzato, ad esempio, per pilotare un'elettrovalvola.

Simbolo	Dispositivo	Funzionalità
	Ingresso analogico	rileva il valore di una tensione in ingresso al morsetto corrispondente. Ad esempio, il valore generato da una dinamo tachimetrica.
	Uscita analogica	impone una tensione in uscita al morsetto corrispondente. Può, ad esempio, essere utilizzato per pilotare un inverter.
	Porto di ingressi	è composto da 8 linee di ingresso digitali.
	Porto di uscite	è composto da 8 linee di uscita digitali.
	Asse	gestisce la movimentazione di un asse elettrico. È possibile gestire assi di vario tipo: con controllo analogico, con controllo digitale, motori passo-passo, assi di conteggio (solo lettura encoder).

Dispositivi logici

I dispositivi logici sono elementi che agiscono esclusivamente all'interno dei programmi di lavoro e non hanno una controparte fisica:

Simbolo	Dispositivo	Funzionalità
	Timer	elemento di conteggio del tempo. L'unità di misura è il secondo. Ha una risoluzione di 4 ms. Può assumere solamente valori positivi ed il valore massimo rappresentabile è di 8.589.934 secondi (circa 99 giorni) (con real-time a 250 Hz). Il suo valore viene memorizzato nella memoria non volatile della scheda assi.
	Contatore	elemento di conteggio di operazioni. Può assumere un valore compreso tra -2.147.483.648 e +2.147.483.647. Il suo valore viene memorizzato nella memoria non volatile della scheda assi.
	Flag bit	elemento che può assumere un valore "acceso" o "spento".
	Flag switch	sono dei flag particolari che possono essere associati ad alcuni pulsanti della barra degli strumenti, come ad esempio il flag di Start.
	Porto di flag	è composto da 8 linee di flag bit.
	Variabile	variabile globale tipo <i>integer</i> del codice GPL.
	Variabile	variabile globale tipo <i>char</i> del codice GPL.
	Variabile	variabile globale tipo <i>float</i> del codice GPL.
	Variabile	variabile globale tipo <i>double</i> del codice GPL.
	Variabile	variabile globale tipo <i>stringa</i> del codice GPL.
	Variabile	variabile globale tipo <i>array</i> del codice GPL.
	Variabile	variabile globale tipo <i>matrice</i> del codice GPL.

3 Quadro sinottico

3.1 Uso del Quadro Sinottico

Durante il funzionamento della macchina, è possibile aprire la finestra del *Quadro Sinottico* che permette di visualizzare lo stato dei dispositivi più significativi.

Le informazioni presentate dai quadri sinottici coincidono con quelle della finestra diagnostica. Tuttavia, mentre in quest'ultima, le informazioni vengono presentate in una struttura ad albero, (che include tutti i dispositivi della macchina), i sinottici permettono di presentare le informazioni in modo grafico, (visualizzando ad esempio un'immagine della macchina e presentando le quote degli assi in prossimità di questi). Permettono inoltre di selezionare le informazioni più significative e di raggruppare le altre in schermate secondarie richiamabili dall'utente.

3.2 Come operare sul Quadro Sinottico

A scopo diagnostico, l'Operatore può selezionare le diverse pagine che compongono il sinottico, facendo *doppio click* con il mouse su una delle aree della macchina, delimitate nell'immagine da un rettangolo tratteggiato, chiamate anche "zone calde".

Per puntare a una "zona calda", a un dispositivo o a un asse, basta spostare il puntatore del mouse sull'immagine dell'oggetto desiderato. Contemporaneamente, nella Barra di stato compare il nome del dispositivo su cui il puntatore sta transitando.

Il puntatore del mouse assume forme diverse a seconda del tipo di oggetto a cui punta, indicando così le azioni consentite su quell'oggetto. Queste sono:



lente di ingrandimento

se è una "zona calda"



mano

se è un dispositivo di uscita



cursore di testo

se è una casella in cui è possibile impostare un valore

3.3 Come agire sui Dispositivi

L'azione sui dispositivi si attua puntando col mouse al dispositivo desiderato, completando poi l'azione come descritto di seguito (questa varia in base al tipo di dispositivo).

Modo di rappresentazione	Azione	Dispositivo
<i>Icona dispositivo</i>	puntare e fare <i>click</i>	Uscita digitale Flag switch Flag bit
<i>Casella di impostazione valore</i>	puntare e fare <i>click</i> e impostare il valore	Uscita analogica Porto di uscita Porto di flag Quota asse Timer Contatore




3.4 Movimento Assi in Manuale

Per accedere alla funzione di movimentazione manuale degli assi si devono possedere gli opportuni [diritti di accesso](#). Questi vengono assegnati dal costruttore della macchina.

Per interagire con uno degli assi è sufficiente un *doppio click* del mouse in corrispondenza del campo di visualizzazione delle quote dell'asse desiderato. Si ottiene così l'apertura della finestra per la movimentazione dell'asse. Nel caso di assi di tipo Virtuale, Passo-Passo e Conteggio, la finestra presenta un numero di informazioni ridotte. Ad esempio se l'asse è di tipo Conteggio vengono visualizzati i soli valori di Quota Reale e di Velocità.



Nella finestra sono presenti due zone che contengono:



Zona di visualizzazione



- Tre caselle contenenti la visualizzazione della *Quota reale* dell'asse [mm], della sua *Velocità* [m/min] e dell'*Errore d'anello* o di inseguimento.
- Due pulsanti di scelta che segnalano lo *Stato* dell'asse (*Free* = ad anello aperto, per esempio a causa di un errore di sistema, *Normal* = ad anello chiuso, cioè nel normale stato di controllo di posizione). Questi pulsanti consentono anche di impostare lo stato.
- La segnalazione, durante il movimento, dello *Stato* dell'asse (es. Accelerazione).
- Due pulsanti per eseguire nella *Direzione* negativa  o positiva  lo spostamento dell'asse.
- Il pulsante  per l'Arresto, in qualsiasi istante, del movimento dell'asse durante gli spostamenti nelle modalità Assoluto e Step.

Zona Movimento




- Due caselle per impostare una *Quota Negativa* ed una *Quota Positiva* che verranno utilizzate nella modalità *Assoluto*.
- Una casella per impostare la *Velocità* da imporre all'asse durante i movimenti in manuale.
- Tre pulsanti per la scelta del modo in cui verrà effettuato il movimento: *Jog*, a quota *Assoluta* o *Step*.
- Una casella per impostare il valore del *Passo* da utilizzare nella modalità *Step*.

Per movimentare un asse si dovranno impostare opportunamente i parametri descritti. Si dovrà scegliere la modalità di movimentazione e si dovrà premere il pulsante  (per muovere l'asse in direzione positiva) o il pulsante  (per muovere l'asse in direzione negativa).

In modalità *Jog* l'asse si muoverà finché il pulsante  o il pulsante  rimane premuto.

In modalità *Step* l'asse si muoverà della quantità specificata nella casella "passo" ogni volta che viene premuto il pulsante  o il pulsante .

In modalità *Assoluto* l'asse raggiungerà direttamente la quota impostata nella casella quota positiva o quota negativa.

In alternativa ai pulsanti ,  e  è possibile utilizzare i tasti "+" (o Ctrl+P), "-" (o Ctrl+M) e "barra spaziatrice" della tastiera.

4 Parametri Tecnologici e Utensili

4.1 La finestra Parametri Tecnologici

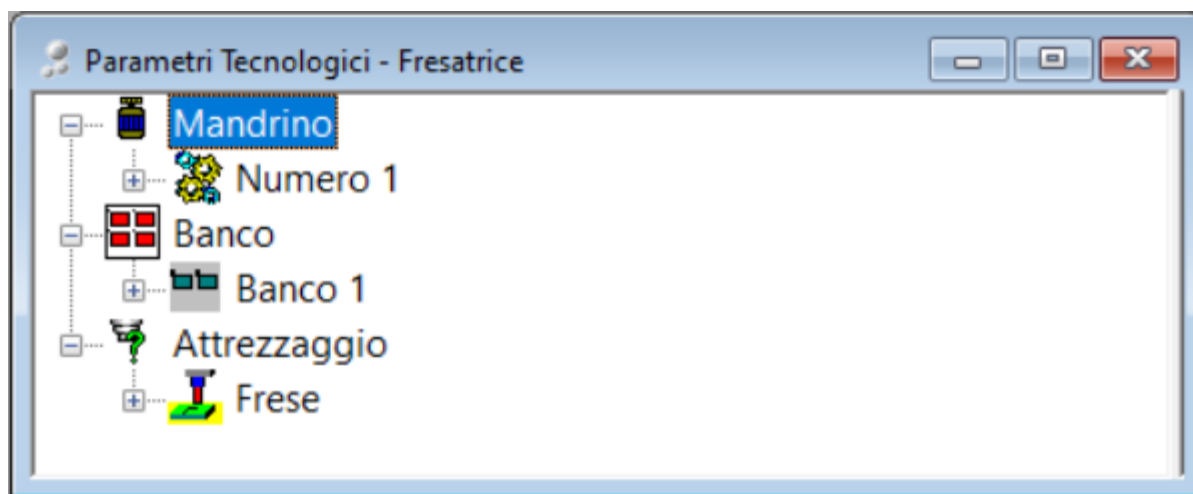
L'archivio dei Parametri Tecnologici permette di memorizzare le informazioni di carattere geometrico e tecnologico di una macchina. Queste informazioni sono necessarie al controllo numerico per poter gestire correttamente il funzionamento della macchina.

Si apre da menu **File->Apri Parametri Tecnologici**.



I Parametri Tecnologici sono solitamente organizzati in Gruppi / Sottogruppi (in generale i gruppi ed i sottogruppi dei Parametri Tecnologici non sono legati ai gruppi ed ai sottogruppi in cui vengono organizzati i dispositivi della macchina). Le modalità di visualizzazione vengono definite dal Costruttore di Macchina e differenziate in funzione della specifica applicazione.

I valori presenti nell'archivio sono, in genere, stabiliti dal costruttore in fase di taratura della Macchina e normalmente non possono, se non occasionalmente, essere modificati dall'Utente. È pertanto possibile che alcuni dati siano soggetti a protezione tramite Password per evitare accidentali modifiche che potrebbero compromettere la corretta funzionalità del sistema.

La finestra relativa ai Parametri Tecnologici visualizza in una struttura ad albero tutti i Gruppi e Sottogruppi di parametri che compongono l'archivio, come mostrato nella seguente figura.



Struttura dell'archivio dei Parametri Tecnologici

La finestra contiene, visualizzati in una struttura ad albero, alcuni Gruppi con i relativi Sottogruppi di parametri. La struttura ad albero può essere espansa o ridotta operando sui pulsanti  e  di ogni nodo. L'apertura e la chiusura delle parti che compongono l'albero può essere effettuata anche con i tasti: **+**, **-**, frecce di direzione destra e sinistra.

Come operare sui Parametri Tecnologici

Una volta aperto l'albero del Gruppo / Sottogruppo desiderato, si accede alla pagina che contiene i dati.

I dati possono apparire in forma di tabella oppure in caselle di testo o di selezione, ciò dipende dal tipo di dati e da come gli stessi sono stati impostati dal Costruttore.

Se vengono apportate delle modifiche ai dati è necessario premere il pulsante **[OK]** per rendere le modifiche permanenti.

Attrezzaggio

Un caso particolare tra i dati relativi ad una macchina è quello degli attrezzaggi. Tipicamente le informazioni relative all'insieme degli utensili di cui la macchina è dotata (attrezzaggio), vengono memorizzate nell'archivio dei Parametri Tecnologici. Le informazioni relative agli utensili vengono invece memorizzate nell'archivio dei Parametri Utensili.

Per definire l'attrezzaggio della macchina è quindi necessario legare le informazioni dei due archivi. Se l'applicazione lo prevede sarà quindi possibile richiamare dall'archivio dei Parametri Tecnologici le informazioni dell'archivio dei Parametri Utensili. Tipicamente il legame viene implementato per mezzo di un pulsante con un'icona simile alla seguente.



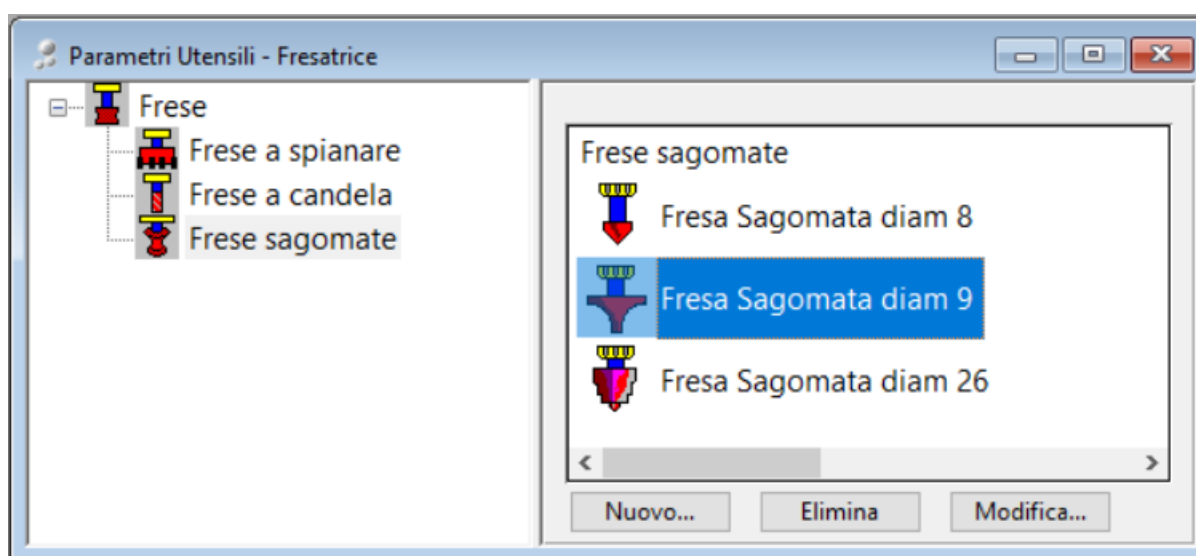
Selezionando l'icona ed eseguendo un doppio click con il tasto sinistro del mouse, si apre una finestra contenente la lista degli utensili definiti nell'archivio dei Parametri Utensili da cui è possibile selezionare l'utensile desiderato. Una volta eseguita la selezione, l'icona del pulsante cambia e viene visualizzata quella associata allo specifico utensile.

È, inoltre, possibile visualizzare i dati relativi all'utensile eseguendo sull'icona un doppio click con il tasto destro del mouse.

4.2 La finestra Parametri Utensili



La finestra dei Parametri Utensili si apre da menu **File->Apri Parametri Utensili**.

Le parametriche utensili, predisposte dal Costruttore in base alle lavorazioni che la macchina è in grado di eseguire, sono di solito organizzate come mostrato nella figura seguente:



Esempio di una finestra Parametri Utensili

La finestra Parametri Utensili si compone di due zone:

- la *zona di sinistra* contiene, visualizzati in una struttura ad albero, i Gruppi con i relativi Sottogruppi di utensili. La struttura ad albero può essere espansa o ridotta operando sui pulsanti  e  di ogni nodo. Si può ad esempio avere un Gruppo Frese composto da Sottogruppi di frese con caratteristiche diverse tra loro, quali frese sagomate, frese con brandeggio etc. Ognuno di questi sottogruppi ha associato uno o più utensili le cui caratteristiche vengono assegnate in una finestra di dialogo definita dal costruttore. Gli utensili presenti in ogni sottogruppo sono visualizzati nella zona di destra dello schermo.
- la *zona di destra*, che ha per titolo il nome del Sottogruppo selezionato, contiene l'elenco degli Utensili che fanno parte del Sottogruppo. Gli utensili definiti in questa sede non sono necessariamente presenti sulla macchina. L'associazione tra utensile e posizione in macchina (attrezzaggio) viene normalmente fatta nell'archivio dei parametri tecnologici.

Come operare sui Parametri Utensili

Le operazioni di inserimento, modifica e cancellazione di utensili dall'archivio sono rese possibili dai *pulsanti* che si trovano nella parte bassa della finestra:

-
- [Nuovo...]** consente di inserire nel Sottogruppo un nuovo utensile. Apre la finestra di dialogo "Nuovo utensile" dove è possibile assegnare i seguenti dati:
- *Descrizione*: è un messaggio che identifica l'utensile. La descrizione può essere scelta tra quelle già in elenco, sempre che non sia già stata assegnata ad un altro utensile, oppure se ne può definire una nuova.
 - *Immagine*: è una icona che identifica l'utensile. Può essere scelta tra quelle già in elenco, oppure può essere richiamata da una cartella utilizzando il pulsante **[Immagine]**. L'utensile viene inserito rispettando l'ordine alfabetico delle descrizioni.
- [Elimina]** consente di eliminare un utensile dal Sottogruppo, previa conferma; la relativa descrizione non viene eliminata e resta disponibile per un altro utensile.
- [Modifica...]** permette di sostituire la *descrizione* o l'*immagine* all'utensile selezionato, appare infatti la stessa finestra di dialogo descritta nel comando **[Nuovo...]**.

5 Diagnostica

5.1 La finestra Diagnostica

Durante la fase esecutiva della macchina, è possibile aprire la finestra *Diagnostica* che consente all'Operatore di tenere sotto controllo lo stato di funzionamento della macchina, monitorando lo stato logico dei segnali di I/O digitali, il valore degli I/O analogici, quello dei contatori e dei timer e il movimento degli assi. In funzione dei [diritti di accesso](#) assegnati dal costruttore è inoltre possibile modificare lo stato dei dispositivi.

È possibile, se consentito dal livello di accesso, avere in tempo reale:

- la visualizzazione dello stato (ON/OFF) di tutti i segnali di Ingresso e di Uscita digitale
- la possibilità di attivare e disattivare i segnali di Uscita digitali
- la visualizzazione dei valori correnti (nell'intervallo +/- 10V) degli ingressi Analogici
- la possibilità di assegnare un valore (nell'intervallo +/- 10V) a tutte le uscite Analogiche
- la possibilità di muovere un asse in Manuale selezionandone la velocità, il valore del Passo o la Quota assoluta finale, con la visualizzazione della posizione reale, della velocità e dell'errore d'anello.
- visualizzazione e modifica delle variabili globali

Nei prossimi paragrafi verranno descritti in dettaglio i dispositivi e le variabili globali e la loro rappresentazione grafica.

NOTA: Nella finestra diagnostica vengono visualizzati esclusivamente i dispositivi abilitati per il livello di accesso corrente.

5.2 Composizione della finestra Diagnostica

Con riferimento alla struttura "Gruppi / Sottogruppi", già descritta nel capitolo [Composizione del sistema](#), è possibile accedere ai dispositivi che vengono poi visualizzati in una struttura ad albero. Come capostipite della struttura troviamo il gruppo rappresentato dall'icona



, seguita dal Nome e da un Commento.

La struttura viene espansa o ridotta facendo *clic* sul pulsante o . L'apertura e la chiusura delle parti che compongono l'albero, può essere effettuata anche con i tasti: +, -, frecce di direzione destra e sinistra.

Aprendo un Gruppo compaiono nell'albero

- la "Lista dispositivi" del Gruppo, rappresentata dal simbolo .
- gli eventuali Sottogruppi che compongono il Gruppo.

Aprendo uno dei Sottogruppi compaiono i dispositivi che lo compongono.

5.3 Rappresentazione dei Dispositivi

Di ciascuno dei dispositivi visualizzati vengono mostrate le seguenti informazioni:

- il Simbolo grafico;
- lo Stato o il Valore correnti;
- il Nome;
- il Commento.

Di seguito viene mostrata la rappresentazione grafica dei dispositivi, il tipo di dispositivo e quale grandezza dello stesso viene visualizzata in tempo reale.

Lo stato degli ingressi digitali, delle uscite digitali e dei flag è rappresentato graficamente da un LED che assume un determinato colore, a seconda che l'ingresso sia attivo o disattivo.

Nel caso di Porti, quindi di più linee rappresentate contemporaneamente (8), apparirà una fila di LED; la prima linea del gruppo è rappresentata dal LED più a destra, l'ultima da quello più a sinistra.

Dispositivo	Symbol	Stato	Visualizzazione in tempo reale
Ingresso digitale			stato: Attivo = VERDE, Disattivo = GRIGIO
Uscita digitale			stato: Attivo = ROSSO, Disattivo = GRIGIO
Ingresso analogico		22.000	valore corrente
Uscita analogica		22.000	valore numerico corrente in Volt
Porto di ingressi			stato di ogni linea (come Ingresso digitale). Stato: Attivo = VERDE, Disattivo = GRIGIO
Porto di uscite			stato di ogni linea (come Uscita digitale). Stato: Attivo = ROSSO, Disattivo = GRIGIO
Asse		100.000	posizione assoluta corrente
Timer		12.000	valore corrente in secondi
Contatore		58	valore numerico corrente
Flag bit			stato: Attivo = GIALLO, Disattivo = GRIGIO
Flag switch			stato (come Flag bit). Stato: Attivo = GIALLO, Disattivo = GRIGIO
Porto di flag			stato di ogni linea (come Flag bit). Stato: Attivo = GIALLO, Disattivo = GRIGIO
Variabile globale		2	variabile globale tipo integer del codice GPL
Variabile globale		127	variabile globale tipo char del codice GPL
Variabile globale		50.00000000	variabile globale tipo float del codice GPL
Variabile globale		200.00000000	variabile globale tipo double del codice GPL
Variabile globale		Area	variabile globale tipo stringa del codice GPL
Variabile globale		[256]	variabile globale tipo array del codice GPL
Variabile globale		[10][3]	variabile globale tipo matrice del codice GPL

5.4 Interagire con i Dispositivi

A scopo diagnostico, è possibile interagire con un dispositivo per leggerne lo stato o modificarne il valore. Questo non è possibile per alcuni tipi di dispositivi, quali quelli di ingresso e per quelli protetti dal Costruttore. In questo caso un messaggio informerà l'Operatore nel caso tentasse di compiere operazioni sul dispositivo.

Una volta selezionato il dispositivo, fare *doppio clic* col mouse, oppure premere il tasto **Invio**, o la **Barra Spaziatrice**, per accedere alla finestra che consente di variare lo stato o il valore del dispositivo.

Nel caso si tratti di una **Uscita digitale** o di un **Flag bit** non appare nessuna finestra, ma viene immediatamente variato lo stato del dispositivo. Il buon funzionamento dell'uscita viene segnalato dalla variazione del colore del LED che ne indica lo stato.

Nel caso di **Porto di Uscite** bisogna puntare col mouse sul LED che corrisponde all'uscita desiderata e fare *doppio clic* per attuare la variazione dello stato.

Lo stesso dicasi per i **Flag switch** e per il **Porto di flag**.

Per le **Uscite analogiche**, i **Timer** e i **Contatori** appare una Finestra di dialogo che visualizza il valore corrente e che consente di impostare il nuovo valore che si desidera fare assumere immediatamente al dispositivo.

La modalità di interazione con un **Asse** è descritta nel punto [Movimento Assi in Manuale](#) a cui si rimanda.

5.5 Elenco dei tasti per navigare all'interno di una struttura ad albero

Tasto	Descrizione
Freccia Su	sposta la selezione alla riga immediatamente precedente o successiva
Freccia Giù	
Freccia Destra	esplode di un livello il ramo selezionato e se già esploso sposta la selezione sul ramo successivo
Freccia Sinistra	chiude il ramo selezionato e se già chiuso sposta la selezione sul ramo precedente
+	esplode di un livello il ramo selezionato
-	chiude il ramo selezionato

- *
 Ctrl+Alt+Shift e Enter visualizza le tabelle dei correttori di linearità associate ad un asse. Se la
 Ctrl+Alt+Shift e click visualizza le tabelle dei correttori di linearità associate ad un asse. Se la
 sinistro visualizza le tabelle dei correttori di linearità associate ad un asse. Se la
 combinazione dei tasti è attivata mentre è selezionato un asse nell'albero dei
 dispositivi di un modulo, vengono visualizzati tutti i correttori associati a un asse,
 come fossero una matrice, in cui le colonne sono gli assi associati (la prima colonna
 è quella degli autocorrettori), e le righe sono i valori di correzione. I valori
 eventualmente modificati vengono presi in carico durante il movimento dell'asse ma
 non vengono memorizzati su disco.

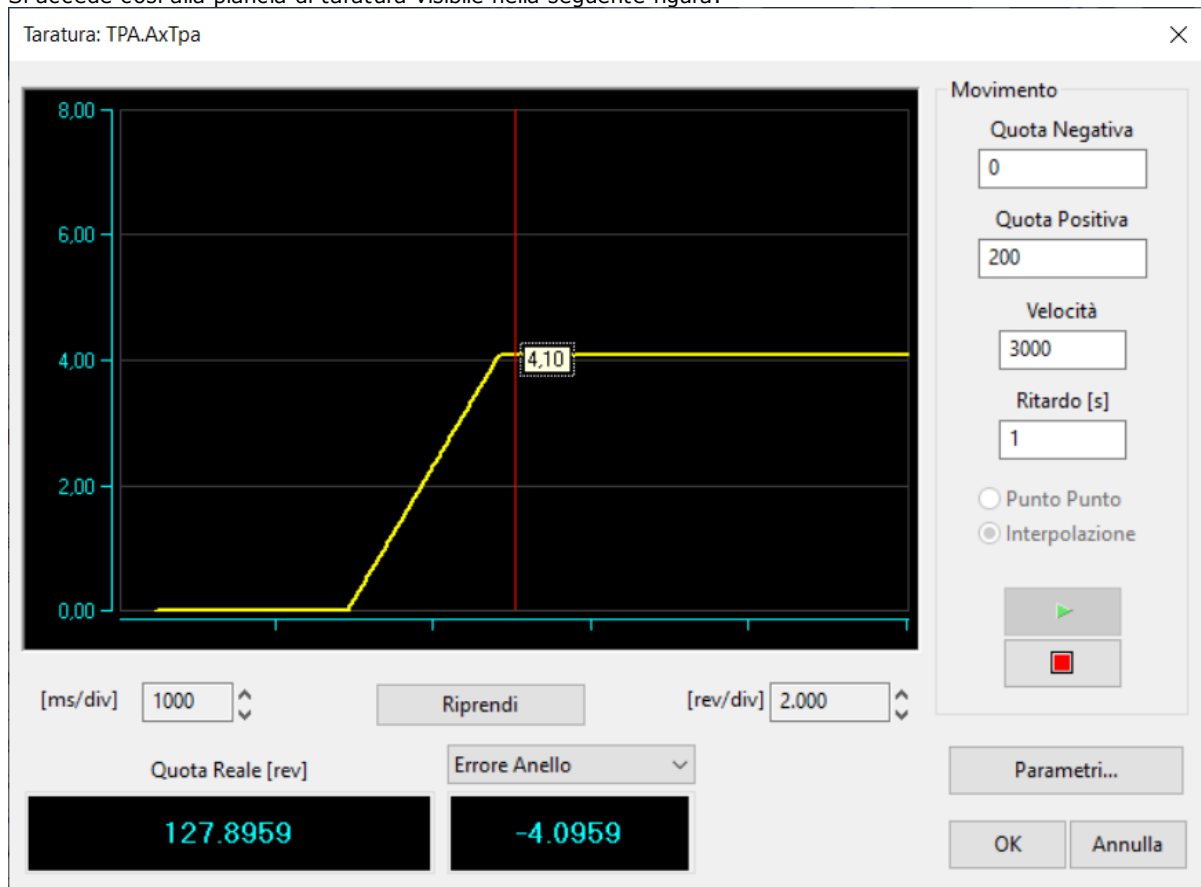
5.6 Correttori di linearità

In diagnostica si possono visualizzare e modificare i correttori di linearità dell'asse selezionato, aprendo il menu contestuale e selezionando la voce **Correttori di linearità**. La voce è visibile solo se per l'asse sono stati definiti in configurazione dei correttori di linearità. In alternativa al menu si può usare la combinazione di tasti **[Ctrl+Shift+Enter]**.

5.7 Plancia di taratura assi

La plancia di taratura assi è uno strumento che permette di modificare i parametri di configurazione di un asse e contemporaneamente di movimentare l'asse e vederne visualizzato l'errore d'anello su un oscilloscopio virtuale. Per accedere alla plancia di taratura assi si deve avere un [livello di accesso](#) al sistema maggiore o uguale a "Costruttore". Si accede alla plancia di taratura dalla modalità diagnostica o manuale effettuando un doppio click sull'asse che si vuole tarare mentre si tiene premuto il tasto **[shift]** o da menu contestuale selezionando la voce "Taratura".

Si accede così alla plancia di taratura visibile nella seguente figura:



Per verificare il comportamento dell'asse al variare dei parametri lo si fa muovere in modo continuo tra due quote limite dette **Quota Positiva** e **Quota Negativa**. Oltre a questi parametri andrà impostata la **Velocità** di

movimentazione dell'asse. Nelle prime fasi della taratura si raccomanda di utilizzare un valore basso di velocità. Si può inoltre impostare un **Ritardo** da applicare tra un movimento e il successivo.

Nella finestra dell'oscilloscopio verrà visualizzato il tracciato dell'errore d'anello o di una delle altre grandezze dell'asse. È possibile, analogamente agli oscilloscopi da banco, scalare opportunamente il grafico in modo da adattarlo alla dimensione della finestra oppure per evidenziarne i dettagli. Tramite mouse e tasti di controllo è possibile rivedere l'ultimo minuto di taratura, visualizzare uno o due cursori per fare misure e verifiche sul dato campionato, ingrandire un'area del grafico per analizzare particolari del dato campionato, variare l'offset e la scala sia nelle ascisse sia nelle ordinate. Si può inoltre bloccare, premendo il pulsante **Ferma**, lo scorrimento del grafico per osservarlo in tranquillità senza dover fermare l'asse. Oltre al grafico vengono visualizzate in due caselle la quota reale (a sinistra) e la grandezza graficata (a destra). Questa può essere impostata con il combo box situato sopra la casella di visualizzazione.

Per modificare i parametri dell'asse si deve premere il pulsante **[Parametri...]**, questo attiva una finestra ove è possibile modificare al volo la maggior parte dei parametri dell'asse. Una volta apportata una modifica ad uno o più parametri sarà possibile attuarla premendo il pulsante **[Applica]**. Questo pulsante permette di vedere immediatamente l'effetto della modifica nel comportamento dinamico dell'asse. Se si preme **[Ok]** le modifiche rimangono in presa, mentre se si preme **[Annulla]** vengono ripristinati i valori presenti prima dell'apertura della finestra **[Parametri...]**

I parametri su cui tipicamente si dovrà operare sono:

- Coefficiente **Proporzionale**
- Coefficiente **Integrativo**
- Coefficiente **Derivativo**
- **Feed Forward**: percentuale della velocità corrente fornito direttamente all'azionamento (indipendentemente dall'errore d'anello)
- **Feed Forward di Accelerazione**: percentuale del riferimento di velocità fornito direttamente all'azionamento durante le fasi di accelerazione e decelerazione dell'asse (in aggiunta al feed forward)
- **Accelerazione**: durata della rampa di accelerazione
- **Decelerazione**: durata della rampa di decelerazione

Il cursore nell'area grafica

Il cursore è uno strumento che permette di misurare e visualizzare alcuni dati della traccia. Consiste in una linea verticale del colore associato alla traccia selezionata. Si può spostare all'interno del grafico con i tasti cursore o con il mouse. In un tooltip legato alla linea verticale si possono visualizzare valori diversi, selezionabili da un menu popup, richiamabile cliccando con il tasto destro del mouse in prossimità del cursore.

Nel menu popup si possono scegliere le opzioni:

- **Canale**: presenta la lista delle tracce visualizzate nel grafico ed evidenzia con un segno di spunta quella a cui si riferisce il cursore. È, inoltre, possibile scegliere una nuova traccia da associare al cursore.
- **Stile**: presenta una lista di dati che possono essere visualizzati nel rettangolo di tooltip.
- **Valore X-Y**: visualizza il valore sull'asse x e sull'asse y del punto della traccia in cui si trova il cursore.
- **Valore X**: visualizza il valore sull'asse x del punto della traccia in cui si trova il cursore.
- **Valore Y**: visualizza il valore sull'asse y del punto della traccia in cui si trova il cursore.
- **Periodo**: visualizza il valore della distanza tra due cursori visualizzati, quando si effettua la misurazione lungo l'asse X.
- **Picco-picco**: visualizza il valore della distanza tra due cursori visualizzati, quando si effettua la misurazione tra due punti lungo l'asse Y.
- **Frequenza**: visualizza l'inverso della distanza lungo l'asse X.
- **Opzioni**: configura il modo di visualizzazione del cursore e del tooltip associato
- **Usa il colore del canale**: se abilitato, il cursore è disegnato del colore del dato campionato a cui si riferisce, altrimenti la scelta del colore con cui è visualizzato il cursore è casuale.
- **Nascondi il suggerimento al rilascio**: se abilitato, il tooltip viene visualizzato solo fino a che è mantenuto premuto il pulsante sinistro del mouse, poi viene nascosto.
- **Inverti l'allineamento**: il tooltip viene normalmente visualizzato alla destra del cursore. Se l'opzione è abilitata, il cursore viene visualizzato alla sinistra del cursore.
- **Posizionamento al campione**: se abilitato, posiziona il cursore solo sui valori campionati.

Taratura dell'asse

La taratura di un asse è un'operazione delicata che va eseguita con la massima attenzione e cautela.

Tramite la voce CalibSampleTime nella sezione [Albatros] in Tpa.ini è possibile modificare il periodo di campionamento dei dati di un asse per la finestra di Taratura. Il valore è in millisecondi e non può essere inferiore alla frequenza di controllo assi né maggiore di 100.

Prima di effettuare la taratura dell'asse dalla plancia è necessario impostare tutti i parametri in configurazione e dovrà essere impostato il fondo scala della velocità dell'azionamento. Per gli assi analogici il valore di tensione, a cui in Albatros corrisponde la velocità massima, è 9 Volt.

Per evitare di danneggiare la macchina con l'utilizzo di parametri errati conviene impostare una velocità bassa, ad esempio pari al 10% della velocità massima prevista per l'asse. In questo modo anche se si imposta un guadagno eccessivo non si otterranno reazioni troppo violente dell'asse.

In genere si effettua prima la taratura per i movimenti punto-punto ed in seguito la taratura per i movimenti interpolati.

La prima operazione, se non è già stato fatto in configurazione, è l'impostazione dei tempi di accelerazione e decelerazione. Maggiori sono questi tempi, minore sarà l'accelerazione cui è sottoposto l'asse.

Si passa poi ad impostare un guadagno minimo che consenta la movimentazione dell'asse. Questo serve per verificare la corretta taratura dell'azionamento. Albatros infatti è configurato per fornire un riferimento di 9 Volt in corrispondenza della massima velocità impostata in configurazione dell'asse. Ad esempio, se si muove l'asse ad una velocità pari al 10% di quella massima e se l'azionamento è tarato correttamente, si leggerà una tensione di riferimento pari al 10% di quella massima cioè 0.9 Volt. Se non si ottiene questa tensione di riferimento è necessario modificare il fondo scala dell'azionamento.

Una volta tarato l'azionamento si inizia ad aumentare poco per volta e sempre con la massima cautela, il guadagno dell'anello di posizione. Di volta in volta si deve verificare che non si manifestino condizioni di sovraelongazione o di instabilità. La velocità in questa fase dovrà sempre essere mantenuta minore o uguale al 10% della velocità massima. È sempre opportuno in questa fase analizzare in modo approfondito il profilo di velocità ottenuto con l'oscilloscopio virtuale, ingrandendone quanto possibile l'immagine in modo da evidenziarne i dettagli.

Quando l'asse raggiunge un comportamento allo stesso tempo pronto e stabile, si potrà aumentare, poco per volta, la velocità, verificando il comportamento dell'asse di volta in volta ed adattando il guadagno qualora questo non fosse soddisfacente. Guadagno e velocità non dovranno mai essere aumentati bruscamente. Condizioni di taratura apparentemente stabili alle basse velocità possono non esserlo a velocità maggiori.

Una volta determinato il valore ottimale del guadagno si potrà, se necessario, aumentare progressivamente i coefficienti integrativo e derivativo ed in seguito il feed forward per diminuire l'errore d'anello e portarlo entro valori accettabili dal punto di vista della precisione dell'asse. Il feed forward permette di abbattere quasi totalmente l'errore d'anello durante la fase di regime del movimento ma non durante le fasi di accelerazione e decelerazione. Per diminuire ulteriormente l'errore d'anello in queste fasi si potrà aumentare il feed forward di accelerazione. In genere valori anche molto bassi di questo parametro sono sufficienti ad ottenere un buon risultato.

Per quanto riguarda la taratura di un asse per i movimenti interpolati si possono utilizzare i valori già determinati per i movimenti punto-punto, tuttavia si deve tener conto della presenza degli altri assi della macchina. In particolare per ottenere una buona precisione durante i movimenti interpolati è necessario bilanciare gli errori d'anello degli assi. Pertanto si dovrà determinare l'asse con il maggior errore d'anello (a pari velocità) e "peggiore" (limitatamente ai parametri di interpolazione) la taratura degli altri in modo che gli errori d'anello siano identici.

6 Errori e Segnalazioni

6.1 Introduzione

Albatros gestisce eventi di errore e segnalazioni

Errori di sistema

Si tratta degli errori che il sistema Albatros è in grado di rilevare automaticamente, sia durante le fasi di esecuzione dei programmi, che durante le operazioni di manutenzione e diagnostica dell'impianto.

Questi errori sono di varia natura e spaziano da problemi sulla gestione di un asse a problemi che si possono verificare durante l'esecuzione del programma.

Gli errori di sistema possono essere gestiti all'interno dei programmi di lavoro per mezzo dell'istruzione [ONERRSYS](#), in caso contrario al loro verificarsi viene terminata l'esecuzione dei programmi sul modulo in cui l'errore si è verificato. La spiegazione di ogni errore di sistema è descritta nelle pagine successive di questo manuale.

Errori di ciclo

Si tratta di errori che si verificano durante l'esecuzione del programma ma che consentono generalmente la sua continuazione previa rimozione dell'errore stesso. Si possono generare errori di ciclo tramite l'istruzione GPL [ERROR](#).

Messaggi

Si tratta di messaggi di avvertimento che vengono prodotti in situazioni particolari dell'esecuzione del programma, o segnalazioni di richiesta di intervento da parte dell'operatore, ma che non arrestano l'esecuzione del programma stesso. Si possono generare messaggi tramite l'istruzione GPL [MESSAGE](#).

Segnalazioni

Si tratta di segnalazioni generiche non legate alla ciclica di macchina e che vengono scritte da Albatros. Non vengono visualizzate in Albatros. La descrizione di ogni segnalazione è presente nelle pagine successive di questo manuale.

La Barra degli errori

Nella barra degli errori viene visualizzato l'ultimo errore di sistema verificatosi in ordine di tempo insieme all'ultimo errore di ciclo e all'ultimo messaggio.

Gli **errori di sistema** sono contraddistinti dal colore rosso.

Gli **errori di ciclo** sono contraddistinti dal colore giallo. Si tratta di errori che si verificano durante l'esecuzione del programma, ma che consentono generalmente la sua continuazione previa rimozione dell'errore stesso.

I **messaggi** sono contraddistinti dal colore verde.

Fresatrice: Modulo Inizializzato	00001
Fresatrice: L'azzeramento degli assi non è ancora stato eseguito	00001
Fresatrice: Inizializzazione in corso....	00001

Barra degli errori

Gli errori, che si sono verificati a partire dall'accensione del sistema, sono visualizzabili in una finestra che può essere aperta da un doppio click del mouse sulla *Barra degli errori* oppure dal menu **Visualizza**. In questa finestra vengono inoltre visualizzate delle informazioni aggiuntive sugli errori di sistema.

La finestra è divisa nelle seguenti aree. Nella **parte alta** sono visualizzate le seguenti informazioni:

- Tipo: identifica il tipo di errore (errore di sistema, errore di ciclo e messaggio).
- Ora&Data: sono l'ora e la data in cui l'errore si è verificato.
- Descrizione: è la descrizione dell'errore.
- Codice: è il numero del messaggio di errore.
- Task: è il nome del task che ha generato l'errore (non presente nella Barra degli errori).

Facendo *doppio click* su una di queste colonne, le informazioni vengono ordinate in base al contenuto della colonna.

Nella **parte sottostante** troviamo le caselle:

- Errori di ciclo: se attivata vengono mostrati anche questi errori.
- Messaggi: se attivata vengono mostrati anche i messaggi.
- Tutti: se attivata vengono mostrati i messaggi di tutti i moduli del sistema, relativi al tipo di informazioni che si sta visualizzando.

- Casella nome modulo: mostra il nome del modulo di cui si stanno effettuando le visualizzazioni, permettendo inoltre di selezionare, nel caso di sistema con più moduli, il modulo di cui si vogliono visualizzare le informazioni.

Troviamo infine i **pulsanti** di comando che sono:

- **[Elimina tutti]** cancella tutte le informazioni visualizzate dalla memoria, ma non le rimuove dall'archivio.
- **[Elimina]** consente di cancellare l'informazione corrente dalla memoria, senza tuttavia rimuoverla dall'archivio.
- **[OK]** consente di chiudere la finestra.

Memorizzazione degli errori e delle segnalazioni in un file di report

Tutti gli errori vengono memorizzati in un file per una ricostruzione storica degli stessi. Si tratta di un file di testo in formato TSV. Il nome del file è MONTH (numero mese in corso).TER. Nella suite di Albatros viene fornito un programma ViewRER che carica e visualizza i file .TER.

6.2 Errori di sistema

6.2.1 Errori generati dalla gestione assi

1 NomeAsse: collegamento encoder errato

Causa:

Ad asse fermo si è verificata una differenza tra la quota teorica e la quota reale dell'asse, maggiore di 1024 passi di encoder.

Tipicamente questo avviene, durante la messa in servizio dell'asse, quando sono invertite tra loro le fasi dell'encoder. Durante il normale funzionamento si verifica quando ad azionamento spento viene movimentato manualmente l'asse senza averlo prima messo in FREE oppure quando l'asse, a causa di una taratura scorretta, è soggetto ad una eccessiva sovraelongazione nella fase di arrivo in quota (overshoot).

Dopo questo errore viene azzerato il segnale di riferimento e l'asse viene posto in stato FREE.

Soluzione:

In fase di messa in servizio dell'asse controllare il collegamento delle fasi encoder dell'asse interessato (eventualmente attivare l'opzione inversione fasi encoder nella configurazione dell'asse). Controllare la taratura asse con l'apposita modalità della Diagnostica.

2 NomeAsse: movimento non finito

Causa:

Alla fine dello spostamento, dopo 5 secondi dal termine del movimento teorico, la differenza tra la quota teorica e la quota reale dell'asse è risultata maggiore della finestra indicata in Configurazione. Può semplicemente dipendere dal fatto che l'azionamento è spento o non abilitato oppure può essere dovuto ad una cattiva regolazione dell'offset dell'azionamento. Può comunque dipendere anche da giochi meccanici sull'asse o a un guadagno dell'anello di posizione dell'asse eccessivamente basso.

Soluzione:

Verificare che l'azionamento sia acceso ed abilitato. Verificare la taratura dell'asse e regolare l'offset dell'azionamento dell'asse interessato.

3 NomeAsse: servoerror

Causa:

Durante qualsiasi tipo di movimento, la differenza tra la quota teorica e la quota reale dell'asse è diventata maggiore dell'errore massimo indicato in Configurazione o di quello impostato con l'istruzione [SETMAXER](#).

Normalmente è dovuto ad una cattiva regolazione del guadagno dell'anello di posizione o del fondo scala di velocità dell'azionamento oppure da una eccessiva inerzia dell'asse.

Soluzione:

Verificare la regolazione del guadagno e il fondo scala di velocità dell'azionamento.

Verificare il corretto funzionamento dell'encoder e del gruppo motore/azionamento.

Verificare inoltre eventuali blocchi meccanici.

4 NomeAsse: oltre limite positivo

Causa:

La quota teorica dell'asse ha superato la quota limite positiva indicata in Configurazione o impostata con l'istruzione [SETLIMPOS](#).

Soluzione:

Correggere nel programma la quota che eccede la quota limite positiva o ridefinire i limiti di quota dell'asse.

5 NomeAsse: oltre limite negativo**Causa:**

La quota teorica dell'asse ha superato la quota limite negativa indicata in Configurazione o impostata con l'istruzione [SETLIMNEG](#).

Soluzione:

Correggere nel programma la quota che eccede la quota limite negativa o ridefinire i limiti di quota dell'asse.

10 NomeAsse: l'esecuzione Real-Time è più veloce della costruzione del profilo**Causa:**

L'esecuzione real-time del profilo di moto è più rapida della generazione GPL del profilo stesso. Il lookahead viene svuotato più velocemente di quanto venga riempito. L'errore può essere dovuto a due cause in genere concomitanti:

- la velocità di interpolazione è troppo elevata rispetto alle dimensioni dei tratti da percorrere
- i tratti da percorrere sono troppo brevi.

Soluzione:

Verificare che la velocità di interpolazione impostata non sia troppo elevata rispetto alle dimensioni dei tratti da percorrere e verificare che i tratti di interpolazione da percorrere non siano troppo brevi.

6.2.2 Errori generati dalla gestione I/O remoti**2049 Ricevitore numero: configurazione errata****Causa:**

Il ricevitore remoto ha ricevuto una configurazione diversa da quella rilevata sul campo. Questo può accadere se il remoto non è uguale a quello scelto nella configurazione hardware di Albatros. Ad esempio, il ricevitore remoto è un Albre16 e in Albatros è stato configurato un remoto Albre24 (GreenBus v3.0), oppure ancora un TRS-IO con un numero sbagliato di espansioni TRS-IO-E (GreenBus v4.0).

Soluzione:

Verificare la Configurazione Hardware.

2050 Ricevitore numero: scollegato**Causa:**

Il ricevitore remoto non risponde ai comandi del trasmettitore.

Soluzione:

Verificare l'alimentazione del ricevitore e il collegamento seriale.

2051 Ricevitore numero: ricollegato**Causa:**

Il collegamento tra trasmettitore e ricevitore è stato ripristinato.

2052 Ricevitore numero: errore di riletture uscita non collegata numero NumeroUscita**Causa:**

L'uscita digitale indicata è in protezione o in cortocircuito, comunque si trova in uno stato diverso da quello previsto dal controllo. L'uscita non è associata ad alcun dispositivo logico in Configurazione Virtuale-Fisico, pertanto è sintomo di un'incongruenza tra Configurazione e l'effettivo cablaggio della macchina.

Soluzione:

Verificare la Configurazione Virtuale-Fisico. Rimuovere il cortocircuito o verificare che il carico applicato non superi i limiti massimi (consultare la documentazione tecnica).

2054 Ricevitore numero: tipo errato**Causa:**

Durante la fase di inizializzazione dei remoti è stato riscontrato, ad un certo indirizzo, un ricevitore diverso da quello specificato in configurazione.

Soluzione:

Controllare che la Configurazione Hardware corrisponda alle impostazioni dei moduli remoti.

2055 Ricevitore numero: inizializzato**Causa:**

Il ricevitore si è collegato al trasmettitore dopo uno scollegamento dovuto a mancanza di alimentazione.

2056 Ricevitore numero: errore alimentazione +24 Vcc**Causa:**

L'alimentazione di campo (+24 Vcc) di un modulo di I/O remoto non è attiva o non funziona correttamente.

Soluzione:

Verificare il funzionamento dell'alimentazione +24 Vcc.

2057 Errore alimentazione GreenBUS**Causa:**

L'alimentazione del bus di campo che connette i moduli di I/O remoto con il controllo non funziona correttamente. Questa alimentazione deve avere un valore nominale di +12 Vcc ed è fornita dal controllo.

Soluzione:

Verificare la presenza dell'alimentazione del GreenBus, verificare i cavi GreenBus. Spegner e riaccendere. Eventualmente sostituire la scheda di controllo.

2058 Ricevitore numero: errore in rilettura TipoDispositivo NomeDispositivo**Causa:**

Lo stato dell'uscita specificata non corrisponde a quello impostato. Ciò può dipendere dalla presenza di un cortocircuito, dall'intervento della protezione (carico eccessivo) o più semplicemente dalla mancanza dell'alimentazione. L'uscita specificata può essere un'uscita digitale, un'uscita analogica, un'uscita di controllo asse. Il tipo di uscita viene specificato nella visualizzazione dell'errore.

Soluzione:

Nel caso si tratti di una uscita digitale verificare l'alimentazione +24V (lato campo), rimuovere l'eventuale situazione di cortocircuito o di eccessivo assorbimento dell'uscita (consultare la documentazione tecnica). Nel caso si tratti di una uscita analogica o di una uscita di controllo assi verificare la presenza e il valore della tensione impostata sull'uscita (tester o oscilloscopio), rimuovere l'eventuale situazione di cortocircuito o di eccessivo assorbimento dell'uscita (consultare la documentazione tecnica).

2059 Fallito il test della dual port memory del trasmettitore**Causa:**

Si è verificato un errore durante i test eseguiti in fase di inizializzazione della scheda assi. In particolare, è fallita l'inizializzazione del trasmettitore GreenBus (microcontrollore i296). Può dipendere da un'errata configurazione degli indirizzi di I/O e IRQ della scheda o da un conflitto con altre periferiche presenti sul sistema. Può, infine, essere la conseguenza di un danneggiamento della scheda assi.

Soluzione:

Verificare la configurazione della scheda, verificare che non ci siano conflitti con altre periferiche. Se si utilizza un modulo remoto, ritrasmettere il firmware al modulo. I tecnici qualificati possono effettuare un test Hardware della dual port memory del microcontrollore i296. Se il problema si manifesta ancora contattare il costruttore.

2060 Errore nell'inizializzazione del trasmettitore**Causa:**

Si è verificato un errore durante i test eseguiti in fase di inizializzazione della scheda assi. In particolare è fallito l'invio del firmware al trasmettitore GreenBus (microcontrollore i296). Può dipendere da un'errata configurazione degli indirizzi di I/O e IRQ della scheda o da un conflitto con altre periferiche presenti sul sistema. Può infine essere la conseguenza di un danneggiamento della scheda assi.

Soluzione:

Verificare la configurazione della scheda, verificare che non ci siano conflitti con altre periferiche. Se si utilizza un modulo remoto ritrasmettere il firmware al modulo. I tecnici qualificati possono effettuare un test Hardware della dual port memory del microcontrollore i296. Se il problema si manifesta ancora contattare il costruttore.

2061 Errore durante la trasmissione del firmware al trasmettitore

Causa:

Si è verificato un errore durante i test eseguiti in fase di inizializzazione della scheda assi. In particolare è fallito l'invio, al trasmettitore GreenBus (microcontrollore i296), della configurazione dei moduli di I/O remoto.

Soluzione:

Verificare la configurazione hardware, se si utilizza un modulo remoto ritrasmettere il firmware al modulo. I tecnici qualificati possono effettuare un test Hardware della RAM del microcontrollore i296. Se il problema si manifesta ancora contattare il costruttore.

2062 Errore durante la trasmissione della configurazione al trasmettitore

Causa:

Si è verificato un errore durante i test eseguiti in fase di inizializzazione della scheda assi. In particolare è fallita l'inizializzazione dei moduli di I/O remoto .

Soluzione:

Verificare la configurazione hardware, se si utilizza un modulo remoto ritrasmettere il firmware al modulo. I tecnici qualificati possono effettuare un test Hardware della RAM del microcontrollore i296. Se il problema si manifesta ancora contattare il costruttore.

2063 Errore durante la trasmissione della configurazione al ricevitore

Causa:

Si è verificato un errore durante l'inizializzazione di un modulo remoto.

Soluzione:

Verificare la configurazione hardware. I tecnici qualificati possono effettuare un test Hardware del modulo remoto. Se il problema si manifesta ancora contattare l'assistenza tecnica.

2064 Ricevitore numero: versione firmware non compatibile

Causa:

Il ricevitore remoto è dotato di una versione di firmware non compatibile con il firmware del controllo.

Soluzione:

Verificare l'installazione del controllo. Se il problema persiste contattare l'assistenza tecnica

2065 Ricevitore numero: errore in una comunicazione asincrona

Causa:

Si è verificato un errore o una mancata risposta durante la comunicazione di un comando verso un remoto (GreenBus v4.0).

Soluzione:

Verificare i collegamenti e l'alimentazione del GreenBus. Se il problema persiste contattare l'assistenza tecnica.

2066 Ricevitore numero: errore generico

Causa:

Si è verificato un errore generico durante la comunicazione di un evento o di un allarme da remoto (GreenBus v4.0)

Soluzione:

Verificare i collegamenti e l'alimentazione del GreenBus. Se il problema persiste contattare l'assistenza tecnica.

2067 Ricevitore numero: errore durante la trasmissione della configurazione

Causa:

Si è verificato un errore di comunicazione nella trasmissione dei dati di configurazione ad un remoto (GreenBus v4.0).

Soluzione:

Verificare i collegamenti e l'alimentazione del GreenBus. Spegner e riaccendere. Se il problema persiste contattare l'assistenza tecnica.

2068 Ricevitore numero: errore interno n. numeroerrore

Causa:

Si è verificato un errore interno sul remoto indicato.

Soluzione:

Contattare il costruttore.

2069 Ricevitore numero: errore alimentazione +24 Vcc banco numero

Causa:

L'alimentazione di campo (+24 Vcc) di un gruppo di output collegato allo stesso morsetto di alimentazione non è attiva o non funziona correttamente.

Soluzione:

Verificare il funzionamento dell'alimentazione +24 Vcc.

6.2.3 Errori generati dalla gestione MECHATROLINK-II

2308 Scheda NumeroScheda: inizializzazione fallita dovuta a un'impostazione non corretta di un parametro di configurazione

Causa:

In Configurazione virtuale fisico non è stato collegato alcun asse (dispositivo logico) ad una scheda con bus MECHATROLINK-II (dispositivo fisico).

Soluzione:

Verificare i collegamenti in Configurazione Virtuale-fisico.

2341 Scheda NumeroScheda: il numero di servoazionamenti supera il massimo consentito

Causa:

Ad una scheda con bus MECHATROLINK-II è stato collegato un numero di servoazionamenti elevato rispetto alla configurazione impostata.

Soluzione:

Verificare in configurazione di sistema il valore di Frequenza Controllo Assi.

Nella tabella sono riportati i valori corretti da impostare a seconda del numero di servoazionamenti gestiti dalla scheda.

Scheda	Frequenza Controllo Assi (Hz)	Numero massimo servoazionamenti
AlbMech	1000	8
AlbMech	<=500	16
DualMech Mono	1000	8
DualMech Mono	500	20
DualMech Mono	250	30
DualMech	1000	16
DualMech	500	40
DualMech	250	60

2342 Scheda NumeroScheda: l'indirizzo hardware del servoazionamento Servo supera il valore massimo consentito

Causa:

In una scheda con bus MECHATROLINK-II un asse (dispositivo logico) è stato collegato ad un indirizzo hardware (dispositivo fisico) superiore al numero di servoazionamenti gestibili dalla scheda.

Soluzione:

Verificare in configurazione di sistema il valore di Frequenza Controllo Assi. Nella tabella sono riportati i valori corretti da impostare a seconda del numero di servoazionamenti gestiti dalla scheda.

Scheda	Frequenza Controllo Assi (Hz)	Numero massimo servoazionamenti
AlbMech	1000	8
AlbMech	<=500	16
DualMech Mono	1000	8
DualMech Mono	500	20
DualMech Mono	250	30
DualMech	1000	16
DualMech	500	40
DualMech	250	60

Verificare in Configurazione Virtuale Fisico il collegamento tra dispositivo logico e dispositivo fisico. Ad esempio se il numero massimo di servoazionamenti è 8, allora il collegamento tra dispositivo logico e dispositivo fisico deve essere entro i primi 8 assi (da Ax1 a Ax8).

2349 Scheda NumeroScheda: il servoazionamento Servo non è collegato**Causa:**

La connessione fisica al servoazionamento della scheda indicata è interrotta

Soluzione:

Controllare i cablaggi del bus MECHATROLINK-II e il servoazionamento.

6.2.4 Errori generati dalla gestione CanBUS**2761 Nodo numero: scollegato****Causa:**

Il nodo CAN indicato risulta attualmente non collegato al bus da campo che fa riferimento alla scheda indicata nonostante sia presente nella configurazione.

2762 Nodo numero: ricollegato**Causa:**

Il nodo CAN indicato risulta appena ricollegato al bus da campo che fa riferimento alla scheda indicata.

2763 Errore di mancata trasmissione**Causa:**

Errore interno alla scheda indicata. La trasmissione dei dati al nodo CAN indicato è fallita.

Soluzione:

Contattare il costruttore.

2764 Nodo numero: errore di mancata ricezione**Causa:**

La ricezione dei dati attesi dal nodo CAN indicato è fallita.

Soluzione:

Verificare la connessione e l'alimentazione del dispositivo CAN indicato. Verificare il cablaggio dell'intera linea CAN. Verificare la connessione della linea al controllo numerico. Verificare la coerenza fra le impostazioni di protocollo del dispositivo CAN indicato rispetto alle impostazioni del trasmettitore presente sul controllo numerico (baud rate, indirizzo, impostazioni specifiche del protocollo adottato).

2765 Nodo numero: inizializzato**Causa:**

Il nodo CAN indicato è stato collegato al bus da campo. Di seguito è stato inizializzato correttamente.

2766 Condizione di guasto sull'interfaccia CAN**Causa:**

Viene segnalato un guasto all'alimentazione interna del dispositivo di interfaccia CAN.

Soluzione:

Contattare il costruttore.

2767 Errore perdita di stato CANopen**Causa:**

A causa di un problema grave il trasmettitore CAN non è più in stato operativo (Operational).

Soluzione:

Contattare il costruttore.

2768 Nodo numero: errore mancata ricezione PDO**Causa:**

Il controllo numerico non ha ricevuto il PDO, che attendeva dal nodo CAN.

Soluzione:

Verificare:

- l'alimentazione del nodo.
- che il nodo non sia rimasto in modalità pre-Operational.
- verificare i dati del PDO e del bus CAN, configurati in Albatros.

2769 Nodo numero: errore ricezione di un nodo non configurato**Causa:**

È stata rilevata la presenza di un nodo sulla rete CAN, che non è stato dichiarato nella configurazione hardware di Albatros.

Soluzione:

Verificare l'indirizzo hardware del nodo e l'indirizzo del nodo impostato in configurazione hardware. Verificare che il nodo sia stato dichiarato in configurazione hardware, altrimenti è necessario aggiungerlo.

2770 Nodo numero: configurazione errata**Causa:**

La descrizione dei dati di RPDO e TPDO è errata.

Soluzione:

Correggere la descrizione dei dati dei pdo di trasmissione e di ricezione in configurazione hardware.

2771 Nodo numero: errore sulla comunicazione SDO**Causa:**

Il nodo Can indicato non ha risposto in una comunicazione asincrona (SDO).

Soluzione:

Verificare lo stato di collegamento del nodo. Se il problema persiste contattare il costruttore.

2772 Timeout sul ciclo CAN di interrogazione dei nodi**Causa:**

Si è verificato un errore di timeout sul ciclo CAN di interrogazione dei nodi

Soluzione:

Modificare in configurazione hardware il valore del tempo di campionamento impostato.

3073 Nodo numero: errore Emergency n. NumeroErrore

Causa:

Il dispositivo CANopen ha riscontrato una situazione di errore del nodo, specificata dal codice visualizzato. Il codice di errore è un numero esadecimale. Si tratta di situazioni di errore relative al singolo nodo e conformi allo standard CiA DS301-protocollo EMERGENCY.

Soluzione:

Fare riferimento alla documentazione del nodo.

3074 Nodo numero: errore CAN generico n. NumeroErrore

Causa:

Si è verificato un errore interno sul nodo indicato. Il codice di errore è un numero esadecimale.

Soluzione:

Contattare il costruttore.

3088 Scheda CAN numero: nodo NumeroNodo: errore sulla comunicazione SDO n. NumeroErrore - descrizione

Causa:

In una istruzione READDICTIONARY o WRITEDICTIONARY una o più richieste di lettura/scrittura SDO sono fallite. Il fallimento delle istruzioni può essere causato ad esempio dalla richiesta di lettura di un oggetto CANOpen non implementato nel dispositivo al quale si sta facendo riferimento; oppure può essere legato alla scrittura, in un registro CANOpen, di un dato non compatibile con il tipo dell'oggetto (ad esempio, tentativo di scrittura di una stringa in un oggetto il cui tipo è integer). Il codice di errore fornito è conforme alle specifiche DS402 ed oltre al codice numerico è fornita anche la descrizione testuale. Il codice di errore è un numero esadecimale.

Soluzione:

Controllare la correttezza dei parametri di BaudRate, Tempo di campionamento, ecc. impostati in configurazione hardware ed i parametri di eventuali istruzioni [READDICTIONARY](#) e/o [WRITEDICTIONARY](#) presenti nel codice GPL.

6.2.5 Errori generati dalla gestione bus EtherCAT

3329 Errore nell'inizializzazione del socket di comunicazione

Causa:

Il firmware non è riuscito a comunicare con la scheda di rete.

Soluzione:

Se la scheda è stata configurata in un sistema RTX, verificare che i file .ini presenti nella sottocartella FW di Albatros siano scritti correttamente. Per la verifica della sintassi dei file consultare il manuale di installazione di RTX in Albatros (InstallationRTXGuide.pdf).

3330 Errore durante la scansione della rete EtherCAT

Causa:

Durante la scansione preliminare della rete EtherCAT, il master non ha ricevuto risposta da alcuni o da tutti gli slave configurati, oppure la configurazione non corrisponde all'effettiva rete EtherCAT presente sul campo.

Soluzione

Verificare il cablaggio tra master EtherCAT e slave.

Verificare le descrizioni dei dispositivi in configurazione hardware. Un aiuto per l'individuazione dell'errore è fornito dalla finestra di Diagnostica hardware. Qui sono visualizzati i nodi presenti e, se configurati errati, oltre al nome del dispositivo trovato, è visualizzato il nome del dispositivo atteso.

3331 Errore nella configurazione della mailbox di trasmissione

Causa:

Il nodo EtherCAT non ha risposto al comando impartito dal Master. Le cause possono essere: comunicazione assente, nodo guasto...

Soluzione:

Verificare il cablaggio e il funzionamento remoto.

3332 Errore nella configurazione della mailbox di ricezione

Causa:

Il nodo EtherCAT non ha risposto al comando impartito dal Master. Le cause possono essere: comunicazione assente, nodo guasto...

Soluzione:

Verificare il cablaggio e il funzionamento remoto.

3333 Scheda EtherCAT numero: errore nel tipo delle espansioni del nodo NumeroNodo

Causa:

Il tipo di espansioni configurate su un nodo EtherCAT in configurazione hardware non corrisponde al tipo di espansioni realmente presenti. (Ad esempio, in configurazione hardware è stato definito un TRS-CAT con una espansione TRS-IO-E, mentre nel sistema è presente un TRS-CAT con una espansione TRS-AN-E).

Soluzione:

Verificare che i dispositivi definiti in configurazione hardware corrispondano a quelli presenti.

3334 Errore in fase di configurazione dei PDO

Causa:

Il nodo EtherCAT, per il quale si è cercato di configurare i PDO, non è presente sulla rete o è guasto.

Soluzione:

Verificare che la configurazione della rete EtherCAT, descritta in configurazione di Albatros, corrisponda alla configurazione fisica della rete.

3335 Nodo NumeroNodo in allarme (NumeroErrore)

Causa:

Il nodo indicato si trova in una situazione di allarme.

Soluzione:

Verificare il codice di allarme nella seguente tabella

Codice allarme	Descrizione
0x0001	Unspecified error
0x0002	No memory
0x0011	Invalid requested state change
0x0012	Unknown requested state
0x0013	Bootstrap not supported
0x0014	No valid firmware
0x0015	Invalid mailbox configuration
0x0016	Invalid mailbox configuration
0x0017	Invalid sync manager configuration
0x0018	No valid inputs available
0x0019	No valid outputs
0x001A	Synchronization error
0x001B	Sync manager watchdog
0x001C	Invalid Sync Manager Types
0x001D	Invalid Output Configuration
0x001E	Invalid Input Configuration
0x001F	Invalid Watchdog Configuration

0x0020	Slave needs cold start
0x0021	Slave needs INIT
0x0022	Slave needs PREOP
0x0023	Slave needs SAFEOP
0x0024	Invalid input mapping
0x0025	Invalid output mapping
0x0026	Inconsistent settings
0x0027	Free-Run not supported
0x0028	Synchronization not supported
0x0029	Free-Run needs 3 buffer mode
0x002A	Background watchdog
0x002B	No valid inputs and outputs
0x002C	Fatal Sync error
0x002D	No Sync error
0x0030	Invalid DC SYNCH Configuration
0x0031	Invalid DC Latch Configuration
0x0032	PLL Error
0x0033	Invalid DC IO Error
0x0034	Invalid DC Timeout Error
0x0035	DC Invalid Sync Cycle Time
0x0036	DC Sync0 Cycle Time
0x0037	DC Sync1 Cycle Time
0x0041	MBX_AOE
0x0042	MBX_EOE
0x0043	MBX_COE
0x0044	MBX_FOE
0x0045	MBX_SOE
0x004F	MBX_VOE
0x0050	EEPROM no access
0x0051	EEPROM error
0x0060	Slave restarted locally

3336 Scheda EtherCAT numero: il numero di espansioni del nodo NumeroNodo è errato

Causa:

Il numero di espansioni configurate su un nodo EtherCAT in Albatros non corrisponde al numero di espansioni presenti. (Ad esempio, in configurazione hardware è stato definito un TRS-CAT con due espansioni TRS-IO-E, mentre sul sistema esiste una sola espansione).

Soluzione:

Verificare che i dispositivi definiti in configurazione hardware corrispondano a quelli presenti.

3337 Scheda EtherCAT: nodo NumeroNodo scollegato

Causa:

Il nodo EtherCAT non ha restituito alcuna risposta al controllo. Il nodo potrebbe essere scollegato dalla rete oppure potrebbe essere spento.

Soluzione:

Verificare il cablaggio e che il nodo sia alimentato.

3338 Scheda EtherCAT: nodo NumeroNodo ricollegato**Causa:**

Il nodo EtherCAT indicato è stato ricollegato alla rete e ha ricominciato a rispondere alle richieste del controllo.

3340 Scheda EtherCAT: il nodo NumeroNodo non ha risposto alla richiesta (Codice)**Causa:**

Il nodo EtherCAT, interrogato tramite il servizio SDO, non ha risposto alla richiesta. Inoltre, se il dispositivo ha fornito un abort code, questo viene riportato nel messaggio di errore mostrato (codice). È possibile che questo errore compaia durante l'esecuzione delle istruzioni [SETPZERO](#), [SETPFLY](#) e [FASTREAD](#) su assi EtherCAT, dato che esse usano la comunicazione SDO per configurare l'azionamento. In questo caso specifico, il messaggio d'errore, oltre agli abort code standard, può contenere i seguenti codici:

- 1: Timeout interno scaduto: il nodo non ha risposto entro 250 ms.
- 00000005: indice interno della mailbox errata.
- 00000006: dato ricevuto non conforme alla richiesta.

Soluzione:

Contattare il costruttore.

3341 Scheda EtherCAT: il nodo NumeroNodo non esiste**Causa:**

Si è cercato di far eseguire un comando ad un nodo inesistente.

Soluzione:

Verificare il numero del nodo nell'istruzione GPL che ha generato l'errore.
Collegare il nodo.

3342 Cavo scollegato**Causa:**

Il cavo EtherCAT non è collegato al controllo.

Soluzione:

Controllare che il cavo sia correttamente collegato al controllo e che non sia danneggiato.

3343 Scheda EtherCAT numero: nodo NumeroNodo non passa nello stato SAFE-OPERATIONAL (Codice)**Causa:**

Il nodo EtherCAT indicato non è passato allo stato SAFE-OPERATIONAL.

Soluzione:

Contattare il costruttore e riferire il numero di errore indicato (codice), che rappresenta il codice ALstatusCode.

3344 Scheda EtherCAT numero: nodo NumeroNodo non passa nello stato OPERATIONAL (Codice)**Causa:**

Il nodo EtherCAT indicato non è passato allo stato OPERATIONAL.

Soluzione:

Contattare il costruttore e riferire il numero di errore indicato (codice), che rappresenta il codice ALstatusCode.

3345 Scheda EtherCAT: comunicazione instabile

Causa:

La comunicazione sulla rete EtherCAT è instabile a causa di disturbi o cavi o nodi difettosi.

4400 Troppi assi attivi in FASTREAD (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Si è cercato di utilizzare una istruzione FASTREAD con un numero di assi maggiore di quello consentito. È necessario tener presente che gli assi dotati di encoder aggiuntivo equivalgono ad un asse doppio (si veda l'istruzione [SWITCHENC](#)).

Soluzione:

Ridurre il numero di assi utilizzati con encoder aggiuntivo.

6.2.6 Errori generati dall'inizializzazione

769 Configurazione software errata

Causa:

La configurazione hardware del modulo remoto non corrisponde alla configurazione software specificata in configurazione di sistema.

Soluzione:

Verificare la congruenza tra i parametri hardware del modulo remoto e i parametri software.

770 Numero di IRQ configurato errato

Causa:

Non è stato impostato correttamente l'IRQ della scheda assi nella configurazione di Modulo. Normalmente si è in presenza di un conflitto hardware con altre periferiche presenti nel sistema.

Soluzione:

Verificare nelle impostazioni del BIOS della scheda madre che l'IRQ utilizzato dalla scheda assi sia riservato "Legacy ISA". Verificare che non ci siano altre periferiche che utilizzano lo stesso IRQ assegnato alla scheda assi. Se possibile, modificare l'IRQ della periferica che entra in conflitto con la scheda assi, altrimenti modificare l'IRQ della scheda assi.

772 Errore nella lettura dell'area di memoria tamponata durante l'inizializzazione

Causa:

Si è verificato un errore durante i test eseguiti in fase di inizializzazione della scheda assi. In particolare, è fallito il test della RAM tamponata (Dallas). Può dipendere da un'errata configurazione degli indirizzi di I/O e IRQ della scheda o da un conflitto con altre periferiche presenti sul sistema. Può, infine, essere la conseguenza di un danneggiamento della scheda assi.

Soluzione:

Verificare la configurazione hardware. I tecnici qualificati possono effettuare un test Hardware della RAM del microcontrollore i296. Notare che il test Hardware della RAM comporta la cancellazione di tutti i dati in essa memorizzati. Sulla RAM tamponata vengono conservati i valori di alcuni dispositivi ad esempio i contatori, i timer e gli offset dei DAC degli assi. Prima di effettuare il test andranno salvati questi valori. Se il problema si manifesta ancora, contattare il costruttore.

773 Raggiunto il numero massimo di assi configurabili

Causa:

Si è cercato di configurare un numero di assi superiore al numero massimo consentito.

Soluzione:

Ridurre il numero di assi configurato. Contattare il costruttore per ulteriori informazioni.

774 Il Real-time assi non è entrato in esecuzione

Causa:

Il firmware di gestione degli assi è stato inizializzato, ma non funziona correttamente. Normalmente si è in presenza di un conflitto hardware con altre periferiche presenti nel sistema.

Soluzione:

Verificare che non ci siano conflitti con altre periferiche, modificare la configurazione delle periferiche che causano il conflitto o rimuoverle dal sistema.

775 Tempo insufficiente per l'esecuzione GPL

Causa:

L'esecuzione di un task real-time occupa troppo tempo ciclo. Viene generato quando un task real-time non termina entro l'inizio del real-time assi successivo (ad es. è stato creato un ciclo infinito).

Soluzione:

Modificare il codice GPL in modo da ridurre la durata del task real-time.

776 Tempo eccessivo di esecuzione del Real-Time

Causa:

L'esecuzione di un task real-time occupa troppo tempo ciclo. Il tempo di esecuzione risulta leggermente superiore al limite massimo consentito.

Soluzione:

Modificare il codice GPL in modo da ridurre la durata del task real-time.

777 Watchdog scaduto

Causa:

Il firmware è bloccato.

Soluzione:

Contattare il costruttore.

778 Il codice di Main del firmware è bloccato

Causa:

Il firmware si è bloccato per più di 5 real-time.

Soluzione:

Contattare il costruttore.

1025 Scheda NumeroScheda: non risponde al comando

Causa:

All'inizializzazione è stata rilevata la presenza di una scheda assi, ma questa non risponde correttamente ai comandi.

Soluzione:

I tecnici qualificati possono effettuare un test Hardware della scheda assi. Se il problema si manifesta ancora, contattare il costruttore.

1026 Scheda NumeroScheda: errore nell'invio del firmware alla scheda assi

Causa:

Non è stato possibile inviare il firmware alla scheda.

Soluzione:

Contattare il costruttore.

1028 Scheda NumeroScheda: firmware non presente

Causa:

I firmware presenti sulla scheda non sono corretti per il tipo di scheda rilevata.

Soluzione:

Inviare il firmware corretto.

1029 Scheda NumeroScheda: Main bloccato

Causa:

Il firmware della scheda non è entrato in esecuzione.

Soluzione:

Contattare il costruttore.

1031 Scheda NumeroScheda: errore in inizializzazione

Causa:

Si è verificato un errore durante la procedura di inizializzazione della scheda assi.

Soluzione:

Controllare e sistemare le cause degli errori di sistema avvenuti negli istanti precedenti al verificarsi dell'errore corrente. Di seguito inizializzare il sistema.

1032 Scheda NumeroScheda: fallito il test della dual port memory

Causa:

Si è verificato un errore durante i test eseguiti in fase di inizializzazione della scheda assi. In particolare, è fallita l'inizializzazione della dual port memory.

In genere, ciò è dovuto ad un conflitto hardware con altre periferiche presenti sul sistema, ma può dipendere anche da un danneggiamento della scheda.

Soluzione:

Verificare la configurazione della scheda, verificare che non ci siano conflitti con altre periferiche. Se si utilizza un modulo remoto, ritrasmettere il firmware al modulo. I tecnici qualificati possono effettuare un test Hardware della dual port memory. Se il problema si manifesta ancora, contattare il costruttore.

1033 Scheda NumeroScheda: il codice di boot del firmware non è in esecuzione

Causa:

Il firmware di avvio della scheda è stato inizializzato, ma non funziona correttamente. Normalmente si è in presenza di un conflitto hardware con altre periferiche presenti nel sistema.

Soluzione:

Verificare la configurazione della scheda, verificare che non ci siano conflitti con altre periferiche. Se si utilizza un modulo remoto ritrasmettere il firmware al modulo. I tecnici qualificati possono effettuare un test Hardware della dual port memory presente sulla scheda. Se il problema si manifesta ancora, contattare il costruttore.

1035 Scheda NumeroScheda: non presente

Causa:

Si è verificato un errore durante i test eseguiti in fase di inizializzazione della scheda assi. In particolare, non è stata rilevata la scheda.

Soluzione:

Verificare che la scheda sia presente sul sistema e che non sia danneggiata. I tecnici qualificati possono effettuare un test Hardware della scheda. Se il problema si manifesta ancora, contattare il costruttore.

1037 Scheda NumeroScheda: fallita l'apertura della dual port memory**Causa:**

Fallita l'apertura della dual port memory della scheda.

Soluzione:

I tecnici qualificati possono effettuare un test Hardware della scheda. Contattare il costruttore.

1039 Scheda NumeroScheda: watchdog scaduto**Causa:**

Il firmware della scheda assi NumeroScheda è bloccato.

Soluzione:

Contattare il costruttore.

1040 Scheda NumeroScheda: errore alimentazione +24 Vcc**Causa:**

L'alimentazione di campo (+24 Vcc) delle uscite non è presente o non funziona correttamente.

Soluzione:

Verificare il funzionamento dell'alimentazione +24 Vcc di campo.

1047 Scheda NumeroScheda: configurazione software non permessa**Causa:**

Il dispositivo ha ricevuto una configurazione che non è compatibile con l'hardware presente o abilitato. Ad esempio, viene richiesta la configurazione di un asse non abilitato o non presente sul dispositivo.

Soluzione:

Verificare la congruenza tra i parametri hardware della scheda e quelli software.

1052 Scheda NumeroScheda: è in esecuzione il codice di boot**Causa:**

La scheda è impostata in modalità provvisoria e sta eseguendo il codice di boot.

Soluzione:

Contattare il costruttore.

1053 Scheda NumeroScheda: watchdog assi scaduto**Causa:**

Si è verificato un errore grave durante l'esecuzione del firmware della scheda di controllo assi. Gli assi vengono disabilitati e l'eventuale segnale di SYSOK viene abbassato. Non ripristinare il sistema.

Soluzione:

Contattare il costruttore.

1055 Watchdog scaduto sulla scheda NumeroScheda**Causa:**

Il firmware della scheda NumeroScheda è bloccato.

Soluzione:

Contattare il costruttore.

1056 Scheda NumeroScheda: errore di alimentazione interfaccia CAN**Causa:**

Il dispositivo di trasmissione sulla linea CanBus presente sulla scheda indicata è un errore di alimentazione. Ciò può dipendere dalla presenza di un cortocircuito, da un errore di cablaggio del bus, oppure da un danneggiamento della scheda.

Soluzione:

Verificare il cablaggio dell'intera linea CAN. Verificare la connessione della linea al controllo numerico. Rimuovere la presenza dell'eventuale cortocircuito. Nel caso non si dovesse ripristinare la comunicazione, contattare il costruttore.

1057 Scheda NumeroScheda: errore interno numero NumeroErrore**Causa:**

Errore nell'hardware del nodo.

Soluzione:

Contattare il costruttore.

6.2.7 Errori generati dalla gestione della memoria**1281 Errore nell'allocazione di memoria nell'area di heap****Causa:**

La memoria RAM disponibile è inferiore a quella richiesta, per esempio, da una matrice globale.

Soluzione:

Ridurre la dimensione delle variabili globali allocate in RAM.

1286 Errore nella gestione dello heap**Causa:**

Errore nella gestione della memoria da parte del firmware.

Soluzione:

Contattare il costruttore.

1287 Troppe disallocazioni di memoria dallo heap**Causa:**

Errore nella gestione della memoria da parte del firmware.

Soluzione:

Contattare il costruttore.

1289 Errore nella creazione delle variabili globali**Causa:**

È stato definito un numero troppo elevato di [variabili globali](#) oppure sono state definite delle matrici globali di dimensioni troppo grandi.

Soluzione:

Ridurre il numero delle variabili globali o la dimensione delle matrici.

1290 Errore nella dimensione delle variabili non volatili**Causa:**

È stato definito un numero troppo elevato di variabili non volatili oppure sono state definite delle matrici non volatili di dimensioni troppo grandi.

Soluzione:

Ridurre il numero delle variabili non volatili o la dimensione delle matrici non volatili.

1291 Errore nella dimensione delle variabili sola lettura**Causa:**

È stato definito un numero troppo elevato di variabili a sola lettura oppure sono state definite delle matrici a sola lettura di dimensioni troppo grandi.

Soluzione:

Ridurre il numero delle variabili a sola lettura o la dimensione delle matrici a sola lettura.

6.2.8 Errori generati dai fault

1559 Trace di breakpoint

Causa:

Errore grave del firmware.

Soluzione:

Contattare il costruttore.

1569 Codice operativo del microprocessore non valido

Causa:

Il microprocessore ha incontrato una istruzione sconosciuta. Può dipendere sia da problemi hardware del PC che dal danneggiamento dei file contenenti il firmware di Albatros.

Soluzione:

Nel caso di modulo locale verificare che i file non siano danneggiati ed eventualmente reinstallare Albatros. Per i moduli remoti effettuare un aggiornamento del firmware. Effettuare un test dell'hardware del PC in particolare della RAM. Se il problema persiste, contattare il costruttore.

1586 Valore INTEGER diviso per zero

Causa:

Si è tentato di dividere un INTEGER per zero.

Soluzione:

Verificare nelle funzioni GPL che tutte le divisioni siano corrette.

1600 Overflow nel risultato di un'operazione in virgola mobile

Causa:

Il risultato di una operazione tra FLOAT è più grande delle capacità del destinatario:

± 3,402823E+38 per i float
± 1,79769313486231E+308 per i double.

Soluzione:

Verificare nelle funzioni GPL se i calcoli in virgola mobile sono corretti.

1601 Underflow nel risultato di un'operazione in virgola mobile

Causa:

Il risultato di una operazione tra FLOAT è più piccolo delle capacità del destinatario:

± 1,401298E-45 per i float
± 4,94065645841247E-324 per i double.

Soluzione:

Verificare nelle funzioni GPL se i calcoli float sono corretti.

1602 Argomento non valido per un'operazione in virgola mobile

Causa:

In una operazione float è stato usato un operando non di tipo float.

Soluzione:

Verificare nelle funzioni GPL se i calcoli float sono corretti.

1603 Valore in virgola mobile diviso per zero

Causa:

Si è tentato di dividere un float o un double per zero. Viene generato anche nel caso in cui si esegue un logaritmo di zero.

Soluzione:

Verificare nelle funzioni GPL se tutte le divisioni sono corrette.

1604 Risultato inesatto di un'operazione in virgola mobile

Causa:

Il risultato di una operazione tra float non è corretto.

Soluzione:

Verificare nelle funzioni GPL se i calcoli float sono corretti.

1605 È stato usato un valore in virgola mobile errato

Causa:

È stato utilizzato un valore in virgola mobile più piccolo del minimo rappresentabile:

± 1,401298E-45 per i float
± 4,94065645841247E-324 per i double.

Soluzione:

Verificare nelle funzioni GPL se i calcoli float sono corretti.

1728 Si è cercato di accedere a un indirizzo non valido

Causa:

Il programma ha eseguito un accesso a una zona di memoria non valida.

Soluzione:

Verificare la congruenza delle variabili globali/locali e, se il problema persiste, segnalare l'anomalia.

1735 Eccezione generica

Causa:

Si è verificata un'eccezione non riconosciuta.

Soluzione:

Contattare il costruttore.

1736 Dati non allineati

Causa:

Errore grave del firmware.

Soluzione:

Contattare il costruttore.

1801 Allarme temperatura

Causa:

La temperatura della CPU del controllo ha superato i limiti massimi consentiti.

Soluzione:

Verificare che non vi siano problemi di ventilazione o cause di surriscaldamento. Se il problema si manifesta ancora, contattare il costruttore.

1802 Allarme ventola

Causa:

La ventola della CPU del controllo non funziona correttamente. Il problema può causare in breve tempo il surriscaldamento della CPU.

Soluzione:

Contattare il costruttore.

1803 La frequenza della CPU è instabile

Causa:

La frequenza di lavoro della CPU non è stabile.

Soluzione:

Contattare il costruttore.

6.2.9 Errori generati da istruzioni GPL

4097 Il dispositivo TipoDispositivo NomeDispositivo non è configurato

Causa:

Un'istruzione GPL ha utilizzato un dispositivo non configurato, ovvero che non ha un collegamento Virtuale-Fisico. Può essere generato da tutte le istruzioni a cui viene passato un dispositivo per parametro.

Soluzione:

Verificare nelle configurazioni del controllo che tutti i dispositivi utilizzati dalle funzioni abbiano un collegamento Virtuale-Fisico.

Ritrasmettere successivamente le configurazioni alla scheda.

4098 La variabile globale NomeVariabile è inesistente

Causa:

Ad una istruzione GPL è stata passata come argomento una variabile globale che non è stata definita. Generalmente succede quando il controllo non è stato inizializzato correttamente.

Soluzione:

Ricompile tutto il codice GPL e inizializzare nuovamente il controllo.

4099 Funzione NomeFunzione non trovata

Causa:

È stata eseguita una chiamata a una funzione che non è presente. Può verificarsi quando, a seguito di una modifica del codice GPL, non è stata eseguita una inizializzazione del controllo.

Soluzione:

Ricompile tutto il codice GPL e inizializzare nuovamente il controllo.

4101 Gestione incongruente dell'asse NomeAsse

Causa:

È stato eseguito un passaggio di stato illegale su un asse. Per i passaggi di stato consultare la relativa documentazione. L'errore può essere generato da tutte le istruzioni che gestiscono gli assi, solitamente si verifica in questi casi:

- se si cerca di interpolare, coordinare con un asse già impegnato in un movimento punto a punto (o viceversa).
- se si esegue una istruzione Chain, SetPFly, SetPZero su un asse che è in transparent mode.
- se si cerca di interpolare, coordinare su un asse che è slave di un altro.

Soluzione:

Controllare che tutti i movimenti di assi siano terminati con una istruzione di attesa in quota, soprattutto se gli assi alternano movimenti di tipo diverso (punto punto, interpolazione, ecc.).

4105 Istruzione non eseguibile sull'asse NomeAsse

Causa:

Si è cercato di eseguire un'istruzione su un asse che non la supporta. Ad esempio, un'istruzione di interpolazione su un asse passo-passo.

Soluzione:

Correggere il codice GPL.

4106 Il remoto riferito all'asse passo-passo NomeAsse non è collegato

Causa:

Si è cercato di operare su un asse passo-passo che non è collegato al controllo.

Soluzione:

Verificare il collegamento del remoto che controlla l'asse.

4107 Istruzione SYSOK con argomenti errati

Causa:

Si è eseguita un'istruzione SYSOK con degli argomenti errati. Si verifica se una o più delle uscite digitali passate come argomenti all'istruzione non sono configurati correttamente.

Soluzione:

Verificare il codice GPL e la configurazione Virtuale-fisico.

4108 L'asse NomeAsse: quota finale oltre i limiti software

Causa:

Si è cercato di muovere un asse oltre i limiti impostati in configurazione o da codice GPL.

Soluzione:

Correggere il programma di lavorazione che ha causato l'errore. Eventualmente correggere il codice GPL o la configurazione dell'asse.

4110 Velocità errata

Causa:

Si è cercato di assegnare ad un asse una velocità nulla o negativa.

Soluzione:

Correggere il codice GPL.

4111 Accelerazione asse NomeAsse negativa

Causa:

Si è cercato di assegnare ad un asse un'accelerazione negativa.

Soluzione:

Correggere il codice GPL.

4112 Decelerazione asse NomeAsse negativa

Causa:

Si è cercato di assegnare ad un asse una decelerazione negativa.

Soluzione:

Correggere il codice GPL.

4114 Asse NomeAsse: azzeramento su ingresso veloce non effettuato

Causa:

L'azzeramento quota su ingresso veloce (azzeramento al volo) non è stato completato correttamente. Questa procedura permette di azzerare la quota di un asse in movimento, nell'istante in cui l'ingresso veloce corrispondente cambia di stato. Se l'asse termina il movimento in corso, senza che si sia verificata la commutazione dell'ingresso, viene generato l'errore di sistema. La causa può essere un'impostazione errata dei parametri di movimento dell'asse o un problema di cablaggio dell'ingresso veloce.

Soluzione:

Verificare il codice GPL che implementa l'azzeramento al volo e verificare il cablaggio dell'ingresso veloce.

4115 Asse NomeAsse: tacca di zero non trovata

Causa:

L'azzeramento quota su tacca di zero dell'encoder non è stato completato correttamente. Questa procedura permette di azzerare la quota di un asse in movimento, nell'istante in cui viene rilevata la tacca di zero dell'encoder. Se l'asse raggiunge la quota di ricerca tacca, senza che questa venga rilevata, viene generato l'errore di sistema. La causa può essere un'impostazione errata dei parametri di movimento dell'asse o un problema di cablaggio del segnale di tacca (fase C del connettore asse).

Soluzione:

Verificare il codice GPL che implementa l'azzeramento su tacca e verificare il cablaggio dell'asse.

4353 Codice operativo istruzione sconosciuta (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Nell'esecuzione di una funzione GPL è stata riscontrata una istruzione illegale. In genere, è sintomo di danneggiamento dei file che contengono il codice GPL compilato. Se si è aggiornato il software del controllo, verificare di aver ricompilato il codice GPL. La precedente versione del software del controllo potrebbe, infatti, contenere delle istruzioni non più supportate dalla nuova versione.

Soluzione:

Ricompilare tutto il codice GPL e inizializzare il controllo. Se il problema persiste, contattare il costruttore.

4354 Operazione matematica errata (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Un'istruzione GPL ha cercato di eseguire una operazione matematica sbagliata, ad esempio una divisione per zero. Oppure alcuni dati introdotti nelle istruzioni GPL sono incongruenti. Tipicamente questo errore viene generato dalle istruzioni di movimento in interpolazione, poiché è la parte del Firmware che esegue più calcoli matematici.

Soluzione:

Verificare che i parametri passati alle istruzioni di interpolazione siano corretti. Se il problema persiste, segnalare il malfunzionamento al costruttore.

4355 Indirizzo errato di matrice o vettore (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Un'istruzione GPL ha cercato di accedere a un elemento di array o di matrice che oltrepassa la dimensione massima. Per esempio, ha cercato di accedere all'elemento 10 di un array di 5 elementi. Può essere generato da tutte le istruzioni che accettano come parametro un array o una matrice.

Soluzione:

Verificare che tutti gli indici di matrice e array passati alle istruzioni siano all'interno delle dimensioni dell'array e della matrice.

4356 Istruzione RET non chiamata da CALL (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

È stata eseguita una istruzione RET senza che sullo stack fosse presente il relativo indirizzo di ritorno. La causa più comune è l'aver dichiarato una sottoprocedura prima dell'istruzione FRET di uscita da funzione, senza averla protetta con un GOTO che ne eviti l'esecuzione accidentale. Oppure si è effettuato un salto non voluto all'interno di una sottoprocedura.

Soluzione:

Controllare il flusso del programma GPL. Preferibilmente posizionare le sottoprocedure alla fine del corpo delle funzioni (dopo l'istruzione FRET).

4357 Variabile locale inesistente (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Un'istruzione GPL ha cercato di accedere a una variabile locale che non è stata allocata.

Soluzione:

Ricompile e ritrasmettere tutte le funzioni alla scheda. Se il problema persiste, segnalare il malfunzionamento.

4358 Etichetta di salto inesistente (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Un'istruzione GPL ha effettuato un salto a una etichetta di salto inesistente. Può essere generato da GOTO, CALL, FCALL e tutte le IF.

Soluzione:

Ricompile e ritrasmettere tutte le funzioni alla scheda. Se il problema persiste, segnalare il malfunzionamento.

4359 Argomento macro errato (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Ad una istruzione GPL sono stati passati degli argomenti non validi. Può essere generato da tutte le istruzioni. Tuttavia, nella stragrande maggioranza dei casi, il sistema GPL cerca di correggere automaticamente questa situazione, eseguendo delle conversioni automatiche di tipo (cast), che comportano, però, delle perdite di tempo. L'errore viene generato quando queste conversioni non sono possibili e maggiormente nei seguenti casi:

- istruzioni che operano su dispositivi specifici (SETTIMER, SETCOUNTER), a cui ne viene passato uno di tipo diverso.
- istruzioni che operano su bit, a cui viene passato un numero in virgola mobile (AND, OR, ecc.)
- istruzioni che operano su matrici o array, a cui viene passata una variabile semplice (SORT, MOVEMAT, ecc.)
- istruzioni che operano su stringhe, a cui non vengono passate stringhe.

L'errore viene generato anche quando si cerca di eseguire una istruzione su una scheda che non gestisce tale istruzione. (Ad esempio, una istruzione [SENDPDO](#) o una istruzione [RECEIVEPDO](#) su una scheda diversa da TMSCan o da TMSCan+).

Soluzione:

Correggere il codice GPL.

4360 Errore in allocazione di memoria durante l'esecuzione (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

La funzione GPL ha cercato di allocare una zona di memoria per usi interni, ma non ha trovato memoria disponibile.

È possibile che la segnalazione indichi una situazione fisiologica, dovuta per esempio a un eccessivo numero di task contemporaneamente in esecuzione oppure a variabili globali dimensionate in maniera eccessiva.

Soluzione:

Controllare il dimensionamento delle variabili globali e locali e provare a ridurre le dimensioni. Verificare che non vi siano troppi task in esecuzione contemporaneamente e, nel caso, ridurli.

4361 Troppi task attivi (Funzione:NomeFunzione linea:NumeroLinea)**Causa:**

Si è cercato di porre in esecuzione più di 256 task contemporaneamente.

Soluzione:

Ridurre il numero di task attivi contemporaneamente.

4362 Formato matrice errato (Funzione:NomeFunzione linea:NumeroLinea)**Causa:**

Una istruzione che opera su matrici ha trovato un formato non valido. Le istruzioni che possono generare questo errore di sistema sono:

- MOVEMAT nel caso che il formato della matrice sorgente non corrisponda al formato della matrice destinazione.
- CLEAR se si vuole azzerare una riga di matrice che non c'è.
- GETAXIS nel caso che il formato della matrice, passato per parametro, non corrisponda a quello che l'istruzione si aspetta (consultare la documentazione del linguaggio GPL).

Soluzione:

Controllare le istruzioni appena citate all'interno del task che ha generato l'errore. Controllare in particolar modo che le matrici passate alla MOVEMAT abbiano lo stesso numero di colonne dello stesso tipo e che la matrice passata alla GETAXIS abbia il formato giusto.

4363 Troppe istruzioni ONINPUT attive (Funzione:NomeFunzione linea:NumeroLinea)**Causa:**

Sono state attivate più di 128 istruzioni OnInput.

Soluzione:

Ridurre il numero delle ONINPUT.

4364 Asse già impegnato in un riferimento locale (Funzione:NomeFunzione linea:NumeroLinea)**Causa:**

L'errore riguarda l'attivazione delle terne di assi rototraslati per eseguire le interpolazioni su più assi cartesiani. Si è tentato di eseguire una SETRIFLOC, passando all'istruzione un asse che è già impegnato in una terna di assi di riferimento. Viene anche generato nel caso si esegua una RESRIFLOC su un asse che non era impegnato in alcuna terna di assi. L'ultimo caso è che non esistano più terne di riferimento disponibili (al massimo possono essere 32).

Soluzione:

Controllare che le terne passate tramite la SETRIFLOC non abbiano assi in comune.

Controllare le RESRIFLOC.

Controllare anche che ci siano istruzioni di attesa in quota prima della RESRIFLOC.

Considerare comunque che fintanto che l'interpolazione non è terminata, la RESRIFLOC non viene effettivamente eseguita.

4365 Istruzione ONINPUT attivata sullo stesso input (Funzione:NomeFunzione linea:NumeroLinea)**Causa:**

Ad un'istruzione ONINPUT è stato passato lo stesso input per più di una volta.

Soluzione:

Controllare che non ci siano due ONINPUT, a cui venga passato come parametro lo stesso input.

4366 Troppe istruzioni ONFLAG attive (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Sono state attivate più di 128 istruzioni OnFlag.

Soluzione:

Ridurre il numero delle ONFLAG.

4367 Istruzione ONFLAG attivata sullo stesso FLAG (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Ad un'istruzione ONFLAG è stato passato lo stesso flag per più di una volta.

Soluzione:

Controllare che non ci siano due ONFLAG, a cui venga passato lo stesso flag come parametro.

4368 Tentativo di scrittura di una variabile di tipo ReadOnly (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Si è tentato di accedere in scrittura a una variabile di tipo readonly.

Le variabili di tipo readonly sono sempre globali e risiedono nella flash del controllo. Vengono indicate come "static" nell'editor delle variabili globali. Se si prova a scrivere su una di queste variabili globali, verrà generato questo errore di sistema.

L'errore viene, inoltre, generato nel caso si utilizzino variabili che risiedono nella RAM tamponata ("non volatile"), come argomento di alcune istruzioni in scrittura.

Ad esempio, nell'istruzione COORDIN la variabile passata per indicare la riga in elaborazione deve essere in RAM.

Soluzione:

Controllare tutte le variabili static e non volatili.

4369 Troppi assi master attivi (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Si è tentato di attivare come master più di quattro assi contemporaneamente.

Questo errore viene generato solo in esecuzione dell'istruzione di CHAIN.

Soluzione:

Ridurre il numero di assi master.

4370 Troppi assi slave attivi (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Si è tentato di attivare più di otto assi come slave di un unico asse master.

Questo errore viene generato solo in esecuzione dell'istruzione CHAIN.

Soluzione:

Ridurre il numero di assi slave.

4372 Uso errato di una istruzione (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Questo errore viene generato in una delle situazioni seguenti:

1. Si sta usando una istruzione per la gestione mailbox (sendmail, waitmail, endmail, ifmail) o una istruzione per la gestione IPC (sendipc, testipc, waitipc) all'interno di una funzione chiamata da una istruzione Errsys, OnInput o OnFlag.
2. Si sta usando una istruzione IfError o una istruzione IfMessage senza aver abilitato la gestione degli allarmi a stato.
3. Si sta usando l'istruzione Watchdog senza che sia presente la scheda TMSWD.
4. I parametri definiti in una istruzione di interpolazione (linearinc, linearabs, circle, circinc, circabs, helicinc e helicabs) non sono coerenti. Ad esempio, il numero di assi dichiarati è diverso dal numero di quote dichiarate o sono stati dichiarati più assi di quelli che l'istruzione può gestire.

Soluzione:

Di seguito sono descritte le soluzioni per ogni causa elencata:

1. Spostare l'istruzione che provoca l'errore in un'altra funzione o rimuovere l'istruzione.
2. Verificare che sia abilitata la gestione degli allarmi a stato. Nel file tpa.ini nella sezione [ALBATROS] alla voce AlarmsHaveState deve essere assegnato valore 1.
3. Rimuovere l'istruzione Watchdog o procurarsi una scheda TMSWD.
4. Controllare che i parametri dell'istruzione GPL siano corretti. Ad ogni asse deve corrispondere una quota. Il numero di assi dichiarati nell'istruzione non deve essere superiore al numero di assi che la funzione può gestire. Ad esempio, l'istruzione LINEARABS può gestire fino a 6 assi. Se, tra i parametri, ne vengono dichiarati più di 6, l'istruzione genera l'errore di sistema.

**4373 Impossibile eseguire la lettura del feed rate (Funzione:NomeFunzione
linea:NumeroLinea)****Causa:**

È stata usata l'istruzione [GETFEED](#) su una scheda TmsBus o TmsCan, che non è master.

Soluzione:

Verificare in configurazione hardware che la scheda, su cui è collegato il feed rate, sia master.

**4374 Troppe istruzioni di tipo IPC in esecuzione (Funzione:NomeFunzione
linea:NumeroLinea)****Causa:**

Si è superato il limite massimo di 16 istruzioni IPC in esecuzione contemporanea.

Soluzione:

Modificare il codice GPL.

**4375 FASTREAD eseguita su assi di schede differenti (Funzione:NomeFunzione
linea:NumeroLinea)****Causa:**

Si è cercato di eseguire un'istruzione FASTREAD, passandole, come parametri, degli assi che non sono tutti collegati alla stessa scheda.

Soluzione:

Modificare opportunamente il codice GPL o la configurazione Virtuale-fisico.

4378 Istruzione non abilitata (Funzione:NomeFunzione linea:NumeroLinea)**Causa:**

Si è cercato di utilizzare un'istruzione, la cui esecuzione non è abilitata. Probabilmente la chiave hardware non è inserita correttamente o manca.

Soluzione:

Inserire correttamente la chiave hardware. Se il problema persiste, contattare il costruttore.

**4379 L'istruzione non può essere utilizzata in funzioni lanciate da interrupt
(Funzione:NomeFunzione linea:NumeroLinea)****Causa:**

Si è cercato di utilizzare un'istruzione non ammessa in una funzione lanciata da interrupt. Le funzioni lanciate da interrupt sono quelle passate come parametro alle istruzioni ONERRSYS, ONINPUT e ONFLAG.

Soluzione:

Modificare il codice GPL. Consultare [l'elenco delle istruzioni non utilizzabili su interrupt](#)

4380 Troppe richieste di scrittura nell'area di memoria tamponata (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Si è cercato di eseguire troppe operazioni di scrittura sulla memoria tamponata contemporaneamente (la memoria tamponata è caratterizzata da un tempo di accesso relativamente elevato).

Soluzione:

Verificare le istruzioni che effettuano operazioni di scrittura sulle variabili allocate in memoria tamponata: contatori, timer, matrici e variabili dichiarate "nonvolatile".

4381 Non è possibile usare una linea seriale non ancora aperta (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Si è cercato di eseguire un'istruzione che opera su una porta seriale, prima di aver eseguito, per la stessa porta, l'istruzione COMOPEN.

Soluzione:

Modificare il codice GPL.

4382 Non è possibile aprire una linea seriale già aperta (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Si è cercato di eseguire un'istruzione COMOPEN su una porta seriale già aperta con la medesima istruzione.

Soluzione:

Modificare il codice GPL.

4383 Si è cercato di aprire troppi processi ausiliari (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Si è cercato di aprire più di 4 processi ausiliari contemporaneamente.

Soluzione:

Modificare il codice GPL.

4384 Il processo ausiliario non sta girando (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Si è cercato di accedere ad un processo ausiliario che non è in esecuzione.

Soluzione:

Modificare il codice GPL.

4385 Si è cercato di aprire un processo ausiliario da un altro task (Funzione:NomeFunzione linea:NumeroLinea)

Causa:

Si è cercato di aprire un processo ausiliario da un task, che non è quello che ne ha avviato l'esecuzione. Un processo ausiliario può essere utilizzato solo dal task che ne ha iniziato l'esecuzione.

Soluzione:

Modificare il codice GPL.

4391 Errore durante l'attivazione del SYSOK (Funzione:NomeFunzione linea:NumeroLinea)**Causa:**

L'attivazione del segnale di SYSOK non è andata a buon fine. In genere può dipendere da un malfunzionamento del trasmettitore del Greenbus sulla scheda assi.

Soluzione:

I tecnici qualificati possono effettuare un test Hardware della dual port memory del microcontrollore i296. Se il problema si manifesta ancora, contattare il costruttore.

4394 Troppi errori di ciclo (Funzione:NomeFunzione linea:NumeroLinea)**Causa:**

Sono attivi più di 2000 errori di ciclo.

Soluzione:

Correggere il codice GPL limitando il numero di segnalazioni.

4395 Troppi messaggi (Funzione:NomeFunzione linea:NumeroLinea)**Causa:**

Sono attivi più di 2000 messaggi.

Soluzione:

Correggere il codice GPL limitando il numero di segnalazioni.

4397 Stack overflow (Funzione:NomeFunzione linea:NumeroLinea)**Causa:**

Lo stack di una funzione GPL ha superato il limite massimo di 2Kbyte.

Soluzione:

Ricompile il codice GPL e controllare nel report del compilatore l'occupazione stimata di stack della funzione che ha causato l'errore di sistema. Diminuire quindi il numero di variabili locali e di parametri passati alle function (sostituendoli per esempio con variabili globali). Diminuire il numero di CALL.

4398 Stack underflow (Funzione:NomeFunzione linea:NumeroLinea)**Causa:**

Può succedere solo su un errore grave del Firmware: per esempio, nel caso di una gestione errata dei parametri di una funzione o delle variabili locali.

Soluzione:

Contattare il costruttore.

4399 Parametro fuori range (Funzione:NomeFunzione linea:NumeroLinea)**Causa:**

È stato assegnato ad una variabile GPL o ad un dispositivo un valore non compreso nel range consentito.

Soluzione:

Correggere e ricompilare il codice GPL.

4865 Manca la definizione della macchina per l'interpolazione (G216 o G217)**Causa:**

Si è cercato di muovere gli assi con una interpolazione ISO o si sono impostati gli indici di configurazione, senza aver prima definito le matrici di configurazione e gli assi che costituiscono la macchina.

Soluzione:

Correggere e ricompilare il codice GPL utilizzando l'istruzione [ISOG216](#) .

4866 Manca la definizione degli indici della configurazione macchina in presa (M6)

Causa:

Si è cercato di muovere gli assi con una interpolazione ISO, senza avere prima definito gli indici delle matrici di configurazione della macchina

Soluzione:

Correggere e ricompilare il codice GPL utilizzando l'istruzione [ISOM6](#).

6.2.10 Errori generati dal driver delle comunicazioni CNCTPA

16385 Modulo scollegato

Causa:

La connessione tra il PC Supervisore ed un modulo è stata interrotta.

Possibili cause sono:

- mancanza di alimentazione del modulo remoto
- interruzione anche temporanea del collegamento dei cavi ethernet dovuto ad un falso contatto nei connettori o al danneggiamento dei cavi stessi
- mancanza di alimentazione o cattivo funzionamento dell'hub ethernet (se presente)
- blocco del firmware del modulo remoto dovuto al danneggiamento dei file di configurazione
- reset della CPU del modulo remoto dovuto a surriscaldamento o a disturbi EM

Soluzione:

Verificare che il modulo sia acceso. Verificare i cavi ed i connettori ethernet. Aggiornare il firmware a bordo del modulo remoto. Verificare che non sia soggetto a surriscaldamento dovuto a ventilazione insufficiente e che non sia soggetto a disturbi EM. Se il problema persiste contattare il costruttore.

16386 Modulo collegato

Causa:

Un modulo remoto si è collegato al PC Supervisore successivamente alla fase di inizializzazione di Albatros. Durante l'avvio Albatros cerca di collegare tutti i moduli previsti dalla Configurazione di Sistema, questa fase dura circa 4 secondi. I moduli che si collegano successivamente provocano l'errore di sistema.

16387 Modulo ricollegato

Causa:

Un modulo remoto si è ricollegato al PC Supervisore in seguito ad uno scollegamento. Questo errore è quindi sempre consecutivo all'errore 16385 "Modulo scollegato".

16388 Modulo inizializzato

Causa:

Un modulo remoto è stato reinizializzato durante il normale funzionamento. Ciò implica che il modulo si sia scollegato e ricollegato al PC Supervisore. Questo errore è, quindi, sempre consecutivo all'errore 16385 "Modulo scollegato".

Questo errore indica l'avvenuto reset del modulo dovuto, ad esempio, ad una mancanza di alimentazione.

16389 Il modulo ha interrotto il collegamento

Causa:

Un modulo remoto ha interrotto il collegamento con Albatros. Ciò avviene quando il modulo non riceve comandi o interrogazioni da parte del PC Supervisore per un tempo prolungato. Questo errore indica, quindi, un problema (forte rallentamento o blocco) sul PC Supervisore.

Soluzione:

Controllare che sul PC Supervisore non ci siano programmi che provocano il blocco o il rallentamento del sistema. Disabilitare lo screen saver sul PC Supervisore. Se il problema persiste, contattare il costruttore della macchina.

16641 Il firmware del controllo non risponde ai comandi

Causa:

Si è verificato un errore durante l'inizializzazione del sistema. In particolare, il firmware non risponde correttamente. Il problema può dipendere da un danneggiamento dei file di firmware.

Soluzione:

Provare a resettare il sistema ed, eventualmente, reinstallare Albatros. Se il problema persiste, contattare il costruttore.

16642 TpaSock non risponde ai comandi**Causa:**

Si è verificato un errore durante l'inizializzazione del sistema. In particolare, il software di comunicazione con i moduli remoti non risponde correttamente. Il problema può dipendere da un danneggiamento dei file.

Soluzione:

Provare a resettare il sistema ed, eventualmente, reinstallare Albatros. Se il problema persiste, contattare il costruttore.

16643 Il Sistema Operativo non permette di usare RTX**Causa:**

Il sistema operativo installato sul PC non permette l'utilizzo di RTX e, di conseguenza, non consente il corretto funzionamento delle versioni di Albatros, che ne prevedono la presenza.

Soluzione:

Aggiornare il sistema operativo del PC. Consultare i requisiti minimi di sistema sul manuale di installazione di Albatros (InstallationGuide.pdf).

16645 Errore nell'invio del codice firmware**Causa:**

Si è verificato un errore durante l'inizializzazione del sistema. In particolare, è fallito l'invio ad un modulo di un file di firmware.

Soluzione:

Provare a resettare il controllo. Se il problema persiste, contattare il costruttore.

16646 Non è stato possibile rimettere in esecuzione il codice firmware**Causa:**

Si è verificato un errore durante la reinizializzazione del sistema. In particolare, è fallito l'avvio del firmware, dopo che questo era stato precedentemente fermato.

Soluzione:

Provare a resettare il sistema. Se il problema persiste, contattare il costruttore.

16897 Non è stato installato RTX**Causa:**

La versione di Albatros installata richiede che sul PC sia installato RTX, questo però non è stato individuato.

Soluzione:

Installare RTX o, se già presente, installarlo nuovamente. Consultare il manuale di installazione di RTX Albatros (InstallationRTXGuide.pdf).

16898 L'utente non ha i diritti di amministratore**Causa:**

Albatros è stato avviato da un utente privo dei diritti di Amministratore del PC. I diritti di Amministratore sono necessari per il corretto funzionamento di Albatros.

Soluzione:

Chiudere la sessione di lavoro in corso e accedere al sistema come "Administrator" o come altro utente avente i diritti di amministratore.

16899 La dimensione della RAM del modulo è errata

Causa:

La quantità di RAM rilevata sul modulo remoto non corrisponde a quella prevista. In genere, questo errore denota un danneggiamento dell'hardware.

Soluzione:

Se il problema persiste contattare il costruttore.

16900 L'indirizzo IP del modulo è errato

Causa:

È stato rilevato un modulo remoto, il cui indirizzo IP non appartiene alla sottorete del PC supervisore. Albatros non può quindi comunicare correttamente con il modulo.

Soluzione:

Verificare che le impostazioni del servizio AlbdHCP e della scheda di rete del PC siano corrette. Consultare il manuale di installazione di Albatros (InstallationGuide.pdf).

16901 Il modulo è già collegato ad un altro impianto

Causa:

È stato rilevato un modulo remoto che risulta essere collegato ad un altro PC supervisore. Ciò può dipendere dalla presenza in rete di un altro PC, su cui Albatros è in esecuzione e che utilizza il medesimo modulo. Può anche dipendere da un malfunzionamento del software di comunicazione del modulo.

Soluzione:

Verificare che il modulo remoto non venga utilizzato da un altro PC supervisore. Resettare il modulo. Se il problema persiste, contattare il costruttore.

16902 Il modulo non è configurato

Causa:

È stato rilevato un modulo che non risulta essere configurato nella "Configurazione di Sistema" di Albatros.

Soluzione:

Configurare il modulo.

16903 Le impostazioni del firewall impediscono la comunicazione

Causa:

È stato rilevato un firewall installato sul PC che impedisce la comunicazione tra Albatros e i moduli remoti.

Nota: Albatros è in grado di identificare la presenza del firewall di Windows e non di altri firewall, come, ad esempio, quelli inclusi in alcuni software antivirus.

Soluzione:

Modificare le impostazioni del firewall o disattivarlo.

16904 Scheda di rete non presente o disabilitata

Causa:

Non è stata rilevata una scheda di rete utilizzabile per il collegamento ai moduli remoti.

Nota: il fatto che venga rilevata una scheda non garantisce che questa sia correttamente configurata e collegata.

Soluzione:

Verificare la presenza e la corretta configurazione della scheda di rete. Se il problema persiste, contattare il costruttore della macchina.

16905 Manca il codice firmware del controllo

Causa:

Albatros non trova un file di firmware sull'hard disk del PC. Il problema si può manifestare in seguito alla cancellazione accidentale dei file di firmware oppure in seguito ad un aggiornamento errato.

Soluzione:

Verificare che i file contenuti nella cartella FW dell'installazione di Albatros siano presenti e della versione corretta. Contattare il costruttore della macchina.

16906 Versione RTX incompatibile con il codice firmware del controllo

Causa:

È stata rilevata una versione di RTX non compatibile con il firmware installato.

Soluzione:

Installare la versione di RTX corretta o aggiornare il firmware. Contattare il costruttore della macchina.

16907 Versione del sistema operativo incompatibile con il codice di firmware del controllo

Causa:

La versione di sistema operativo del modulo remoto non è compatibile con il firmware installato.

Soluzione:

Installare sul modulo remoto la versione di sistema operativo corretta o aggiornare il firmware. Contattare il costruttore della macchina.

17153 TipoScheda: manca il codice firmware del trasmettitore GreenBUS

Causa:

Non è stato trovato un file di firmware nella cartella FW. In genere, il problema dipende da una cancellazione accidentale del file o da un'installazione incompleta o danneggiata.

Soluzione:

Reinstallare Albatros dopo aver effettuato un backup del sistema. Consultare il costruttore della macchina.

17154 TipoScheda: manca la parte di codice firmware del trasmettitore GreenBUS

Causa:

Il file contenente il codice firmware del trasmettitore GreenBus è presente nella cartella FW, ma risulta essere danneggiato o incompleto.

Soluzione:

Reinstallare Albatros dopo aver effettuato un backup del sistema. Contattare il costruttore della macchina.

17155 TipoScheda: errore nell'invio del codice di bootstrap del trasmettitore GreenBUS

Causa:

Si è verificato un errore durante l'inizializzazione del sistema. In particolare, è fallito l'invio ad un modulo di un file di firmware.

Soluzione:

Provare a resettare il controllo. Se il problema persiste, contattare il costruttore.

17156 TipoScheda: errore nell'invio del codice di Main del trasmettitore GreenBUS

Causa:

Si è verificato un errore durante l'inizializzazione del sistema. In particolare, è fallito l'invio ad un modulo di un file di firmware.

Soluzione:

Provare a resettare il controllo. Se il problema persiste, contattare il costruttore.

17157 TipoScheda: manca il codice di bootstrap

Causa:

Non è stato trovato un file di firmware nella cartella FW. In genere, il problema dipende da una cancellazione accidentale del file o da un'installazione incompleta o danneggiata.

Soluzione:

Reinstallare Albatros dopo aver effettuato un backup del sistema. Contattare il costruttore della macchina.

17158 TipoScheda: manca il codice di Main

Causa:

Non è stato trovato un file di firmware nella cartella FW. In genere, il problema dipende da una cancellazione accidentale del file o da un'installazione incompleta o danneggiata.

Soluzione:

Reinstallare Albatros dopo aver effettuato un backup del sistema. Contattare il costruttore della macchina.

17159 TipoScheda: errore nell'invio del codice di bootstrap

Causa:

Si è verificato un errore durante l'inizializzazione del sistema. In particolare, è fallito l'invio ad un modulo di un file di firmware.

Soluzione:

Provare a resettare il controllo. Se il problema persiste, contattare il costruttore.

17160 TipoScheda: errore nell'invio del codice di Main

Causa:

Si è verificato un errore durante l'inizializzazione del sistema. In particolare, è fallito l'invio ad un modulo di un file di firmware.

Soluzione:

Provare a resettare il controllo. Se il problema persiste, contattare il costruttore.

17409 Non è stato possibile inviare l'eseguibile ausiliario

Causa:

L'errore si può verificare durante l'aggiornamento del firmware di un modulo remoto. Può essere causato da un momentaneo malfunzionamento della rete, ma anche dal danneggiamento del firmware del modulo. Il messaggio di errore può includere un codice di errore.

Soluzione:

Provare a spegnere e riaccendere il modulo remoto e ripetere la procedura di aggiornamento. Se il problema persiste, contattare il costruttore.

17410 Non è stato possibile mettere in esecuzione l'eseguibile ausiliario

Causa:

Si è verificato un errore durante l'inizializzazione del sistema. In particolare, non è stato possibile eseguire un programma ausiliario. Il messaggio di errore riporta, inoltre, il nome del programma ausiliario ed, eventualmente, un codice di errore.

Soluzione:

Provare a resettare il sistema. Se il problema persiste, contattare il costruttore.

17667 NomeDLL: non è stato possibile mettere in esecuzione il codice firmware**Causa:**

Si è verificato un errore durante l'inizializzazione del sistema. In particolare, non è stato possibile eseguire il codice firmware. "NomeDLL" corrisponde al componente che ha sollevato l'errore.

Soluzione:

Provare a resettare il sistema. Se il problema persiste, contattare il costruttore.

17668 NomeDLL: non è stato possibile ottenere il puntatore alla RAM condivisa**Causa:**

Si è verificato un errore durante l'inizializzazione del sistema. In particolare, non è stato possibile aprire il canale di comunicazione con il firmware. "NomeDLL" corrisponde al componente che ha sollevato l'errore.

Soluzione:

Provare a resettare il sistema. Se il problema persiste, contattare il costruttore.

17921 Non è stato possibile inviare NODETPA**Causa:**

L'errore si può verificare durante l'aggiornamento del firmware di un modulo remoto. Può essere causato da un momentaneo malfunzionamento della rete, ma anche dal danneggiamento del firmware del modulo. Il messaggio di errore può includere un codice di errore.

Soluzione:

Provare a spegnere e riaccendere il modulo e ripetere la procedura di aggiornamento. Se il problema persiste, contattare il costruttore.

17922 NODETPA non è ripartito**Causa:**

L'errore si può verificare durante l'aggiornamento del firmware di un modulo remoto. Può essere causato da un momentaneo malfunzionamento, ma anche dal danneggiamento del firmware del modulo. Il messaggio di errore può includere un codice di errore.

Soluzione:

Provare a spegnere e riaccendere il modulo e ripetere la procedura di aggiornamento. Se il problema persiste, contattare il costruttore.

17923 NODETPA non è in esecuzione**Causa:**

È stato rilevato in rete un modulo remoto, il cui software di comunicazione non è in esecuzione. In genere, è sintomo di un malfunzionamento del software di comunicazione. Il messaggio di errore può includere un codice di errore.

Soluzione:

Provare a spegnere e riaccendere il modulo. Se il problema persiste, contattare il costruttore.

18177 NODETPA ha cercato di accedere ad un indirizzo non valido**Causa:**

Il software di comunicazione di un modulo remoto ha generato un errore. Il messaggio di errore può includere un codice di errore.

Soluzione:

Provare a spegnere e riaccendere il modulo. Se il problema persiste, contattare il costruttore.

6.3 Segnalazioni generiche

6.3.1 Albatros inizia l'esecuzione

Segnala la partenza di Albatros e visualizza alcune utili informazioni sulla versione del programma e l'ambiente di esecuzione.

6.3.2 Albatros termina l'esecuzione

Segnala che Albatros sta per terminare l'esecuzione.

6.3.3 Il computer entra nella modalità di sospensione

Segnala che il computer sta per entrare nella modalità di sospensione. Da questo momento in poi Albatros non è più in grado di rispondere a richieste e segnalazioni provenienti dalla ciclica GPL.

6.3.4 Il computer esce dalla modalità di sospensione

Segnala che il computer è appena uscito dalla modalità di sospensione. Albatros riprende l'esecuzione senza essere riavviato.

6.3.5 Spegnimento del computer

Segnala che il computer sta per essere spento mentre Albatros è ancora in esecuzione.

6.3.6 Attuale livello di accesso

Segnala un cambiamento nel livello di accesso alle funzionalità di Albatros, in genere per fare manutenzione o modificare la ciclica o la configurazione.

6.3.7 Aggiornamento software dei moduli

Segnale che è stato richiesto di aggiornare il software e il firmware presenti nei controlli remoti.

6.3.8 Invio della configurazione ai moduli

Segnala che è stato richiesto di aggiornare la configurazione e la ciclica presenti nei controlli remoti.

7 Configurazione del Sistema

7.1 Introduzione

Si è già visto, nel capitolo riguardante la composizione del sistema, come un sistema Albatros sia composto da uno o più moduli che compongono un impianto e come ciascuno di questi sia organizzato in una struttura gerarchica.

Per configurare una macchina dal punto di vista di Albatros, è necessario seguire una sequenza di operazioni, con cui si vanno a configurare i vari livelli logici e l'Hardware sottostante.

L'ordine di massima da seguire nella configurazione di un sistema è il seguente:

- [Configurazione di Modulo](#)
- [Definizione dei Gruppi e Sottogruppi](#)
- [Configurazione dei Dispositivi](#)
- [Configurazione di Sistema](#)
- [Configurazione Hardware](#)
- [Configurazione Virtuale-Fisico](#)

Sostanzialmente le Configurazioni di Modulo, Gruppo e Macchina definiscono la struttura logica della macchina; mentre le Configurazioni di Sistema, Hardware e Virtuale-fisico ne definiscono la struttura a livello fisico. Vedremo in dettaglio ciascuno di questi punti nei prossimi paragrafi.

7.2 Configurazione dei Dispositivi

7.2.1 Introduzione

Nel capitolo relativo alla composizione del sistema Albatros sono stati presentati i tipi di dispositivi che possono comparire in un modulo. Di seguito i dispositivi vengono presentati nuovamente, ma dal punto di vista della loro configurazione.

Per ogni tipo esiste un numero massimo di dispositivi configurabili come specificato nel seguente elenco:

Tipo di dispositivo	Numero max.
Input analogico	128
Output analogico	128
Input digitale	4096
Output digitale	4096
Porto di input	512
Porto di output	512
Asse	240
Timer	128
Contatore	128
Flag Bit	1024
Flag Switch	256
Flag Porto	256

7.2.2 Dispositivo generico

La maggior parte dei dispositivi richiedono i medesimi parametri di configurazione. Di seguito viene presentata la configurazione di un Ingresso Digitale, le medesime considerazioni valgono però anche per:

- Flag bit
- Flag switch
- Uscite analogiche
- Porto di Ingresso
- Porto di Uscita
- Porto di Flag
- Timer
- Contatori

Per configurare un dispositivo tra quelli sopra elencati, si devono specificare le seguenti impostazioni:

- **Nome:** nome del dispositivo avente lunghezza massima di 40 caratteri.
- **Commento:** breve descrizione del dispositivo, può essere tradotta in differenti lingue.

- **Accessi in lettura:** specifica il livello di accesso minimo richiesto, affinché il dispositivo sia visualizzato nelle finestre di Diagnostica e nei Quadri Sinottici.
- **Accessi in scrittura:** specifica il livello di accesso minimo richiesto, per poter modificare lo stato del dispositivo.
- **Pubblico:** specifica se lo stato del dispositivo può essere letto e modificato da codice GPL non appartenente al gruppo in cui il dispositivo è collocato.

7.2.3 Uscita digitale

L'uscita digitale ha un parametro in più rispetto ai dispositivi standard: Monostabile

Per configurare un'uscita digitale si devono specificare le seguenti impostazioni:

- **Nome:** nome del dispositivo avente lunghezza massima di 40 caratteri.
- **Commento:** breve descrizione del dispositivo, può essere tradotta in differenti lingue.
- **Monostabile:** se selezionato, configura l'uscita come monostabile, ovvero quando l'uscita viene impostata ad ON, torna automaticamente ad OFF dopo 200 ms.
- **Accessi in lettura:** specifica il livello di accesso minimo richiesto, affinché il dispositivo sia visualizzato nelle finestre di Diagnostica e nei Quadri Sinottici.
- **Accessi in scrittura:** specifica il livello di accesso minimo richiesto per poter modificare lo stato del dispositivo.
- **Pubblico:** specifica se lo stato del dispositivo può essere letto e modificato da codice GPL non appartenente al gruppo in cui il dispositivo è collocato.

7.2.4 Ingresso analogico

L'ingresso analogico ha un parametro in più rispetto ai dispositivi standard: *il tipo di tensione in ingresso.*

Per configurare un ingresso analogico si devono specificare le seguenti impostazioni:

- **Nome:** nome del dispositivo avente lunghezza massima di 40 caratteri.
- **Commento:** breve descrizione del dispositivo, può essere tradotta in differenti lingue.
- **Tipo:** permette di selezionare l'intervallo di tensioni lette in ingresso.
- **Accessi in lettura:** specifica il livello di accesso minimo richiesto, affinché il dispositivo sia visualizzato nelle finestre di Diagnostica e nei Quadri Sinottici.
- **Accessi in scrittura:** specifica il livello di accesso minimo richiesto, per poter modificare lo stato del dispositivo.
- **Pubblico:** specifica se lo stato del dispositivo può essere letto e modificato da codice GPL non appartenente al gruppo in cui il dispositivo è collocato.

7.2.5 Asse

Dati di Base

I dati di base da specificare sono:

- **Nome:** nome del dispositivo avente lunghezza massima di 40 caratteri.
- **Descrizione:** breve descrizione del dispositivo, può essere tradotta in differenti lingue.
- **Risoluzione:** risoluzione dell'encoder, dipende dalle caratteristiche dell'encoder e dall'unità di misura specificata. Si tenga presente che le schede assi di Albatros conteggiano come impulso sia i fronti di salita che i fronti di discesa di entrambe le fasi encoder (un encoder da 2500 impulsi giro verrà, quindi, visto come un encoder da 10000 impulsi giro).
- **Tipo:** tipo di asse. Sono previsti i tipi **Analogico** (controllato in analogico), **Passo Passo**, **Digitale**, **Conteggio** (sola lettura encoder), **Virtuale**.
- **Unità di misura:** unità di misura utilizzata per esprimere le quote dell'asse. Da essa dipendono tutte le grandezze derivate, per cui si consiglia di impostarla prima di ogni altro parametro.
- **Inversione fasi:** permette di compensare via software un'eventuale inversione di cablaggio delle fasi encoder.
- **Inversione riferimento:** permette di invertire il riferimento di velocità dell'asse. Se utilizzato assieme all'inversione fasi, permette di invertire il verso dell'asse (se il cablaggio è corretto).
- **Abilitazione tacca:** disponibile solo per assi di conteggio, azzerata automaticamente la quota quando viene rilevata la tacca encoder.

Parametri di Movimento

Parametri utilizzati per la movimentazione punto-punto dell'asse.

- **Velocità massima:** velocità massima dell'asse.
- **Accelerazione:** tempo di durata della rampa di accelerazione.
- **Decelerazione:** tempo di durata della rampa di decelerazione.

- **Velocità minima:** impostabile solo per assi passo passo, velocità che il motore raggiunge in un singolo passo.
- **Tipo di rampa:** tipo di rampa di accelerazione e decelerazione. Non impostabile per motori passo passo.
- **Proporzionale:** coefficiente proporzionale del controllore PID dell'anello di posizione.
- **Integrativa:** coefficiente integrativo del controllore PID dell'anello di posizione.
- **Derivativa:** coefficiente derivativo del controllore PID dell'anello di posizione.
- **Feed Forward:** percentuale di feed forward. Permette un corrispondente abbattimento dell'errore d'anello a pari velocità.
- **Feed Forward di Accel.:** percentuale di feed forward di accelerazione. Consente l'abbattimento dell'errore d'anello residuo (non abbattuto dal feed forward) durante le fasi di accelerazione e decelerazione dell'asse.
- **Campioni Integrativa** Imposta il numero di campioni dell'errore d'anello utilizzati per il calcolo della componente integrale. I valori validi sono compresi tra 1 e 200. Il valore di default è 50. Vedi istruzione GPL [SETINTEGTIME](#).

Parametri di Interpolazione

Parametri utilizzati per la movimentazione in interpolazione dell'asse.

Hanno lo stesso significato dei parametri visti nelle impostazioni di Movimento, ad eccezione della Velocità minima, che viene definita solo per gli assi passo passo nei movimenti punto-punto. Questi vengono però impiegati per i movimenti interpolati.

N.B.: I valori di accelerazione e decelerazione, impostati nei parametri di interpolazione, non possono essere inferiori ai valori corrispondenti impostati nei Parametri di Movimento.

Altri Parametri

- **Velocità Manuale:** specifica la velocità massima di configurazione utilizzabile nei movimenti manuali. Non sarà mai superiore alla velocità massima impostata.
- **Servoerror Dinamico:** abilita o disabilita il servoerror dinamico. Il valore di default è servoerror dinamico disabilitato, quindi rimane abilitato il servoerror a soglia. Vedi istruzione GPL [SETMAXERTYPE](#).
- **Velocità di riferimento e Errore d'anello:** questi due valori servono per calcolare il rapporto lineare reale delle due grandezze in macchina. Affinché i valori siano considerati, devono essere entrambi positivi e diversi da zero e deve essere abilitato il campo **Servoerror Dinamico**.
- **Attesa asse fermo:** abilita o disabilita la funzione di recupero overshoot. Introduce un tempo di attesa di 50ms alla fine di ogni movimento.
- **Timeout movimento Asse:** I valori validi sono compresi tra 0 e 1024. Vedi istruzione GPL [ENABLESTARTCONTROL](#).
- **Limite collegamento encoder errato:** I valori impostati sono espressi nell'unità di misura in cui è espressa la risoluzione dell'asse. I valori impostabili devono essere compresi tra 128/risoluzione dell'asse e 16384/risoluzione dell'asse. Il default è calcolato sulla base di un numero di passi pari a 1024, cioè 1024/risoluzione dell'asse.
- **Limite Servoerror Positivo:** valore massimo dell'errore d'anello per il loop di regolazione in direzione positiva.
- **Limite Servoerror Negativo:** valore massimo dell'errore d'anello per il loop di regolazione in direzione negativa.
- **Limite Asse Positivo:** valore massimo della corsa dell'asse in direzione positiva.
- **Limite Asse Negativo:** valore massimo della corsa dell'asse in direzione negativa.
- **Finestra arrivo in quota positivo:** tolleranza sulla quota di arrivo in direzione positiva.
- **Finestra arrivo in quota negativo:** tolleranza sulla quota di arrivo in direzione negativa.

Parametri Riferimento

- **Riferimento:** valore della tensione di riferimento cui corrisponde la velocità massima.
- **Adjust Automatico:** abilita o disabilita il calcolo di recupero dell'offset automatico. Normalmente è abilitato.
- **Offset iniziale:** Valore a cui impostare l'offset iniziale di riferimento. Il valore deve essere compreso tra -10 e 10. Il valore di default è 0.
- **Frequenza filtro di Notch:** Valore di frequenza da filtrare. Il valore deve essere compreso tra 0 e 500.
- **Tensione minima:** Imposta i parametri di tensione minima per l'asse indicato. Il valore negativo deve essere compreso tra -10 e 0, quello positivo tra 0 e +10. Vedi istruzione GPL [SETDEADBAND](#).
- **Soglia:** Imposta i valori di soglia. Sono sempre minori o uguali ai rispettivi valori di tensione minima, quindi il valore negativo di soglia deve essere compreso tra 0 e il valore di tensione minima negativa. Il valore massimo di soglia deve essere compreso tra 0 ed il valore di tensione minima positiva.

Livelli di Accesso

- **Accessi in Lettura:** specifica il livello di accesso minimo richiesto, affinché l'asse venga visualizzato nelle finestre di Diagnostica e nei Quadri Sinottici.
- **Accessi in Scrittura:** specifica il livello di accesso minimo richiesto, per poter modificare lo stato dell'asse.
- **Pubblico:** specifica se lo stato dell'asse può essere letto e modificato da codice GPL non appartenente al gruppo in cui l'asse è collocato.

Concatenamento Assi

Parametri di concatenamento assi. Sono i coefficienti del controllore PID, che compensa le differenze di errore d'anello tra l'asse master e gli assi slave.

- **Proporzionale:** coefficiente proporzionale
- **Integrativa:** coefficiente integrativo
- **Derivativa:** coefficiente derivativo

Correttori di linearità

Impostazione dei correttori di linearità dell'asse. I correttori permettono di compensare sia gli errori di posizionamento di un asse, dovuti all'imprecisione della meccanica dell'asse stesso (autocorrettori) e sia gli errori dovuti all'effetto dato dagli altri assi della macchina (correttori incrociati), tipicamente legato alla flessione della struttura. I correttori non vengono abilitati automaticamente, ma vanno abilitati nella finestra di modifica dei valori di correzione (pulsante **[Modifica...]**) e attivati da codice GPL con l'istruzione [ENABLECORRECTION](#).

- **Intervallo di Correzione:** permette di impostare la distanza tra una correzione e la successiva. Il numero di misurazioni sarà dato dalla lunghezza dell'asse divisa per la lunghezza dell'intervallo di correzione.
- **Nome file correttori:** permette di impostare il nome del file in cui i valori di correzione verranno salvati. Si tratta di un file ASCII, in cui i valori sono separati dal carattere ";". Ciò ne permette la modifica con un normale editor di testo. L'estensione del file non va specificata, viene assegnata automaticamente l'estensione ".csv" (comma separated values).
- **Dati di Correzione:** permette di specificare l'elenco degli assi per cui l'asse corrente genera correzione. L'asse corrente è sempre incluso nell'elenco, ovvero l'autocorrettore è sempre presente. È possibile specificare fino ad altri 5 assi. Per aggiungere un asse, selezionarlo nell'elenco sulla sinistra e premere il pulsante **[>>Aggiungi]**. Per rimuovere un asse, selezionarlo nell'elenco sulla destra e premere il pulsante **[Rimuovi <<]**. Per specificare i valori di correzione, selezionare un asse dell'elenco sulla destra e premere il pulsante **[Modifica...]**, verrà aperta una finestra con una tabella in cui inserire i valori di correzione.

NOTA: Esiste un limite massimo di **235** correttori di linearità gestiti dal sistema per ogni asse. Di conseguenza, data la lunghezza dell'asse, l'intervallo di misurazione dovrà avere una lunghezza maggiore o uguale alla duecentotrentacinquesima parte della lunghezza dell'asse. Ad esempio, per un asse di 2500 mm di lunghezza si dovrà impostare un intervallo di correzione maggiore o uguale a 10.63 mm. Esiste, inoltre, un limite al valore massimo di una singola correzione, questa dovrà essere inferiore a 1024 passi di encoder; ad esempio, per un asse con risoluzione di 256 passi/mm la correzione massima è di ± 4 mm.

7.3 Configurazione Logica

7.3.1 Configurazione di impianto

Per definire una nuova macchina o modificarne una già esistente, si deve accedere alla schermata di Configurazione Modulo. La Configurazione Modulo è la configurazione dei moduli che compongono l'impianto.

L'apertura dell'ambiente di Configurazione è possibile, se si è a livello di accesso costruttore o superiore.

Accesso alla Configurazione

Scegliere dal menu **File** la voce **Apri Configurazione**

Se nell'impianto non ci sono già dei moduli configurati, viene aperta automaticamente la Configurazione Modulo, altrimenti, viene aperta la Configurazione Macchina. In questo caso per passare alla Configurazione Modulo si può fare nel seguente modo:

Scegliere dal menu **Modifica** la voce **Configurazione Modulo**

Per aggiungere un modulo all'impianto, basta premere il pulsante **[Nuovo]**. Il pulsante **[Modifica]** permette di modificare i dati di un modulo già esistente, il pulsante **[Elimina]** permette la cancellazione di un modulo, il pulsante **[Chiudi]** permette di uscire dalla configurazione di impianto.

I dati che identificano una macchina, e che si dovranno specificare, sono:

- il numero di modulo: un numero intero progressivo che, se non specificato, viene assegnato dal sistema
- una breve descrizione.

Vi sono alcuni dati che riguardano l'Hardware:

- **Frequenza controllo assi:** indica la frequenza con cui vengono periodicamente scambiati i dati tra il controllo numerico e i dispositivi ad esso collegati tramite i bus di campo.
- **Numero canale interpolazione:** indica il massimo numero di canali di interpolazione (cioè, di gruppi di assi che eseguono un movimento di tipo interpolato), che possono essere gestiti simultaneamente.
- **Percentuali d'uso della CPU:** indica la percentuale di tempo, relativa al periodo di controllo assi (cioè all'inverso della "frequenza controllo assi"), che viene riservata all'esecuzione del firmware.

La stessa finestra può essere aperta dal ramo del modulo della Configurazione dei gruppi, dal ramo Modulo della Configurazione di macchina e dal ramo del modulo della Configurazione Hardware.

7.3.2 Configurazione dei gruppi

La prima volta che si progetta una macchina è necessario definire ogni suo componente e scrivere tutte le cicliche di controllo. In molti casi la progettazione viene svolta a partire da una macchina già realizzata, che viene modificata in base alle caratteristiche della nuova macchina.

Creazione di un Gruppo

Per creare un nuovo gruppo è necessario accedere alla schermata di configurazione gruppi. Il primo ramo dell'albero è il modulo, da cui derivano tutti i gruppi, i sottogruppi e i dispositivi. Se si preme il tasto **[INVIO]** o il pulsante **[Modifica]**, si apre la finestra di dialogo per la modifica dei dati del modulo.

Scegliere dal menu **Modifica** la voce **Gruppi**

Da qui è possibile creare nuovi gruppi, modificare o cancellare quelli esistenti e copiare un gruppo.

Elenco dei comandi per creare, modificare, cancellare, copiare e incollare gruppi, sottogruppi e dispositivi

Comando	Azione
Crea un nuovo gruppo, sottogruppo, dispositivo	[Ctrl+Invio], Pulsante [Nuovo], Modifica->Nuovo, menu contestuale
Modifica un gruppo, sottogruppo, dispositivo	[Invio], Pulsante [Modifica], Modifica->Modifica.., menu contestuale
Elimina un gruppo, sottogruppo, dispositivo	[Canc], Pulsante [Elimina], Modifica->Elimina, menu contestuale
Abilita o disabilita l'uso in macchina di un gruppo	Menu contestuale, Pulsante [Abilita]
Copia un gruppo, sottogruppo, dispositivo	[Ctrl+C], Pulsante [Copia], Modifica->Copia, menu contestuale
Incolla un gruppo, sottogruppo, dispositivo	[Ctrl+V], Pulsante [Incolla], Modifica->Incolla, menu contestuale

Quando si crea un nuovo gruppo compare una finestra, in cui si dovranno impostare:

- Il nome del gruppo
- Un commento (traducibile nelle lingue gestite da Albatros)

È, inoltre, possibile marcare il gruppo come **Intergruppo**. Deve essere impostato almeno un gruppo come intergruppo, perché questa selezione viene utilizzata da Albatros per identificare il gruppo "principale" della macchina. Questo è il gruppo di cui viene lanciata automaticamente all'avvio la funzione principale (quella con lo stesso nome del gruppo). Questo meccanismo serve per inizializzare la macchina ed eseguire all'avvio i task che verificano che tutto funzioni correttamente, prima di passare il controllo all'utilizzatore. Quando si disabilita un gruppo, che ha dei dispositivi collegati a dei dispositivi fisici, viene richiesto se si vuole cancellare il collegamento virtuale fisico. Se si sceglie di mantenere i collegamenti, i pin dei dispositivi fisici, a cui sono collegati, verranno visualizzati in grigio nella rappresentazione grafica del virtuale fisico.

Aggiunta di un sottogruppo ad un gruppo

Per creare un sottogruppo di un gruppo si deve essere posizionati sul gruppo.

Se non si vuole assegnare un nome specifico al sottogruppo, si può selezionare la casella *Lista dei Dispositivi* e premere il pulsante **[OK]**. Il nome del sottogruppo viene assegnato automaticamente.

Ora è possibile inserire i singoli dispositivi all'interno del sottogruppo. Il meccanismo è simile a quello per la creazione dei sottogruppi. Compare una finestra con un elenco dei dispositivi disponibili.

Selezionato il dispositivo che ci interessa, si preme il pulsante **[OK]** per la conferma.

Compare una finestra che ci permette di impostare il nome, un commento e altri dati che variano in relazione al dispositivo selezionato. Una descrizione dettagliata dei tipi di dispositivi e delle loro impostazioni è esposta nel capitolo [Configurazione dei Dispositivi](#).

Copia di un dispositivo

La funzione di copia di un dispositivo permette di copiare un dispositivo. Si deve prima di tutto selezionare il dispositivo e di seguito il pulsante **[Copia]**. Per inserire il dispositivo nella lista, si deve selezionare il ramo in cui incollare il dispositivo e attivare il comando **[Incolla]**. Nella finestra di dialogo deve essere inserito il nuovo nome del dispositivo.

Copia di un sottogruppo

La funzione di copia di un sottogruppo permette di copiare un sottogruppo comprensivo di tutti i dispositivi in esso contenuti. Per inserire il sottogruppo si deve selezionare il ramo in cui incollarlo e attivare il comando **[Incolla]**. Nella finestra di dialogo deve essere inserito il nuovo nome del sottogruppo.

Copia di un gruppo

La funzione di copia di un gruppo permette di copiare un gruppo comprensivo di tutti i sottogruppi ed i dispositivi in esso contenuti. Inoltre, viene copiato l'eventuale sinottico di gruppo ad esso associato (sinottico, il cui nome coincide con quello del gruppo).

Permette di creare velocemente dei gruppi che hanno una struttura simile a quella di un gruppo già esistente, senza dover ricreare ad uno ad uno tutti i dispositivi. Per copiare un gruppo occorre selezionare il gruppo che si intende copiare, selezionare il pulsante **[Copia]** ed inserire nella finestra di dialogo il nome del nuovo gruppo.

La copia di dispositivi, sottogruppi e gruppi può essere eseguita anche tra moduli diversi.

Scelta dei gruppi che appartengono alla macchina

Dopo aver creato l'archivio dei gruppi, si devono disabilitare o abilitare i gruppi effettivamente presenti. I gruppi sono tutti presenti in macchina, a meno che non si decida di disabilitarli tramite il pulsante **[Disabilita]** o selezionando lo stesso comando dal menu contestuale. Se un gruppo è stato disabilitato, appare la dicitura **Non presente** in fianco al nome del gruppo.

Per visualizzare i soli gruppi presenti in macchina scegliere dal menu **Modifica** la voce **Macchina**.

Per inserire un nuovo gruppo, si deve premere il pulsante **[Inserisci]**. Comparirà una finestra contenente un elenco dei gruppi presenti nell'archivio Gruppi e non ancora inseriti in macchina.

A questo punto si deve selezionare il gruppo prescelto e **trascinarlo con il mouse** nella finestra di Configurazione Macchina oppure selezionare il pulsante **[Inserisci]**.

È, inoltre, possibile rimuovere tramite il pulsante **[Rimuovi]** un gruppo esistente o effettuare la ricerca del nome di un gruppo o di un dispositivo all'interno dell'albero di composizione della macchina.

In una macchina deve essere presente un unico intergruppo.

7.4 Configurazione Fisica

7.4.1 Configurazione di sistema

La configurazione di sistema permette di associare delle risorse fisiche (unità di controllo) ai moduli definiti nella configurazione logica. Questo è possibile nella finestra di dialogo Configurazione Sistema. Viene presentata la lista dei **moduli** dell'impianto e ad ognuno di essi è possibile associare un **Nodo di rete**.

- **Nodo locale:** Sistemi "Locali" in cui l'HW che implementa il controllo è montato direttamente sul PC che costituisce l'interfaccia utente del sistema
- **Nome di un nodo di rete:** Sistemi "Remoti" in cui l'HW che implementa il controllo è collegato al PC via rete
- **Non configurato:** nessuna configurazione. È il default iniziale. Se si lascia questa scelta, sarà in seguito possibile nella finestra di dialogo **Connessione nodi di rete** associare un modulo remoto.

Si possono configurare fino a 16 moduli e solo uno può essere configurato come nodo locale.

Per assegnare un modulo selezionare il pulsante **[Modifica]** oppure eseguire un doppio click del mouse in corrispondenza del nodo di rete da modificare. Aprendo il menu a tendina viene visualizzata la lista dei moduli remoti disponibili, la scelta di utilizzare un Nodo locale o la scelta di impostare il modulo come non configurato. Per

confermare la selezione, si deve selezionare il pulsante .

N.B.: Il funzionamento di Albatros in macchina è protetto dalla presenza di una chiave hardware usb, configurata da TPA.

7.4.2 Configurazione Hardware

In configurazione hardware si definiscono le schede e i nodi che compongono il sistema.

La scheda che occupa la prima posizione nell'elenco è detta scheda Master.

I tipi di scheda configurabili sono:

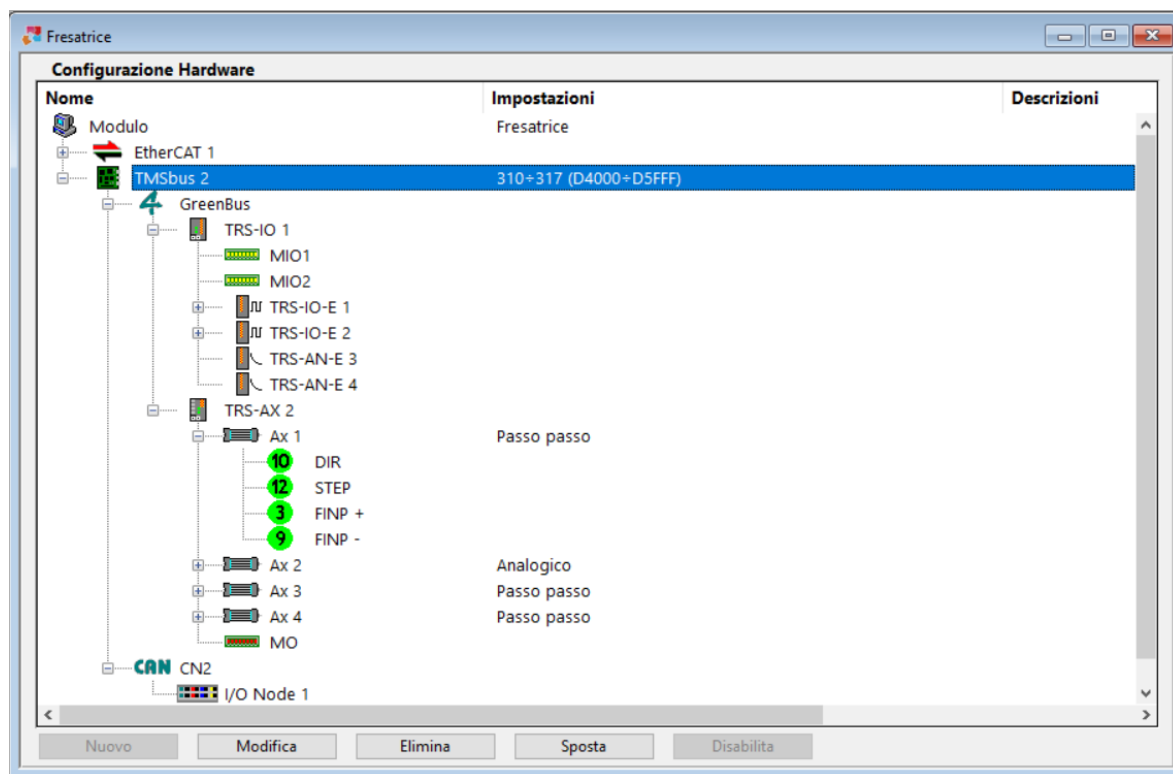
- TMSbus fino a due
- TMSbus+ fino a quattro
- TMSCombo+ fino a quattro
- DualMech fino a quattro
- DualMech Mono fino a quattro
- TMSCan fino a due
- TMSCan+ fino a quattro
- AlbMech fino a due
- EtherCAT una

Descrizione della finestra di configurazione hardware

La finestra di configurazione hardware si apre selezionando da menu la voce **Modifica->Hardware**.

Per inserire una scheda o un modulo di I/O remoto o un nodo CAN o EtherCAT, si deve premere il pulsante **[Nuovo]**. Verrà così visualizzata una finestra che permette la selezione della scheda o del modulo di I/O remoto e per il bus CAN e GreenBUS la posizione in cui deve essere inserita.

In generale, per ogni modulo si possono configurare fino ad un massimo di 4 schede e a seconda del tipo di scheda e di bus un numero variabile di moduli di I/O remoti.



Configurazione Hardware

Nella colonna **Impostazioni** si assegnano le informazioni legate alla scheda o al nodo.

Con il comando **[Sposta]** è possibile spostare una scheda o un nodo o una espansione di un TRS-IO o di un TRS-CAT da una posizione ad un'altra dell'albero. Le espansioni possono essere spostate solo all'interno dello stesso nodo. Questa operazione mantiene i collegamenti presenti nella configurazione [Virtuale-Fisico](#).

È, inoltre, possibile disabilitare un nodo o una espansione di un TRS-IO o di un TRS-CAT.

La disabilitazione fa in modo che vengano mantenuti i collegamenti in configurazione Virtuale-Fisico.

Se il nodo appartiene ad un bus GreenBUS, il nodo e i dispositivi a esso collegati vengono totalmente ignorati dal sistema. Non viene, quindi, generato alcun errore se il modulo non viene rilevato durante l'inizializzazione e non viene generato alcun errore quando viene eseguita un'istruzione GPL su un dispositivo associato al modulo.

Se il nodo o l'espansione appartiene alla rete EtherCAT o l'espansione appartiene ad un bus GreenBUS non deve essere presente nella rete. Se in GPL si accede ad un dispositivo non collegato, si verifica un errore di sistema. *Va posta particolare attenzione nell'utilizzo di questa funzionalità.*

Per disabilitare un nodo o un'espansione usare il comando **[Disabilita]**, per riabilitare un nodo o un'espansione usare il comando **[Abilita]**.

Configurazioni predefinite

Sono disponibili diverse configurazioni predefinite. Da menu **Modifica->Modifica tipo controllo** o dal menu contestuale sul ramo del Modulo è possibile selezionare la configurazione desiderata. Se la configurazione è nuova, l'albero viene popolato con le schede e i nodi definiti, altrimenti vengono eseguiti dei controlli per verificare che gli hardware già presenti siano compatibili con il tipo modulo selezionato. Tutto ciò che non è compatibile viene eliminato.

Configurare un nodo di un bus TPA

I tipi di modulo di I/O remoto configurabili su GreenBus (v3.0) sono:

- Albre8 8 ingressi e 8 uscite digitali

- Albre16 16 canali configurabili via software come ingresso o uscita digitale
- Albre24 24 ingressi e 24 uscite digitali
- Albre48 48 ingressi e 48 uscite digitali
- Albrem 10 porti in ingresso e 10 porti in uscita
- AlbSTEP 8 ingressi e 6 uscite digitali, un motore passo-passo
- AlbEV 20 o 24 elettrovalvole (connettore D-sub 25 pin)
- Albrea 4 ingressi e 4 uscite analogiche

I tipi di modulo remoto configurabili su GreenBus (v4.0) sono:

- TRS-AX 4 assi analogici o passo-passo
- TRS-EV-24 24 elettrovalvole (connettore D sub 25 pin)
- TRS-16 16 canali configurabili via software come ingresso o uscita digitale
- TRS-IO 16 canali configurabili via software come ingresso o uscita digitale espandibile con moduli tipo TRS-IO-E e moduli TRS-AN-E fino ad un massimo di 5 e moduli TRS-AC-E
- TRS-IO-E 16 canali configurabili via software come ingresso o uscita digitale utilizzabili solo come espansioni di un modulo TRS-IO
- TRS-AN-E 1 ingresso analogico e 1 uscita analogica utilizzabile solo come espansioni di un modulo TRS-IO
- TRS-REM remoto generico utilizzato per la realizzazione di moduli speciali. Sono collegabili fino a 12 porti in ingresso e 12 porti in uscita
- TRS-AC-E 1 asse di conteggio e 2 ingressi digitali configurabili come tacca di zero e fast input. La tabella seguente descrive il numero massimo di TRS-AC-E configurabili in un TRS-IO.
Se ci sono 3 espansioni in tutto tra TRS-IO-E e TRS-AN-E, può essere configurata una sola espansione TRS-AC-E. Se è configurata una sola espansione (TRS-IO-E o TRS-AN-E), si possono avere fino a 2 espansioni TRS-AC-E.

I moduli remoti TRS-AX, TRS-IO, TRS-REM e TRS-16 sono collegabili esclusivamente a schede TMSbus, TMSbus+ e TMSCombo+.

Ad ogni scheda TMSbus e TMSbus+ si possono collegare al massimo 4 moduli remoti TRS-AX.

I tipi di modulo remoto di Tpa configurabili su bus EtherCAT sono:

- TRS-CAT 16 canali configurabili via software come ingresso o uscita digitale espandibile con moduli tipo TRS-IO-E, moduli TRS-AN-E e moduli TRS-AC-E
- STAR-CAT trasforma una topologia di rete EtherCAT lineare in una topologia a stella utilizzando un canale in ingresso e fino a 3 diversi canali in uscita

La tabella seguente descrive il numero massimo di espansioni configurabili in un TRS-CAT

Numero espansioni TRS-IO-E e TRS-AN-E	Numero espansioni TRS-AC-E
7	0
5	1
3	2
1	3

Per i moduli remoti di tipo TRS-AX se si aumenta il numero di TRS-AX inseriti, diminuisce il numero di TRS-16 e TRS-IO che possono essere utilizzati.

Per calcolare il numero massimo di remoti di tipo TRS-16 e TRS-IO che possono essere inseriti, occorre applicare la seguente formula: numero di altri remoti = 32 - (numero di TRS-AX * 4). Ad esempio, se ad una scheda TMSbus sono collegati 3 TRS-AX, applicando la formula avremo: numero di altri remoti = 32 - (3*4), quindi può essere inserito un numero massimo di 20 moduli remoti di tipo TRS-16 e/o TRS-IO.

La scelta della posizione del remoto deve essere fatta in base all'indirizzo impostato tramite switch sul modulo remoto. Fare riferimento alla documentazione hardware del singolo remoto.

Se si seleziona un remoto TRS-AX può essere necessario impostare il tipo di assi gestito.

Nello schema seguente vengono descritti quali tipologie di assi possono essere associati ai vari hardware.

- scheda AlbMech assi digitali

- scheda DualMech assi digitali
- scheda DualMech Mono assi digitali
- remoto TRS-AX assi analogici (se lo specifico connettore è di tipo Analogico), assi di conteggio (se lo specifico connettore è di tipo Analogico), assi passo-passo (se lo specifico connettore è di tipo Passo passo)
- remoto AlbStep assi passo-passo
- espansione TRS-AC-E assi conteggio

Nelle schede MECHATROLINK-II ogni asse può essere gestito in controllo di posizione o in controllo di velocità (default). Il tipo di controllo per ogni asse può essere modificato nella finestra di Configurazione hardware nella colonna **Impostazioni**. Il numero di assi configurabili varia in base al valore di frequenza controllo assi impostata:

Scheda	Frequenza Controllo Assi (Hz)	Numero massimo servoazionamenti
AlbMech	1000	8
AlbMech	<=500	16
DualMech Mono	1000	8
DualMech Mono	500	20
DualMech Mono	250	30
DualMech	1000	16
DualMech	500	40
DualMech	250	60

Configurare un nodo di un bus CAN

La scheda di controllo del bus

Albatros può gestire dispositivi su bus di campo CAN tramite schede Tpa dotate di connettore per bus CAN. Il bus CAN è presente sulle schede **TMSbus**, **TMSbus+**, **TMSCan+** e **TMSCan**.

Configurare i dati di base e i servizi

I dati del bus CAN vengono definiti in configurazione hardware. Si seleziona il bus CAN, di cui si vuole definire i parametri e si clicca sul pulsante **[Modifica]**.

I dati di base sono:

- **Tempo di campionamento (TIME)**: tempo di campionamento in msec. Non può essere superiore a 60000 (60 secondi). Il default è 2. Su bus S-CAN è accettato solo il valore 2.
- **Tempo per la comunicazione sincrona del PDO (TIMEPDO)**: tempo espresso in msec. Indica il tempo dedicato alla comunicazione sincrona dei PDO. Il valore impostato non può essere superiore al valore di TIME (Non è un valore obbligatorio).
- **Tempo di attesa (TIMEAFTERRESET)**: tempo espresso in msec. Indica il tempo di attesa durante la fase iniziale a seguito di un reset software dei nodi in rete. Non può essere superiore a 60000 (60 secondi)
- **Numero cicli CAN senza risposta (LIFETIMEFACTOR)**: è il numero di cicli CAN senza risposta alla chiamata di Node Guarding, prima che venga generato l'errore di nodo scollegato. Il valore non può essere superiore a 100 e inferiore a 1. (Il default è 3).
- **Baud rate**: numero velocità della comunicazione CAN in kilobit al secondo (il valore può essere 1000, 500, 250, 125, 100).

I servizi possono essere abilitati o disabilitati selezionando o deselegionando l'opzione che si riferisce al servizio. Al campo **Extra** possono essere assegnati valori che hanno un significato stabilito dal costruttore della macchina. Su questo valore non viene eseguito alcun controllo. Il default è 16.

Il nodo CAN

Inserire un nuovo nodo

Un nodo viene inserito selezionando il ramo CAN del tree e cliccando sul pulsante **[Nuovo]**. Il tipo di nodo viene dedotto dal tipo di bus. Se il bus è di tipo CAN, il nodo è un I/O Node, se il bus è di tipo S-CAN, il tipo di bus inserito è di tipo Servo. Nella finestra di dialogo per l'inserimento del nodo deve essere scelta:

- **Posizione:** è il numero di nodo (da 1 in poi).
- **AutoOp:** se selezionato, il dispositivo ammette il passaggio automatico allo stato Operazionale in seguito a un ricollegamento.

Configurare un nodo

Per ogni nodo è possibile definire i PDO di trasmissione e i PDO di ricezione. Se il bus è CanOpen, si possono definire massimo 8 PDO per schede TMSBus e TMSBus+, massimo 4 per schede TMSCan e TMSCan+. Si seleziona nel tree il nodo e si clicca sul pulsante **[Nuovo]**. I dati da impostare nella finestra di dialogo sono:

- **Tipo PDO:** scegliere **Trasmissione** o **Ricezione**, a seconda che si definisca un TPDO o un RPDO.
- **Dimensione:** dimensione del PDO di ricezione o di trasmissione.
- **COB-ID:** valore che può essere definito solo su schede TMSBus e TMSBus+. Il valore è visualizzato e memorizzato in decimale. Per visualizzare il valore in notazione esadecimale, si deve selezionare il checkbox **Esadecimale** posto a fianco del valore stesso.
- **Asincrono:** se abilitato, vengono configurati PDO asincroni, cioè che non vengono aggiornati ad ogni ciclo. Questa opzione è gestita solo su schede TMSCan e TMSCan+. La ricezione dei PDO asincroni va fatta nel codice GPL tramite l'istruzione [RECEIVEPDO](#).

Caratteristiche della gestione EtherCAT in Albatros

La modalità di comunicazione è sempre DC-Synchronous. Il primo nodo della rete fornisce il clock, perciò è essenziale assicurarsi che quel nodo fornisca un clock preciso e stabile, come viene fornito ad esempio da TRS-CAT. Non è possibile usare altre modalità, quali ad esempio Free-Run.

I protocolli gestiti sono: CoE (CAN application protocol over EtherCAT) e EoE (Ethernet over EtherCAT).

All'interno di CoE, sono gestiti i dispositivi secondo le specifiche DS401 e DS402, con modalità operativa di default di controllo asse *cyclic synchronous velocity mode*.

Il numero massimo di nodi EtherCAT è 200.

Premessa

Ad ogni dispositivo fisico EtherCAT è associato un file ESI (EtherCAT Slave Information) che descrive le caratteristiche e le funzionalità che il dispositivo espone. Il file è in formato XML. Per ogni dispositivo deve esistere un solo file ESI. In genere i file ESI possono essere scaricati dal sito Internet del costruttore. Albatros cerca questi file nella cartella definita in Tpa.ini nella sezione [tpa] alla voce DirESIFiles. Il default è la sottocartella "\\EtherCAT" di SYSTEM.

Dai file ESI Albatros ricava le informazioni sul dispositivo, analizzando tutti gli elementi "/Devices/Device/Type". Ogni dispositivo è identificato da un Vendor ID, un Product ID e un Revision Number.

Sempre dai file ESI vengono ricavate le informazioni sulle espansioni (chiamate anche moduli) dei dispositivi. Albatros trova le informazioni relative ai tipi delle espansioni cercando nel file ESI del dispositivo gli elementi "Modules/Module".

Configurazione dell'hardware EtherCAT

L'hardware si configura descrivendo le schede master e per ogni scheda l'elenco dei dispositivi fisici collegati sul bus a quella scheda. I dispositivi fisici vengono anche chiamati "nodi" del bus di campo. Per EtherCAT la scheda master non è una specifica scheda di controllo del bus, ma si usa una connessione di rete del modulo.

Per quanto riguarda i moduli locali la connessione di rete deve essere tra quelle gestite da RTX, mentre per i moduli remoti viene usata una specifica connessione di rete del modulo, tra quelle gestite da Windows CE 6.0. Per ogni modulo, locale o remoto, è possibile configurare un solo master.

La scheda master deve essere inserita in configurazione hardware selezionando il ramo Modulo e cliccando sul pulsante **[Nuovo]**.

Il nodo EtherCAT

Per inserire un nodo selezionare il ramo scheda del tree e cliccare sul pulsante **[Nuovo]**. Gli indici dei nodi devono essere consecutivi, per cui ad ogni comando **[Nuovo]** viene aggiunto un nodo in coda agli altri. Per ogni nodo devono essere definiti i dati seguenti:

- **Nome del dispositivo:** selezionare il nome del dispositivo tra quelli elencati. Vengono visualizzati solo i dispositivi con revisione più recente; se si vogliono vedere tutte le revisioni, è necessario selezionare la voce **Visualizza revisioni**.
- **Modalità asse:** selezionare la modalità operativa da utilizzare per i nodi di tipo azionamento, secondo quanto specificato nella normativa DS402 in relazione all'oggetto 6060₁₆ "mode of operation". La scelta è tra **Cyclic synchronous position** e **Cyclic synchronous velocity**. Quest'ultima è il default.
- **Forza come nodo di I/O:** se abilitato, forza il controllo numerico a considerare l'azionamento come un nodo di I/O. Questo attributo si applica solo a nodi che supportano DS402 (servoazionamenti).

La modalità asse è l'unico dato che può essere modificato, dopo avere inserito il nodo.

Un nodo EtherCAT può essere eliminato, spostato, disabilitato e copiato. I comandi sono disponibili nel menu contestuale di ogni ramo, nel menu principale di Albatros e in alcuni pulsanti presenti nella finestra di configurazione hardware.

Il comando **Elimina** cancella il nodo selezionato, tutte le sue espansioni e i suoi collegamenti virtuale fisici e tutti i nodi successivi vengono spostati all'indirizzo precedente.

Il comando **Sposta**, sposta il nodo selezionato ad un nuovo indirizzo. I nodi successivi all'indirizzo scelto vengono anch'essi spostati di una posizione.

Il comando **Copia** memorizza i dati del nodo selezionato per utilizzarli in un successivo comando Incolla.

Il comando **Incolla** inserisce, nell'albero dei dispositivi, il nodo precedentemente copiato. Il nodo viene inserito nella posizione scelta dall'utente. Tutti i nodi seguenti vengono spostati.

L'espansione EtherCAT

Per inserire una espansione selezionare il nodo a cui si vuole aggiungere l'espansione e cliccare sul pulsante **[Nuovo]**. Gli indici delle espansioni devono essere consecutivi, per cui ad ogni comando **[Nuovo]** viene aggiunta una espansione in coda alle altre. Gli oggetti dei PDO delle espansioni non sono modificabili.

Descrizione di un PDO

Si possono definire fino a 16 PDO inviati dal nodo (TxPDO) e fino a 16 PDO ricevuti dal nodo (RxPDO). Ogni RxPDO descrive un solo PDO che il nodo riceve dal master, quindi uscite digitali e analogiche per nodi di I/O o velocità target e controlword per nodi asse. Ogni TxPDO descrive un solo PDO che il nodo invia al master, quindi ingressi digitali e analogici per nodi di I/O o posizione corrente e statusword per nodi asse.

Per l'elenco e la descrizione dei PDO e degli oggetti mappabili su un PDO, bisogna fare riferimento alla documentazione dello specifico dispositivo EtherCAT e al suo file ESI.

Ci sono tre modalità per descrivere i PDO di un nodo:

- Non indicare alcun PDO.
In questo modo il controllo numerico utilizza i PDO configurati di default nel dispositivo.
- Indicare solo i PDO senza fornire l'elenco degli oggetti.
Da usare quando un CN ha vari PDO alternativi e non programmabili.
- Descrivere in modo completo il PDO, indicando l'oggetto di comunicazione e l'elenco degli oggetti da mappare.
Questa modalità è quella che dà il maggior controllo sulle informazioni che vengono inviate e ricevute dal nodo.

Ogni oggetto è descritto dal suo indice nell'object dictionary del CN, opzionalmente seguito da un sottoindice. Se il sottoindice non è presente, si considera che sia 0.

L'oggetto del dizionario (object dictionary) è il cuore di ogni dispositivo. Abilita l'accesso a tutti i tipi di dati del dispositivo, ai parametri di comunicazione e ai parametri di configurazione ed elaborazione dati.

Attenzione: non tutti gli oggetti dell'object dictionary possono essere mappati in un PDO.

Modifica di un PDO di un azionamento

Per quanto riguarda i nodi di tipo servoazionamento c'è un PDO per ogni asse, così che l'ennesimo TxPDO e l'ennesimo RxPDO del nodo fanno riferimento all'ennesimo asse del nodo. I primi due oggetti di ogni RxPDO e TxPDO hanno un significato e una dimensione preassegnati. La tabella seguente descrive come configurare i PDO relativi agli assi di uno stesso nodo, controllati in modalità asse **Cyclic synchronous velocity**.

Azionamento	RxPDO		TxPDO	
	1° oggetto 16 bit Controlword	2° oggetto 32 bit Target velocity	1° oggetto 16 bit Statusword	2° oggetto 32 bit Actual position
1° asse	6040 ₁₆	60FF ₁₆	6041 ₁₆	6064 ₁₆
2° asse	6840 ₁₆	68FF ₁₆	6841 ₁₆	
ennesimo asse	Aggiungere 800 ₁₆ a ogni oggetto dell'asse che lo precede.			

La tabella seguente descrive come configurare i PDO relativi agli assi di uno stesso nodo, controllati con modalità asse **Cyclic synchronous position**.

Azionamento	RxPDO		TxPDO	
	1° oggetto 16 bit Controlword	2° oggetto 32 bit Target position	1° oggetto 16 bit Statusword	2° oggetto 32 bit Actual position
1° asse	6040 ₁₆	607A ₁₆	6041 ₁₆	6064 ₁₆
2° asse	6840 ₁₆	687A ₁₆	6841 ₁₆	6864 ₁₆
ennesimo asse	Aggiungere 800 ₁₆ a ogni oggetto dell'asse che lo precede.			

Per modificare gli oggetti di un asse selezionare l'asse nel tree e cliccare sul pulsante **[Modifica]**. Nella finestra di dialogo vengono visualizzati l'indice dei PDO di trasmissione, l'indice del PDO di ricezione e la lista degli oggetti configurati.

Nello specifico i dati della finestra di dialogo sono:

- **Indice PDO di trasmissione:** è il valore del PDO di trasmissione dell'asse. Non è un valore modificabile.
- **Oggetti:** elenco degli oggetti del PDO di trasmissione. I primi due oggetti vengono stabiliti in base alla modalità asse scelta.
- **Elenco oggetti:** elenco degli oggetti del PDO di trasmissione che possono essere utilizzati. Per ogni oggetto vengono definiti, se presenti, oltre all'indice anche il sottoindice e la descrizione.
- **Indice PDO di ricezione:** è il valore del PDO di ricezione dell'asse. Non è un valore modificabile.
- **Oggetti:** elenco degli oggetti del PDO di ricezione. I primi due oggetti vengono stabiliti in base alla modalità asse scelta.
- **Elenco oggetti:** elenco degli oggetti del PDO di ricezione che possono essere utilizzati. Per ogni oggetto vengono definiti, se presenti, oltre all'indice anche il sottoindice e la descrizione.

I pulsanti **[Su]** e **[Giù]** servono per modificare l'ordine degli oggetti nell'elenco della finestra **Oggetti**.

Il pulsante **[Aggiunge]** permette di inserire un oggetto che non è tra quelli presenti nell'**Elenco oggetti**.

Per modificare il valore di un oggetto si deve selezionare l'oggetto e fare doppio click col mouse. La sintassi per la definizione di un nuovo oggetto è:

- **numero oggetto**: è il numero dell'oggetto in formato esadecimale.
- **sottoindice**: è il numero del sottoindice. Deve essere separato dal numero oggetto tramite il carattere '.'(punto). Si tratta di un valore opzionale che, se non è definito, viene assunto 0 come valore di default. È in formato decimale.
- **L**: dimensione dell'oggetto in bit. Deve essere un multiplo di 8.

I valori non presenti vengono ricavati dai dati dell'object dictionary o vengono assegnati valori di default.

Esempio:

oggetto completo anche di sottoindice e lunghezza: 1600.1L16

oggetto senza sottoindice: 1601L24

oggetto semplice: 1603

Tali valori possono poi essere letti da GPL tramite l'istruzione GETAXIS, cui si rimanda. È possibile, inoltre, tracciare gli oggetti aggiuntivi sia dalla finestra di taratura che da oscilloscopio.

Più in generale, da GPL è possibile accedere in lettura e scrittura a specifici oggetti all'interno di un PDO tramite le istruzioni [GETPDO](#) e [SETPDO](#), cui si rimanda.

PDO aggiuntivi

Per definire dei PDO aggiuntivi di trasmissione e di ricezione, si deve selezionare il nodo nel tree e poi cliccare sul pulsante **[Modifica]**.

Nella finestra di dialogo per aggiungere un PDO di trasmissione, cliccare sul pulsante **[TPDO aggiuntivi]**, mentre per aggiungere un PDO di ricezione, cliccare sul pulsante **[RPDO aggiuntivi]**.

La finestra di dialogo per la configurazione di un PDO aggiuntivo è simile alla finestra di dialogo per l'inserimento di oggetti.

In questa finestra si può selezionare l'indice PDO e gli oggetti da associare. Se il PDO ha degli oggetti obbligatori e non modificabili, questi vengono visualizzati e tutti i pulsanti di modifica sono disabilitati. Il nuovo PDO viene aggiunto nel tree in coda agli azionamenti. Per modificare i dati, si deve selezionare e cliccare sul pulsante **[Modifica]**. Per eliminare un PDO aggiuntivo, occorre selezionare il PDO nel tree e di seguito il pulsante **[Elimina]**.

Acquisizione automatica nodi EtherCAT

Se in configurazione hardware è presente il bus EtherCAT, è possibile acquisire dalla rete i nodi collegati, tramite il comando **Acquisizione automatica dei nodi** attivabile da menu contestuale. Affinché il comando possa essere eseguito, i dati della configurazione hardware e i dati presenti sul controllo numerico devono essere allineati e i file ESI che descrivono il nodo sulla rete devono essere presenti nella cartella definita in Tpa.ini nella sezione [tpa] alla voce DirESIFiles.

L'acquisizione dei nodi EtherCAT dalla rete segue le regole elencate:

- in configurazione hardware non è configurato alcun nodo: i dati relativi ai nodi vengono letti dalla rete e visualizzati in configurazione
- in configurazione hardware sono già stati inseriti dei nodi EtherCAT:
 - se il nodo letto dalla rete ha stesso indirizzo, stesso Vendor ID, stesso Product ID e stesso Revision Number del nodo configurato, allora viene mantenuto il nodo configurato con tutti i suoi PDO
 - se il nodo letto dalla rete ha stesso indirizzo del nodo configurato ma diverso Vendor ID o diverso Product ID, allora il nodo della configurazione viene eliminato, insieme al suo virtuale fisico e viene

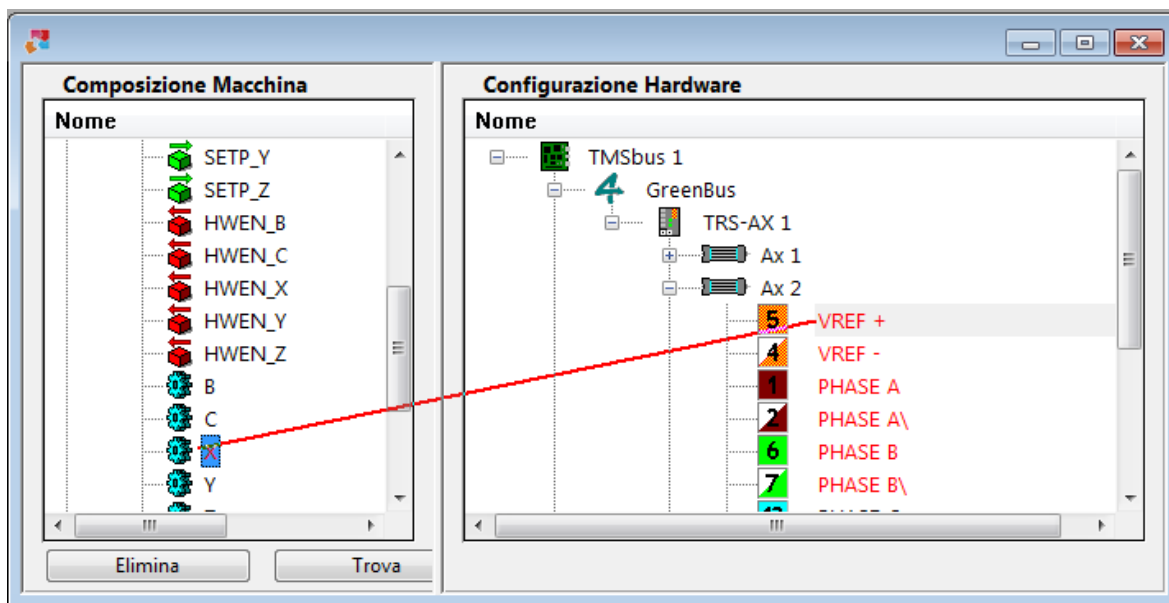
sostituito dal nodo di rete. Se il nodo di rete è un azionamento i PDO vengono letti dal file ESI, altrimenti vengono letti dal nodo di rete.

- se il nodo letto dalla rete ha stesso indirizzo, stesso Vendor ID, stesso Product ID e diverso Revision Number del nodo configurato, allora viene sostituito solo il numero di revision e i PDO vengono mantenuti.
- se il nodo definito in configurazione non ha un nodo corrispondente sulla rete EtherCAT che occupa lo stesso indirizzo, allora viene disabilitato in configurazione e il virtuale fisico viene mantenuto.

7.4.3 Configurazione Virtuale-fisico

La Configurazione Virtuale-Fisico è l'ultima fase della configurazione e consiste nell'abbinare i dispositivi logici ai componenti hardware.

Aperto la Configurazione Virtuale-Fisico appaiono due finestre: a sinistra quella della Composizione Macchina (virtuale), a destra quella della Configurazione Hardware (fisico). In entrambe sono rappresentati in forma grafica e in una struttura ad albero tutti gli elementi che compongono il sistema.



Configurazione Virtuale-Fisico

I collegamenti virtuale-fisico già realizzati sono evidenziati nella finestra "Composizione Macchina", dal Nome del dispositivo in colore rosso, mentre sono evidenziati nella finestra "Configurazione Hardware" dal nome del tipo di segnale, che segue il numero del morsetto, anch'esso in colore rosso.

Per ogni asse di una scheda MECHATROLINK-II sono configurabili in virtuale fisico 6 ingressi e 1 uscita digitale. Si rimanda alla descrizione dettagliata nel capitolo **Linguaggio GPL->Istruzioni->MECHATROLINK-II->MECGETSTATUS**.

Se in un modulo è presente il bus EtherCAT, è comunque possibile configurare schede per il bus MECHATROLINK-II, ma con alcune limitazioni: con real-time a 1 ms non si possono collegare più di sei assi MECHATROLINK-II (per ogni bus); con real-time a 2 ms il limite sale a 16 assi.

I dispositivi o i morsetti ancora da collegare sono in colore nero. In grigio sono rappresentati i dispositivi che appartengono ad un gruppo disabilitato di cui si è voluto mantenere la configurazione in virtuale fisico.

I segnali relativi agli assi, nella finestra "Configurazione Hardware", sono preceduti ciascuno da un rettangolo, il cui colore corrisponde al colore della guaina del filo all'interno del cavo di collegamento.

È possibile evidenziare un collegamento selezionando un dispositivo logico (o un componente hardware) e, premendo la barra spaziatrice, il collegamento viene evidenziato come una linea rossa che unisce il dispositivo al componente hardware. È, inoltre, possibile mantenere sempre attiva la visualizzazione dei collegamenti con il comando **[Alt+Enter]**.

Per visualizzare quale dispositivo logico è collegato al componente hardware, occorre selezionare il componente hardware ed eseguire un doppio click del mouse.

Per selezionare il dispositivo logico e il dispositivo fisico da collegare, sono possibili diverse procedure:

prima procedura

- Rendere visibile sullo schermo, nella finestra "Configurazione Hardware", il morsetto fisico a cui si vuole collegare il dispositivo
- Selezionare o puntare il dispositivo logico desiderato nella finestra "Composizione Macchina"

seconda procedura

- Selezionare o puntare il dispositivo virtuale desiderato nella finestra "Composizione Macchina"
- Selezionare il comando da menu **Modifica->Trova il dispositivo fisico adatto** o la combinazione di tasti **[Ctrl+spazio]**. In automatico Albatros visualizza nella finestra "Configurazione Hardware" il primo dispositivo fisico libero a cui può essere collegato il dispositivo logico.

terza procedura possibile

- Selezionare o puntare un dispositivo virtuale nella finestra "Composizione Macchina"
- Selezionare il comando da menu **Modifica->Trova il successivo dispositivo non collegato** o la combinazione di tasti **[Ctrl+NumPad+]** o il comando **Modifica->Trova il precedente dispositivo non collegato** o la combinazione di tasti **[Ctrl+NumPad-]**.

Per effettuare il collegamento tra i due dispositivi selezionati:

- Sul dispositivo logico da collegare, premere il pulsante sinistro del mouse e, tenendolo premuto, trascinare il mouse nella direzione del morsetto selezionato. Apparirà una riga rossa che evidenzia il collegamento in corso. Una volta giunti sulla riga del morsetto, rilasciare il pulsante per concludere l'operazione; oppure
- selezionare il comando **Collega!** da menu **Modifica** oppure la combinazione di tasti **[Ctrl+L]**.

Per eliminare un collegamento, si deve selezionare il dispositivo o il componente interessato e premere il pulsante **[Elimina]** oppure il tasto **[Canc]** da tastiera.

7.4.4 Mappe di cablaggio

Una volta effettuato il collegamento tra dispositivi virtuali ed i corrispondenti dispositivi fisici, è possibile stampare delle mappe o elenchi, in cui viene riportata l'associazione tra dispositivi fisici e virtuali.

Per poter effettuare questa operazione, è necessario che sul sistema sia installato MS-Word (versione 6 o successiva), Albatros ne sfrutta infatti le funzionalità per formattare le mappe.

Il sistema deve, inoltre, essere configurato correttamente. Ciò significa che devono essere presenti sul sistema i file modello utilizzati per compilare le mappe. Questi sono una serie di file con estensione ".doc", che risiedono normalmente nella cartella System o in un'altra cartella dell'installazione (tipicamente la cartella "Mappe"). Ciò che conta è che la cartella in cui risiedono questi file corrisponda a quella specificata nel file tpa.ini con la chiave "DirMaps". Ad esempio:

```
[TPA]
DirMaps=C:\Albatros\Mappe
```

Per stampare le mappe di cablaggio si deve selezionare un qualsiasi componente hardware, che si trovi nella finestra a destra della [configurazione Virtuale-Fisico](#) o che si trovi nella finestra di [Configurazione hardware](#).

Premendo il pulsante "Stampa" della barra degli strumenti oppure selezionando la voce **Stampa** del menu **File** compare la normale finestra di selezione delle opzioni di stampa. Effettuate le nostre scelte, si confermano premendo il pulsante **[OK]** e comparirà una finestra contenente un elenco dei componenti hardware presenti in configurazione.

In questa finestra possiamo selezionare i componenti per cui vogliamo venga stampata la mappa di cablaggio. Premendo infine il pulsante **[OK]** verranno stampate le mappe di cablaggio. Se si deselecta l'opzione **Stampa su carta**, le mappe verranno salvate come documenti MS-Word nella cartella mappe +nome del modulo corrente (Mod.0, etc.).

Dato l'elevato numero di pagine che possono essere mandate in stampa, si consiglia di effettuare una prova stampando la mappa di un solo componente hardware verificando che tutto funzioni a dovere. Se al posto delle mappe viene stampato un elenco di dispositivi logici, probabilmente non è stato selezionato un componente (ad esempio, una scheda assi o un remoto) nella finestra relativa all'hardware.

7.5 Elenco dei tasti per navigare all'interno di una struttura ad albero

Tasto	Descrizione
Freccia Su	sposta la selezione alla riga immediatamente precedente o successiva
Freccia Giù	

Freccia Destra	esplode di un livello il ramo selezionato e, se già esploso, sposta la selezione sul ramo successivo
Freccia Sinistra	chiude il ramo selezionato e, se già chiuso, sposta la selezione sul ramo precedente
+	esplode di un livello il ramo selezionato
-	chiude il ramo selezionato
*	esplode tutti i livelli del ramo selezionato


8 Strumenti di sviluppo

8.1 Editor GPL

8.1.1 Funzionalità editor GPL

L'editor GPL è lo strumento che permette la creazione e la modifica dei file contenenti il codice GPL di Albatros. Questa funzionalità è attiva solo con un livello di password uguale o superiore a costruttore. Ad ogni file di funzioni sono assegnate delle informazioni visualizzabili da menu **File->Informazioni**.

I modi di funzionamento sono quelli tipici di un editor di testo, per cui troviamo i comandi di **Copia, Incolla, Trova, Sostituisci** etc. Tutti questi comandi sono selezionabili dal menu **Modifica**.

Annulla	Permette, quando possibile, di annullare l'ultima operazione effettuata. Viene ripristinata la situazione precedente all'ultima operazione effettuata.
Ripeti	Ripristina la situazione precedente all'ultimo comando di Annulla
Taglia	Il testo o i dati selezionati vengono eliminati e copiati in una memoria temporanea per consentirne l'eventuale reinserimento col comando <i>Incolla</i> .
Copia	Il testo o l'elemento selezionato viene copiato in una memoria temporanea per poter essere reinserito col comando <i>Incolla</i> .
Incolla	Il contenuto della memoria temporanea viene inserito utilizzando criteri differenti a seconda della funzione attiva.
Elimina	Viene cancellato il testo o le righe o l'elemento selezionato. È possibile recuperare ciò che è stato cancellato agendo immediatamente sul comando <i>Annulla</i> .
Seleziona Tutto	Permette di selezionare tutto il testo del file attivo. Alle righe selezionate possono essere applicati i comandi di Copia, Taglia, Incolla.
Trova...	Ricerca di un testo nel documento corrente. È possibile impostare dei criteri da utilizzare in fase di ricerca: la direzione di ricerca, la distinzione tra caratteri minuscolo e maiuscolo, la ricerca di parola intera e la ricerca tramite espressioni regolari.
Trova Successivo	Consente di ripetere una precedente ricerca, dando la possibilità di modificare i criteri di ricerca impostati con il comando Trova.
Sostituisci	Consente di ricercare del testo nel documento corrente e di sostituirlo con un altro testo.
Inserisci dispositivo	Inserisce un dispositivo selezionandolo dalla lista dei dispositivi. Questa funzionalità è particolarmente comoda quando si lavora con un elevato numero di dispositivi per cui può risultare difficile ricordarne i nomi. Vengono visualizzati solo i dispositivi del modulo corrente che possono essere richiamati e tutti i dispositivi pubblici degli altri moduli
Inserisci funzione	Inserisce una funzione o parte di una funzione, a partire dalla posizione del cursore, leggendola da un file prototipo, scritto dal costruttore della macchina. Possono essere scritti più file prototipo. Il file prototipo è un file di testo, il cui nome deve iniziare con il prefisso GPL e l'estensione deve essere TXT. Deve essere memorizzato nella cartella in cui sono archiviate le librerie (normalmente system\lib). Se vengono definiti più file prototipo, alla selezione del comando viene aperta una finestra di dialogo in cui viene presentato l'elenco dei nomi dei file prototipo senza prefisso e senza estensione. pe I file prototipo possono contenere ad esempio definizioni di costanti di uso comune, funzioni di gestione di errori di sistema, funzioni generiche, codici che implementano algoritmi di uso generico, e così via. Possono contenere anche commenti. Un file prototipo può essere creato salvando il testo selezionato nel file di funzioni GPL. Il comando è disponibile solo come acceleratore di tastiera [Ctrl+Shift+C]. Viene aperta una finestra di dialogo per inserire il nome da attribuire al frammento di codice.
Inserisci messaggio	Inserisce nel testo GPL il codice numerico associato al messaggio scelto. Permette l'inserimento di nuovi messaggi nel file di lingua.
Attiva/disattiva salto pagina	Inserisce o toglie un salto pagina  . Il salto pagina può essere usato come segnalibro per saltare a posizioni notevoli all'interno del file di funzioni.
Vai al salto pagina successivo	Sposta il cursore di edit sulla riga del salto pagina successivo rispetto alla sua posizione
Val al salto pagina precedente	Sposta il cursore di edit sulla riga del salto pagina precedente rispetto alla sua posizione

```

3
4  Function AbsMoveMnt
5      param axisname as axis
6      param speed as float
7      param position as double
8
9      iftarget axisname goto move
10     ifstill  axisname goto move
11     fret
12     move:
13         setvel  axisname, speed
14         movabs  axisname, position
15         waitstill axisname
16         fret
17

```

Editor GPL

La verifica della correttezza della sintassi è eseguita in fase di archiviazione, quando il testo viene anche compilato. È comunque possibile una prima analisi visiva da parte del programmatore, in quanto il testo viene visualizzato in colore differente a seconda di ciò che rappresenta. Per esempio le istruzioni sono colorate in blu, i commenti in verde, le etichette in rosso.

È possibile modificare il valore delle tabulazioni da menu **Opzioni->Tabulazioni...** Si possono definire due tipologie di tabulazioni:

- tabulazioni assolute: impostano la posizione iniziale per le istruzioni del codice GPL, la posizione iniziale del primo argomento delle istruzioni e la posizione iniziale per il commento.
- tabulazione relativa (spazi): imposta a quanti spazi corrisponde una tabulazione

Le tabulazioni rappresentano un aiuto a rendere più comprensibile visivamente il codice GPL.

Per ogni istruzione o parola chiave è disponibile l'help in linea come supporto nella stesura della function. Per richiamare l'help basta posizionare il cursore sull'istruzione e premere il tasto **[F1]**.

Su ogni riga di testo si può scrivere una sola istruzione. È possibile continuare l'istruzione alla riga successiva digitando il carattere '_' (preceduto da uno spazio) come ultimo della riga. Questo permette di inserire dei commenti in mezzo ad un' istruzione:

Message

```

1000 ;codice del messaggio che verrà visualizzato _
3    ;casella di sinottico in cui verrà visualizzato [Enter].

```

Uso delle espressioni regolari

Nella finestra **Trova** e nella finestra **Sostituisci** possono essere utilizzate le espressioni regolari. Una espressione regolare è una sequenza di caratteri numerici e alfanumerici utilizzati per cercare e sostituire parti di testo

all'interno di un testo più grande. Albatros utilizza la grammatica delle espressioni regolari di ECMAScript. In questo paragrafo vengono descritti i principali utilizzi delle espressioni regolari in Albatros.

L'espressione regolare più semplice è quella rappresentata da un singolo carattere. Esistono delle eccezioni, rappresentate dai seguenti caratteri:

- **.** (**punto**): il punto trova qualsiasi carattere. Ad esempio "A.." trova tutte le occorrenze della lettera A maiuscola seguita da due caratteri qualsiasi.
- **[] (parentesi quadre)**: le parentesi quadre permettono di specificare al loro interno un elenco di caratteri. Nel testo vengono cercate occorrenze per ognuno dei caratteri presenti. Se il primo carattere è **^ (accento circonflesso)** verranno cercati tutti i caratteri, eccetto quelli elencati tra le parentesi quadre.

Ad esempio:

[<>]: vengono cercate tutte le occorrenze del carattere < e del carattere >.

[.]AX: vengono cercate tutte le occorrenze che hanno al loro interno la stringa "AX".

[a-d]: vengono cercate tutte le occorrenze dei caratteri a, b, c e d. Il trattino indica un insieme di caratteri.

[\[]]: vengono cercate tutte le occorrenze del carattere [e del carattere] .

[^+]: vengono cercati tutti i caratteri ad eccezione del carattere +.

- ***** (**asterisco**): vengono cercate tutte le occorrenze di un carattere (o gruppi di caratteri) che si trovano prima dell'asterisco. L'asterisco ha effetto solo sul carattere che lo precede: per farlo agire su un gruppo di caratteri, è necessario utilizzare le parentesi tonde.

Ad esempio:

;-*: vengono cercate tutte le occorrenze dei caratteri ; e ;- e ;-----.

- **+** (**segno più**): cerca una o più occorrenze del carattere che lo precede. Differisce dall'uso dell'asterisco, perché il carattere che precede il segno + deve sempre essere presente. Il segno più ha effetto solo sul carattere che lo precede: per farlo agire su un gruppo di caratteri, è necessario utilizzare le parentesi tonde. Riprendiamo l'esempio utilizzato per il carattere asterisco:

;-+: vengono cercate tutte le occorrenze dei caratteri ;- e ;---. Non viene cercato il solo carattere ; (**punto e virgola**).

- **?** (**punto interrogativo**): rende opzionale il carattere che lo precede, che quindi può essere presente al più una volta.

Ad esempio:

Setfeedi?: vengono cercate tutte le occorrenze della parola Setfeed e della parola Setfeedi.

- **{}** (**parentesi graffe**): specificano quante volte un carattere (o un gruppo di caratteri) deve essere presente all'interno del testo.

Ad esempio:

ee{2}: vengono cercate tutte le occorrenze di due ee consecutive.

- **^.** (**accento circonflesso e punto**): trova il primo carattere di ogni riga.
- **^** (**accento circonflesso**): trova il termine cercato solo se si trova all'inizio di una riga
- **|** (**pipe**): trova i termini che compaiono sia prima e sia dopo il carattere '|'.
Ad esempio:
Send|Const: vengono cercate tutte le occorrenze della parola Send e della parola Const.
- **** (**backslash**): il backslash ha una duplice funzione:
 - 1) trasforma il carattere normale in una funzione.
 - \b trova il bordo iniziale o finale della parola
 - \B trova la parola tranne il bordo iniziale
 - \d trova solo le cifre
 - \D trova tutto tranne le cifre
 - \s trova lo spazio
 - \S trova tutto tranne lo spazio

Esempi:

 - \bi trova tutte le parole nel testo che iniziano con la lettera i.

2) trasforma un carattere speciale in un carattere normale. Ad esempio, per ricercare in un testo il carattere speciale ^ (**accento circonflesso**) basta anteporgli il backslash.

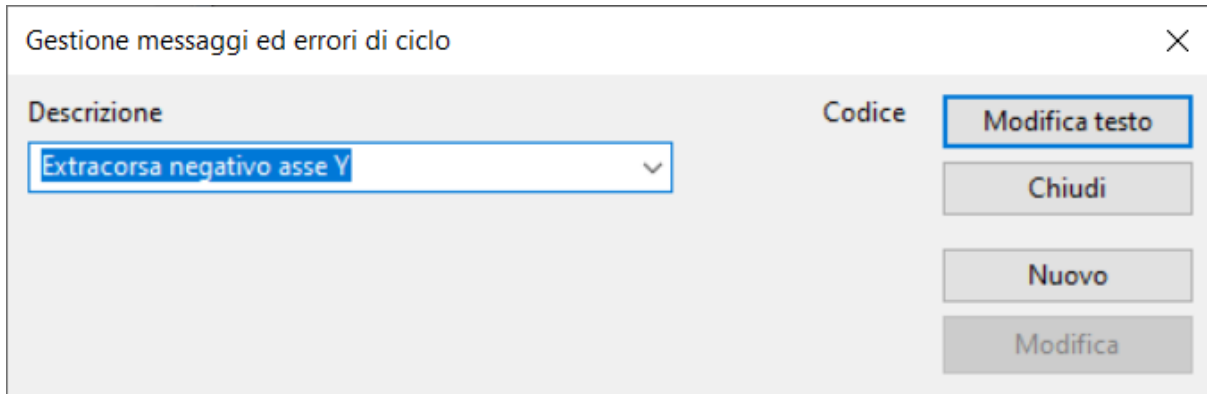
8.1.2 Inserisci messaggio

In Albatros esistono due tipi di messaggi: i messaggi di modulo e i messaggi di gruppo.

Il comando è selezionabile da menu **Modifica->Inserisci messaggio**.

I messaggi di gruppo vengono inseriti direttamente in editor durante la scrittura del codice GPL, attraverso l'istruzione **DEFMSG**. Sono visibili ed utilizzabili solo all'interno del gruppo in cui vengono definiti, per cui è possibile avere la stessa definizione di messaggio in più gruppi, senza che si verifichi alcuna sovrapposizione.

I messaggi di modulo, a differenza di quelli di gruppo, possono essere utilizzati da tutti i gruppi. Possono essere inseriti utilizzando la finestra di dialogo che permette di richiamare un messaggio già presente nel file di lingua o di inserirne uno nuovo.



Finestra di gestione messaggi

In questo modo non ci si deve preoccupare di modificare il file. Il messaggio verrà inserito nella lingua corrente e si dovrà, in seguito, tradurre nelle altre lingue (utilizzando il programma TpaLangs).

Tutti i messaggi presenti nel file di lingua sono elencati alla voce **Descrizione**. Per inserire un messaggio all'interno della funzione, occorre, una volta scelto il testo, selezionare il pulsante **[Modifica testo]**.

Per modificare un messaggio esistente **[Modifica]** o crearne uno nuovo **[Nuovo]**, come prima operazione si deve digitare la modifica o il nuovo testo e premere poi il pulsante corrispondente.

8.1.3 Crittografia

In Albatros è possibile, attraverso la crittografia, rendere impossibile la visualizzazione del testo sorgente delle funzioni.

L'abilitazione della crittografia avviene impostando in tpa.ini la voce Tele+=1. Il default è il valore 0, che sta per crittografia non abilitata.

Quando si salva un file di funzioni e la crittografia è abilitata viene visualizzato il seguente messaggio: "Vuoi crittografare il file?" Se si risponde [Sì] il file viene crittografato. E' possibile crittografare un file precedentemente salvato in chiaro, mentre un file crittografato rimarrà sempre tale.

Con crittografia abilitata, se la password è di tipo Costruttore giornaliera, un file in chiaro non verrà crittografato. Un file di funzioni crittografate potrà essere visualizzato o modificato in Albatros solo dall'utente che lo aveva precedentemente salvato. Il proprietario di un file di funzioni crittografate non può cambiare!

Qualora si volesse togliere la crittografia da un file, si dovrà usare un programma esterno chiamato SBIANCA, che si trova nella cartella Bin di Albatros.

Una modalità di utilizzo del programma sbianca è tramite l'interfaccia grafica. Si selezionano i file che si vuole decrittare. Per ogni file vengono mostrate le proprietà **Stato** e **Credenziali**. Lo **Stato** può essere **In chiaro**, se il file è in chiaro o **Crittato** se il file è crittato. La proprietà **Credenziali** dà informazioni sulla visibilità del file:

Leggibile significa che il file può essere visualizzato dal livello di password corrente, altrimenti **Bloccato**, indicante che il file non può essere visualizzato.

Una volta scelti i file, si dovrà cliccare su **[Decritta!]** per decrittarli o **[Crittografa!]** per crittografarli.

Una seconda modalità di utilizzo è da riga di comando. I comandi e i file sono passati come argomenti al programma. Se non sono presenti argomenti viene attivata la modalità di utilizzo tramite l'interfaccia grafica. La sintassi è:

```
[-l|-e|-d] file1 file2 ...
```

Con:

- -l, o nessuna opzione, per visualizzare lo stato di ogni file
- -d per decrittare i file
- -e per crittare i file

Al termine dell'esecuzione, il programma invia in output i risultati dell'operazione richiesta.

L'output del programma è in formato markdown.

8.1.4 Elenco dei tasti di scelta rapida disponibili

[-] Cancellazione di testo

Tasto

Backspace
Ctrl+Backspace
Del
Ctrl+T
Ctrl+Del

Descrizione

elimina un carattere a sinistra oppure elimina il testo selezionato
elimina la parola a sinistra
elimina un carattere a destra oppure elimina il test selezionato
elimina le parole o gli spazi a destra
elimina la parola a destra e tutti gli spazi successivi fino all'inizio della nuova parola

[-] Commento di più righe di testo

Tasto

Ctrl+';'. Nelle tastiere italiane va premuto anche il tasto [Shift]

Descrizione

aggiunge o toglie il carattere di commento alle righe selezionate.

[-] Posizionamento del cursore

Tasto

Freccia Su
Freccia Giù
Freccia Destra
Freccia Sinistra
Home
End
Ctrl+Home
Ctrl+End
Ctrl+Freccia Sinistra
Ctrl+Freccia Destra
Ctrl+Enter

Descrizione

sposta il cursore nella direzione selezionata

sposta il cursore alternativamente a inizio riga e al primo carattere della riga
sposta il cursore a fine riga
sposta il cursore all'inizio del documento
sposta il cursore alla fine del documento
sposta il cursore di una parola a sinistra
sposta il cursore di una parola a destra
sposta il cursore al primo carattere della riga successiva

[-] Selezione

Tasto

Shift+Home
Ctrl+Shift+Home
Ctrl+Shift+End
Ctrl+Shift+Freccia Sinistra
Ctrl+Shift+Freccia Destra
Shift+Pagina Su

Shift+Pagina Giù

Ctrl+W
Ctrl+A

Descrizione

seleziona fino a inizio riga a partire dalla posizione del cursore
seleziona fino a inizio documento a partire dalla posizione del cursore
seleziona fino a fine documento a partire dalla posizione del cursore
seleziona la parola o gli spazi a sinistra del cursore
seleziona la parola o gli spazi a destra del cursore
seleziona una pagina verso l'alto a partire dalla posizione corrente del cursore
seleziona una pagina verso il basso a partire dalla posizione corrente del cursore
seleziona la parola in corrispondenza della quale è posizionato il cursore
seleziona tutto il documento

[-] Selezione rettangolare

Tasto

Alt+
Shift+Freccia Su
Shift+Freccia Giù
Shift+Freccia Sinistra
Shift+Freccia Destra
oppure
Alt+selezione con il mouse

Descrizione

seleziona un blocco rettangolare di codice

[-] Tabulazioni

Tasto

Tab

Shift+Tab

Descrizione

se non vi è del testo selezionato inserisce la spaziatura tra caratteri definita in **Opzioni->Tabulazioni**. Se sono state selezionate più righe, inserisce a destra la spaziatura impostata per la tabulazione relativa. se non vi è del testo selezionato, sposta il cursore a sinistra della spaziatura definita in **Opzioni->Tabulazioni**. Se sono state selezionate una o più righe, queste vengono spostate a sinistra della spaziatura impostata per la tabulazione relativa.

[-] Copia e Incolla

Tasto

Ctrl+C

Ctrl+Ins

Ctrl+X

Shift+Del

Ctrl+V

Shift+Ins

Ctrl+Y

Drag'n'drop (con mouse)

Ctrl+Drag'n'drop (con mouse)

Descrizione

copia il testo selezionato negli Appunti

elimina il testo selezionato e lo copia negli Appunti

inserisce il contenuto degli Appunti a partire dalla posizione del cursore

elimina la riga su cui è posizionato il cursore e ne copia il contenuto negli Appunti

trascina il testo selezionato e al rilascio lo sposta nella nuova posizione

trascina il testo selezionato e al rilascio lo copia nella nuova posizione

[-] Annulla / Ripristina

Tasto

Ctrl+Z

Alt+BackSpace

Ctrl+Shift+Z

Descrizione

annulla l'ultima digitazione

ripristina l'ultima digitazione

[-] Ricerca e Sostituzione

Tasto

Ctrl+F3

Ctrl+Shift+F3

F3

Shift+F3

Alt+F3

Descrizione

ricerca verso il basso in tutto il documento la parola su cui è posizionato il cursore

ricerca verso l'alto in tutto il documento la parola su cui è posizionato il cursore

ricerca l'occorrenza successiva. La finestra di dialogo **Trova** deve essere chiusa.

ricerca l'occorrenza precedente. La finestra di dialogo **Trova** deve essere chiusa.

apre la finestra di dialogo **Trova** e imposta come testo da ricercare la parola in corrispondenza della quale è posizionato il cursore

[-] Visualizzazione errori di compilazione

Tasto

Doppio click sull'errore

F4

Shift+F4

Descrizione

posiziona il cursore sulla riga della funzione GPL in cui si è verificato l'errore descritto

posiziona il cursore sulla riga della funzione GPL in cui si è verificato l'errore successivo all'ultimo errore selezionato

posiziona il cursore sulla riga della funzione GPL in cui si è verificato l'errore precedente all'ultimo errore selezionato.

[-] Creazione di un file prototipo

Tasto

Ctrl+Shift+C

Descrizione

salva il testo selezionato nel file di funzioni GPL. Viene aperta una finestra di dialogo per inserire il nome da attribuire al frammento di codice.

[-] Gestione del folding

Tasto

Ctrl+M

Descrizione

Apre o chiude il folding selezionato.

8.2 Librerie

Una libreria consiste in una raccolta di funzioni GPL, che possono essere richiamate all'interno del codice GPL custom, senza essere legate ad una particolare configurazione. Le librerie sono molto utili, in quanto possono essere facilmente copiate da una macchina ad un'altra. Si evita di dover riscrivere codice comune quando si implementano nuove macchine. Ad esempio, si può creare una libreria di funzioni geometriche e matematiche.

I file di libreria vengono archiviati nella cartella system\lib. Vengono compilati quando si esegue uno dei seguenti comandi: **CNC->Inizializzazione**, **File->Compila tutto**, **Salva** il file di libreria o il file di variabili globali.

Se nel codice GPL di una macchina viene assegnato un nome di funzione o di variabile che già esiste in una libreria, in fase di compilazione avrà sempre precedenza quello della macchina. Se lo stesso nome esiste in due librerie differenti, per identificare quello che interessa, è bene nel codice GPL utilizzare la seguente sintassi estesa: **nomelibreria.nomefunzione**. Ad esempio, se la funzione LunghezzaSegmento esiste sia nella libreria LIBGEO, che nella libreria LIBMAT e si vuole utilizzare la funzione della libreria LIBGEO, occorre scrivere: LIBGEO.LunghezzaSegmento.

Tutte le operazioni relative ad una libreria sono gestite attraverso una finestra di dialogo. È possibile creare una **[Nuova]** libreria. Il nome assegnato viene riportato nell'elenco delle librerie installate. Si possono importare librerie già esistenti oppure trasformare file di gruppi in librerie, richiamandole attraverso la finestra di dialogo aperta dal pulsante **[Importa...]**. Le librerie eliminate col pulsante **[Elimina]** vengono spostate nel cestino di Windows.

Per modificare il codice di una libreria, si seleziona il pulsante **[Modifica]**. La libreria viene così aperta dall'editor GPL. Durante la stesura delle funzioni di libreria si deve tenere presenti queste regole fondamentali:

- non è possibile accedere a dispositivi, funzioni, variabili, proprie della configurazione all'interno della quale si sta scrivendo la funzione
- è possibile richiamare funzioni e variabili pubbliche di altre librerie
- le funzioni dichiarate all'interno di una libreria vengono di default definite private. Per fare in modo che possano venire richiamate da altri file di funzioni, occorre dichiararle di tipo [PUBLIC](#).

La possibilità di modifica di una libreria è subordinata al livello di accesso di chi sta utilizzando Albatros. È possibile assegnare o modificare le autorizzazioni di accesso ad una libreria selezionando il pulsante **[Proprietà]**.

Le variabili globali dichiarate in una libreria vengono visualizzate in [Diagnostica](#) in una sezione dedicata. La visualizzazione degli elementi della libreria è legata ai diritti di accesso di chi sta utilizzando Albatros.

8.3 Debug

8.3.1 Il debugger

Il debugger è una funzione di Albatros, che permette di seguire passo-passo la sequenza delle istruzioni di un task GPL, permette quindi di capire e correggere eventuali errori logici e comportamenti indesiderati del codice.

Questa funzione è attiva solo con livello di password uguale o superiore a costruttore.

Con il debugger è possibile, per esempio:

- assegnare dei punti di interruzione
- interrompere l'esecuzione di un task e visualizzare il valore di una variabile
- seguire la sequenza di esecuzione di una funzione
- verificare il valore assunto da una variabile locale
- verificare che, nel caso di una istruzione If, il ramo scelto sia quello corretto

I comandi utilizzabili in modalità di debug sono selezionabili dal menu **Debug**. I principali sono:

Continua	Riprende l'esecuzione di un task bloccato. Il task proseguirà finché non termina, non viene bloccato nuovamente o non viene raggiunto un punto di interruzione.
Riavvia	Riavvia il debug del task selezionato.
Interrompi ora	Blocca l'esecuzione del task, di cui si sta effettuando il debug. Il cursore viene posizionato alla riga dove è stata interrotta l'istruzione. Una volta bloccato un task, è possibile pilotarne l'esecuzione e verificare lo stato delle variabili locali
Esegui istruzione	Esegue una singola istruzione GPL. Il task deve esser stato precedentemente interrotto.
Esegui fino a fine funzione	Esegue tutte le istruzioni fino alla prima istruzione successiva alla funzione corrente.

Esegui istruzione/funzione	Esegue una singola istruzione GPL o, se l'istruzione è una chiamata a funzione, esegue l'intera funzione
Esegui fino al cursore	Esegue le istruzioni fino alla posizione del cursore.
Termina debug	Termina l'operatività di debug. Il file di funzioni, di cui si stava eseguendo il debug, viene aperto in modalità di editor.

Per accedere al debug, si deve visualizzare [l'elenco dei task in esecuzione](#) (da menu **Debug-> Task in esecuzione**) o [l'elenco di tutti i task](#) (da menu **Debug->Tutti i task**) e, quindi, selezionare il task di cui si vuole effettuare il debug.

Prima di eseguire il debug, è necessario che non vi siano errori di compilazione delle funzioni (ad esempio: errori di sintassi, variabili non dichiarate) e che il modulo, di cui si vuole effettuare il debug, sia stato inizializzato correttamente.

La finestra di debug è simile a quella dell'editor GPL; tuttavia, in essa non è possibile apportare modifiche al codice. Lo sfondo della finestra è grigio e la linea in esecuzione è evidenziata in colore giallo.

Nota: Non è possibile eseguire contemporaneamente il debug di più task appartenenti allo stesso modulo.










8.3.2 Task in esecuzione

Il comando è selezionabile da menu **Debug->Task in esecuzione**. Visualizza l'elenco dei task in esecuzione associati ad una macchina o modulo. Selezionando un task è possibile [eseguirne il debug](#) o terminarne l'esecuzione, utilizzando rispettivamente i pulsanti **[Debug]** o **[Termina]**.

8.3.3 Tutti i task

Visualizza, in una finestra di dialogo, l'elenco di tutti i task che sono stati definiti nel codice GPL. Graficamente vengono rappresentati in una struttura ad albero. Quando si seleziona una funzione, viene aperto il file in cui essa è definita ed il cursore viene posizionato in corrispondenza della sua prima istruzione. In questo modo è possibile impostare dei [Punti di interruzione](#) ancora prima dell'inizio dell'esecuzione. È possibile eseguire il debug di qualsiasi task e funzione senza parametri.

Di seguito viene descritto il significato di ogni simbolo utilizzato per la composizione dell'albero di esecuzione dei task. Un simbolo particolare è quello che indica la funzione ricorsiva, che sta ad indicare una funzione che richiama al suo interno la funzione da cui è chiamata.

Simbolo	Descrizione
	task della funzione principale dell'Intergruppo
	task autorun
	task generico
	task real-time
	funzione di gruppo
	funzione di gruppo eseguita da istruzioni come ONINPUT, ONFLAG
	funzione di libreria
	funzione di libreria eseguita da istruzioni come ONINPUT, ONFLAG.
	funzione ricorsiva

8.3.4 Chiamate a funzione

Durante il debug è possibile visualizzare l'elenco delle funzioni che sono state chiamate ma che non sono ancora ritornate (ovvero, tutte le funzioni in cui non è stata ancora eseguita l'istruzione FRET). Viene mostrata una finestra di dialogo che elenca le chiamate di funzione che hanno portato fino all'istruzione corrente. In cima alla lista si trova la funzione eseguita più recentemente.

Per osservare il comportamento di una chiamata a funzione:

- spostare il cursore alla posizione desiderata all'interno della funzione.
- scegliere **Debug->Esegui fino al cursore** per portare l'esecuzione del programma alla posizione voluta.
- scegliere **Debug->Chiamate a funzione**, o il tasto scorciatoia Ctrl+K.

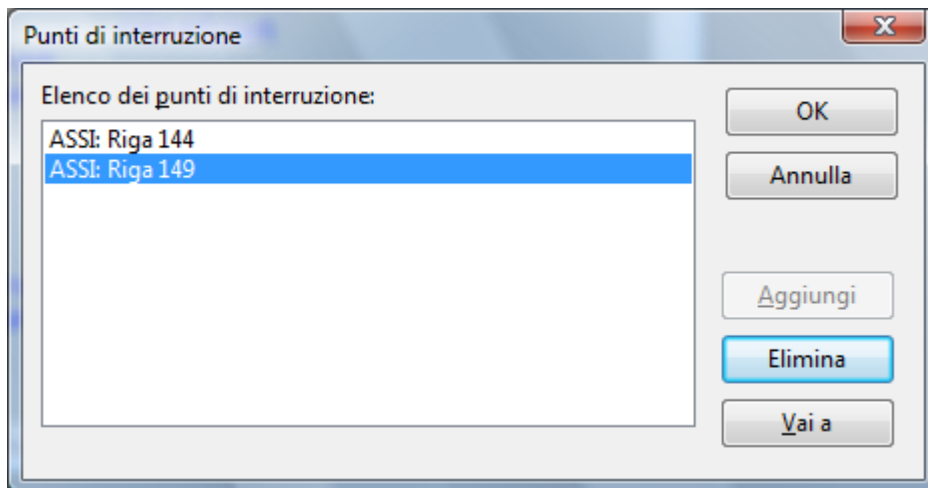
- dalla finestra di dialogo Chiamate si può selezionare il nome di una funzione. Il cursore verrà spostato alla prima istruzione della funzione selezionata.

8.3.5 Punti di interruzione

Un punto di interruzione permette di vedere più nel dettaglio la sequenza di esecuzione delle istruzioni, esaminare o modificare variabili e dispositivi, esaminare l'elenco delle chiamate a funzione, etc.

L'esecuzione di un task si blocca, quando viene raggiunta l'istruzione in cui è stato inserito il punto di interruzione.

I punti di interruzione possono essere impostati sia prima, che un certo task venga messo in esecuzione, sia durante la sua esecuzione (da menu **Debug->Punti di interruzione**). È possibile anche eliminare dei punti di interruzione, una volta che non siano più necessari.



Elenco dei punti di interruzione

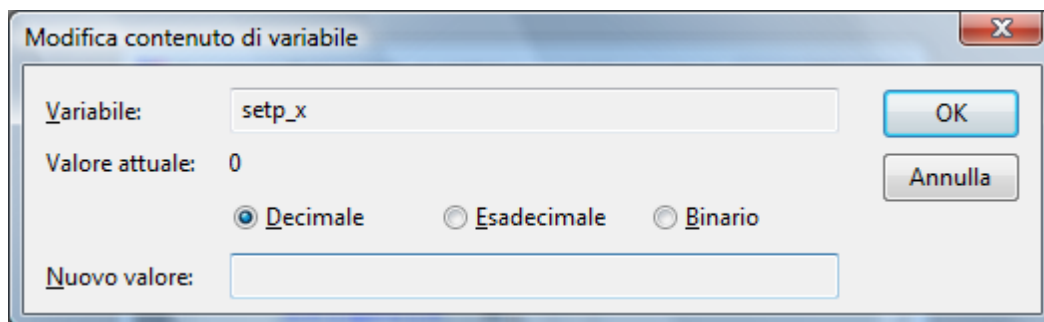
Esistono situazioni in cui, pur avendo inserito punti di interruzione, il task non si blocca, perché l'esecuzione non raggiunge mai il punto di interruzione. In questi casi si può bloccare il task con il comando **Debug->Interrompi ora**. Il cursore viene posizionato sull'istruzione GPL che stava per essere eseguita al momento dell'interruzione.

8.3.6 Contenuto di variabile

Il comando è selezionabile da menu **Debug->Contenuto di variabile**.

Dopo aver interrotto l'esecuzione di un task, è possibile visualizzare:

- il valore delle variabili locali dichiarate nella funzione in cui il task è fermo
- le variabili globali
- il valore assunto da una espressione
- lo stato dei dispositivi e dei parametri di tipo dispositivo.



Visualizzazione/Modifica del contenuto di una variabile

Se la variabile (o il dispositivo) non è di sola lettura, il suo contenuto è modificabile: ovviamente, la modifica influenzerà lo svolgimento successivo del task.

Modificare il valore di una variabile o di un dispositivo permette di provare l'esecuzione in condizioni diverse dall'esecuzione normale, correggere eventuali errori riscontrati e continuare con l'esecuzione delle istruzioni successive.

È possibile visualizzare il contenuto di una variabile, di un dispositivo o di una costante anche spostando il mouse sulla variabile, sul nome del dispositivo o sulla costante. Viene visualizzato un tooltip, in cui viene indicato il tipo, il nome e il valore del dato. Se si seleziona un'espressione viene visualizzato il risultato. Se il cursore del mouse è all'interno di una selezione, viene usata l'intera selezione, altrimenti solo la parola su cui si trova il cursore del mouse. Se il cursore del mouse non è all'interno di una parola, viene usato tutto l'argomento.

Ad esempio, per vedere il valore della matrice `Mx[3][colonna]`, se il cursore del mouse si trova sopra "3" nel tooltip, viene visualizzato 3, se il cursore si trova su "colonna", viene visualizzato il valore di colonna, se è su "matrice", non visualizza niente, se è su una parentesi quadra, visualizza il valore di `Mx[3][colonna]`.

8.3.7 Elenco dei tasti di scelta rapida disponibili

Per attivare i comandi del **Debug**, è possibile selezionare le voci del menu **Debug** oppure agire direttamente da tastiera.

I tasti da utilizzare sono:

Tasto	Descrizione
Ctrl+F5	apre la finestra di dialogo con l'elenco dei task in esecuzione.
Ctrl+Shift+F5	apre la finestra di dialogo con l'elenco di tutti i task.
Ctrl+B	apre la finestra di dialogo per l'inserimento o la cancellazione dei punti di interruzione.
Ctrl+F9	inserisce o elimina un punto di interruzione in corrispondenza della riga su cui è posizionato il cursore.
Ctrl+K	apre la finestra di dialogo per visualizzare l'elenco delle funzioni, che sono state chiamate, ma che non sono ancora ritornate.
Shift+F9	apre la finestra di dialogo per la visualizzazione del contenuto di una variabile.
F8	esegue l'istruzione. Se questa è una funzione, entra nella funzione.
Shift+F7	esegue tutte le istruzioni della funzione.
F10	esegue l'istruzione. Se questa è una funzione, la esegue senza entrarvi.
F7	esegue tutte le istruzioni fino all'istruzione, su cui è posizionato il cursore. Il cursore deve essere posizionato su una istruzione all'interno della funzione.
Alt+Interr	interrompe l'esecuzione del codice all'ultima istruzione eseguita.
F5	riprende l'esecuzione del codice dopo un'interruzione.
Shift+F5	termina il task corrente e lo esegue nuovamente.
Alt+F5	termina il debug.

8.4 Inizializzazione del controllo


8.4.1 Connessioni di rete

Il funzionamento di Albatros in macchina è protetto dalla presenza di una chiave hardware usb, configurata da TPA.

Il comando è selezionabile da menu **Cnc->Connessioni di rete**. Visualizza lo stato dei moduli remoti collegati al sistema. Qualora un modulo non fosse collegato, il simbolo che lo rappresenta viene marcato con una X rossa. Per ogni modulo sono presenti due campi. Il primo campo è il nome del modulo ed il secondo campo è il nome della stazione di rete. Solitamente quest'ultimo ha i primi caratteri a nome fisso "TPA", seguiti dal numero di serie del modulo remoto.

Assegnazione di un nodo di rete (modulo remoto) ad un modulo logico

Si può assegnare un nodo di rete ad un modulo selezionando con il puntatore del mouse la scritta "Non configurato" oppure selezionando il pulsante **[Modifica]**. Dopo alcuni secondi comparirà una finestra con una lista dei moduli remoti disponibili in rete (ogni modulo remoto deve essere acceso e deve aver ottenuto correttamente un indirizzo IP).

È possibile a questo punto selezionare il nodo di rete che si vuole collegare al modulo logico e confermare la scelta premendo il pulsante .

Notare come quest'operazione possa essere svolta con livello di password "Assistenza", senza dover accedere alla configurazione di Sistema di Albatros che richiede, invece, un livello di password "Costruttore".

Aggiornamento del software di un modulo remoto

Si può aggiornare completamente il software del controllo, presente nella memoria interna del modulo remoto, selezionando il pulsante **[Aggiorna]**. Prima di richiedere l'aggiornamento, è indispensabile verificare che il modulo remoto selezionato sia collegato ad Albatros.

8.4.2 Diagnostica hardware

Il comando è selezionabile da menu **Cnc->Diagnostica hardware**. La Diagnostica Hardware visualizza l'elenco e lo stato dei moduli configurati, delle schede e dei nodi ad essi appartenenti, come definiti in configurazione hardware. Se il simbolo della scheda o di un nodo viene marcato con una X rossa, significa che questa non è stata rilevata tra l'hardware presente sul controllo o che non è stato possibile inicializzarla correttamente. Se è marcata con un punto interrogativo giallo, significa che il sistema ha rilevato la presenza di una scheda o nodo, ma che non corrisponde alla tipologia definita in configurazione.

Topologia rete EtherCAT

Nella finestra di diagnostica hardware, quando nel *tree* è selezionato un nodo di una rete EtherCAT, viene abilitato il pulsante **[Dettagli]**, che a sua volta abilita la visualizzazione in formato grafico della topologia della rete EtherCAT.

In questo grafico vengono visualizzate le informazioni dei nodi presenti fisicamente sulla rete: lo stato di ogni nodo, lo stato degli assi, se il nodo è un servozionamento, e la qualità della comunicazione. Ogni nodo e ogni asse sono rappresentati come un rettangolo, il cui colore definisce lo stato.

Passando il cursore del mouse sul rettangolo appare un tooltip che descrive lo stato e gli errori di comunicazione del nodo o lo stato dell'asse.

Visualizzazione e modifica di oggetti nei nodi

Nella finestra di diagnostica hardware, quando nel *tree* è selezionato un nodo di una rete EtherCAT, viene abilitato il pulsante **[Object dictionary]**, per la visualizzazione e la modifica degli oggetti del nodo. La modifica dei dati è possibile solo a livello **Costruttore**.

Gli oggetti sono raggruppati in base al loro indirizzo:

Indirizzo iniziale	Indirizzo finale	Nome Area
0x0000	0x0FFF	Data type area
0x1000	0x1FFF	Communication area
0x2000	0x5FFF	Manufacture specific area
0x6000	0x6FFF	Input area
0x7000	0x7FFF	Output area
0x8000	0x8FFF	Configuration area
0x9000	0x9FFF	Information area

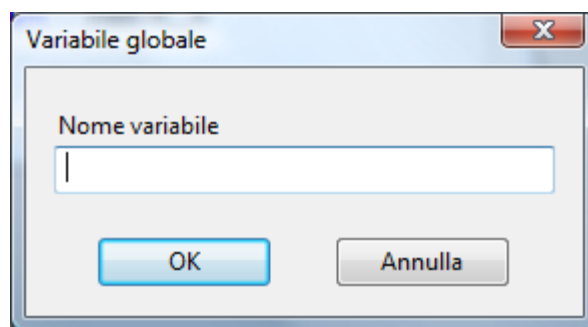
0xA000	0xAFFF	Diagnosis area
0xB000	0xBFFF	Service transfer area
0xC000	0xEFFF	Reserved area
0xF000	0xFFFF	Device area

I valori numerici interi sono visualizzati sia in base decimale che in base esadecimale. Quando si modifica il loro valore, il numero può essere inserito anche in esadecimale, utilizzando la notazione \$...H, e in binario, con \$...B (come si fa in GPL).

8.5 Test

8.5.1 Memorizza variabile globale

Il comando è selezionabile da menu **Test->Memorizza variabile globale**. Memorizza il contenuto di una variabile globale su disco sotto forma di un file di testo formattato. Il file ha nome composto da *nomevariabile.txt* e verrà salvato nella cartella *Report*. L'operazione è possibile solo se il livello di accesso in lettura della variabile globale è compatibile con il livello di accesso corrente.



Memorizzazione di una variabile globale

8.5.2 Esegui funzione

Il comando è selezionabile da menu **Test->Esegui funzione**. Esegue una funzione in modo indipendente rispetto al resto del sistema. Viene, quindi, creato un task che inizia la propria esecuzione dalla funzione selezionata e che da essa prende il nome. Sono eseguibili solo le funzioni che non hanno parametri in ingresso ed il cui livello di accesso in lettura è compatibile con il livello di accesso corrente. Se la funzione che viene eseguita è la funzione principale dell'intergruppo, vengono di seguito eseguiti anche tutti i task autorun.

8.5.3 Importazione messaggi

I messaggi di gruppo assegnati tramite l'istruzione GPL [DEFMSG](#), sono archiviati in un file in formato xmlng. I messaggi contenuti nel file possono essere modificati, aggiunti o eliminati. Per visualizzare le modifiche nel codice GPL si deve utilizzare il comando **Test->Importa messaggi di gruppo**, che è abilitato solo quando non ci sono finestre aperte.

Per poter eseguire un'importazione, tutto il codice GPL deve essere compilato senza errori. Se così non fosse, l'utente verrebbe avvisato con il messaggio "Non tutto il codice GPL risulta compilato". Non vengono importati i messaggi di gruppo appartenenti a file [crittografati](#) (Vedi capitolo **Strumenti di sviluppo->Editor GPL->Crittografia**), per cui l'utente non ha il permesso di visualizzazione in chiaro.

Vengono importati solo i messaggi che sono stati già definiti nel codice GPL. La modifica del testo GPL non viene effettuata, se c'è almeno una DEFMSG successiva a una istruzione IFDEF.

Durante l'importazione possono essere riscontrati degli errori quando:

- tra i testi di un particolare messaggio di gruppo, l'identificativo di una lingua è presente più di una volta
- un qualsiasi testo è vuoto (ovvero: "")
- il nome di un gruppo o di una libreria è definito più di una volta.

Al termine dell'importazione viene eseguita la compilazione di tutti i moduli, in cui sono stati modificati gruppi o librerie.

Tutte le lingue elencate di seguito possono essere utilizzate nei file XMLNG

"AFK" Afrikaans
 "ARA" Arabo
 "AZE" Azero
 "BAS" Baschiro
 "BEL" Bielorusso
 "BGR" Bulgaro
 "BSB" Bosniaco (alfabeto latino)
 "BSC" Bosniaco (alfabeto cirillico)
 "BRE" Bretone
 "CAT" Catalano
 "CHS" Cinese semplificato
 "CHT" Cinese tradizionale
 "COS" Corso
 "CSY" Ceco
 "CYM" Gaelico
 "DAN" Danese
 "DEA" Tedesco (Austria)
 "DEU" Tedesco (Germania)
 "ELL" Greco
 "ENG" Inglese
 "ENU" Inglese (Stati Uniti)
 "ESP" Spagnolo
 "ETI" Estone
 "EUQ" Basco
 "FAR" Persiano
 "FIN" Finlandese
 "FRA" Francese
 "FPO" Filippino
 "FRB" Francese (Belgio)
 "FYN" Frisone
 "GLC" Galiziano
 "HAU" Hausa
 "HEB" Ebraico
 "HRB" Croato (Bosnia-Erzegovina)
 "HRV" Croato (Croazia)
 "HUN" Ungherese
 "IBO" Igbo
 "IND" Indonesiano
 "IRE" Irlandese
 "ISL" Islandese
 "ITA" Italiano
 "JPN" Giapponese
 "KAL" Groenlandese
 "KOR" Coreano
 "SAH" Sakha
 "KYR" Kirghiso
 "LVI" Lettone
 "LTH" Lituano

"LBX" Lussemburghese
"MNN" Mongolo
"NON" Norvegese Nynorsk
"NOR" Norvegese Bokmål
"NLB" Olandese (Belgio)
"NLD" Olandese (Olanda)
"OCI" Occitano
"PLK" Polacco
"PTB" Portoghese (Brasile)
"PTG" Portoghese (Portogallo)
"RMC" Romancio
"ROM" Rumeno
"RUS" Russo
"SKY" Slovacco
"SLV" Sloveno
"SQI" Albanese
"SRM" Serbo (alfabeto latino, Serbia)
"SRN" Serbo (alfabeto cirillico, Bosnia-Erzegovina)
"SRO" Serbo (alfabeto cirillico, Serbia)
"SRP" Serbo (alfabeto latino, Montenegro)
"SRQ" Serbo (alfabeto cirillico, Montenegro)
"SRS" Serbo (alfabeto latino, Bosnia-Erzegovina)
"SVE" Svedese
"TAJ" Tagico
"TRK" Turco
"TTT" Tataro
"TUK" Turkmeno
"UKR" Ucraino
"URD" Urdu
"UZB" Uzbeco
"VIT" Vietnamita
"WOL" Wolof
"XHO" Xhosa
"YOR" Yoruba
"ZUL" Zulu

8.5.4 Nota utente nel file del report degli allarmi

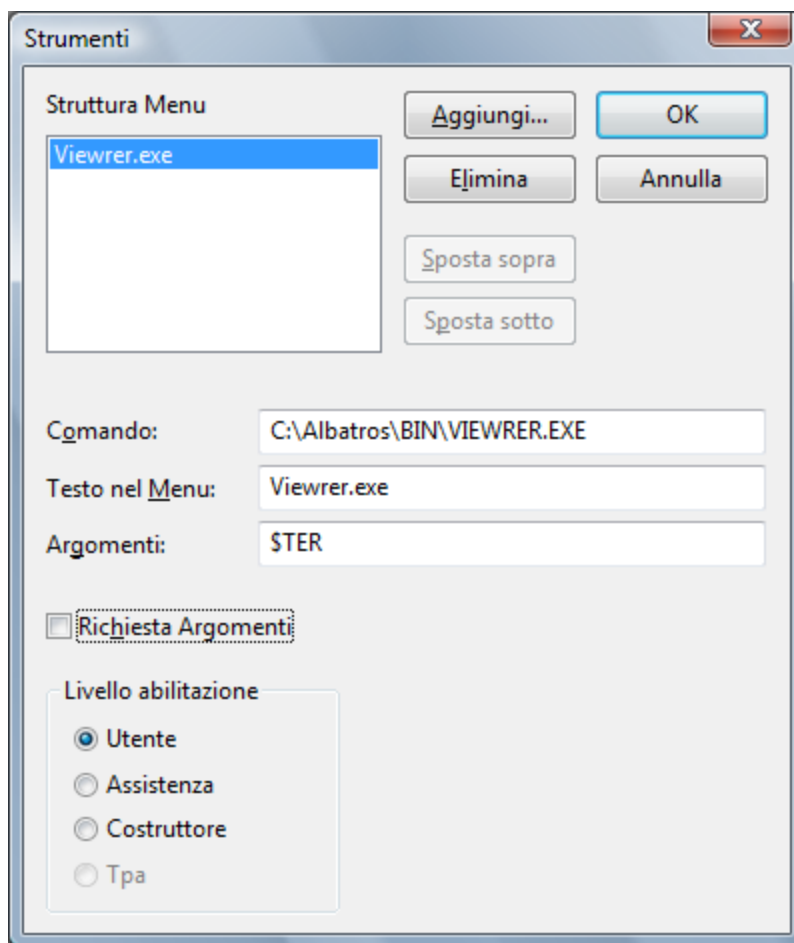
Col comando **Inserisci nota** da menu **Test** è possibile inserire una segnalazione nel file di report degli allarmi del mese corrente (MONTHxx.TER). La voce di menu è abilitata con livello di password uguale o superiore ad assistenza.

8.6 Strumenti

8.6.1 Personalizza...

Il comando è selezionabile da menu **Strumenti->Personalizza...**

Permette l'impostazione di un numero massimo di 20 programmi, la cui esecuzione potrà essere avviata dal menu **Strumenti** di Albatros.

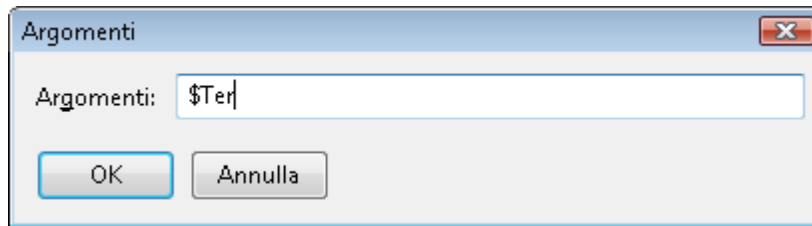


Configurazione del menu Strumenti

- Struttura Menu:** vengono elencati i programmi visualizzati nel menu **Strumenti**.
- Comando:** nome del programma che deve essere eseguito. Può opzionalmente essere definita anche la cartella, in cui è archiviato il programma, qualora fosse diversa dalla cartella da cui è eseguito Albatros o dalle cartelle, in cui il sistema operativo esegue la ricerca dei file eseguibili (variabile di ambiente Windows PATH).
- Testo nel Menu:** è il nome che si vuole far apparire nel menu **Strumenti** per identificare il programma eseguibile.
- Argomenti:** qualsiasi combinazione di argomenti da linea di comando, che il programma necessita per una corretta esecuzione. È possibile inserire argomenti dinamici. \$Ter Ad esempio, utilizzando la stringa \$TER in esecuzione di ViewRER, si apre il file di report del mese corrente.
Ecco la lista degli argomenti:

\$File	Percorso completo del file corrente.
\$FileName	Nome file ed estensione del file corrente.
\$FileDir	Disco e cartella del file corrente.
\$Ter	Percorso completo del file di report errori del mese in corso.
\$DirModule	Disco e cartella contenente MODx del file corrente.
\$Module	Numero modulo del file corrente.
\$Bin	Disco e cartella contenente gli eseguibili di Albatros.
\$TpaIni	Percorso completo del file di inizializzazione tpa.ini
\$ReqDirModule	Percorso (disco e cartelle) del modulo di Albatros. Se sono configurati più moduli, viene aperta la finestra di selezione del modulo.
\$ReqModule	Numero del modulo di Albatros. Se sono configurati più moduli, viene aperta la finestra di selezione del numero di modulo.
\$ReqFile	Nome del file. Viene aperta una finestra di richiesta nome file. Il file selezionato verrà passato tra gli argomenti del programma che verrà eseguito.

Richiesta argomenti: se selezionato, ogni volta che si richiede l'esecuzione del programma, viene presentata una dialog per l'inserimento di argomenti, diversi da quelli impostati alla voce Argomenti, che possono variare a seconda della modalità di lancio del programma.



Specifica argomenti di avvio di un programma

Livello abilitazione: imposta il livello di visualizzazione del programma nel menu **Strumenti**. A programmi di test o di modifica di dati di Albatros normalmente si assegna un livello costruttore. A Programmi per l'editing di lavorazioni di macchina si assegna un livello di abilitazione utente.

L'editing di alcuni campi può avvenire anche attraverso l'utilizzo del pulsante **[Aggiungi...]**. Questo apre la dialog **Aggiungi Strumento** per la selezione del programma che deve essere eseguito. Le tipologie di file eseguibili consentite sono .EXE, .COM, .PIF, .BAT.

Quando si chiude la dialog con conferma dei dati, il programma viene inserito nella finestra **Struttura Menu**, il nome del programma e della cartella a cui appartiene, alla riga **Comando**.

Gli altri pulsanti presenti sono **[Elimina]**, **[Sposta sopra]**, **[Sposta sotto]** e si utilizzano rispettivamente per cancellare un programma e ordinare la lista dei programmi.

8.7 Browser

8.7.1 Il browser

Il browser è una funzione di Albatros che utilizza le informazioni generate dal compilatore per creare un database per la ricerca veloce dei simboli definiti nelle funzioni.

Questa funzione è attiva solo con livello di password uguale o superiore a costruttore. I comandi utilizzabili sono selezionabili dal menu **Debug**.

Con il browser è possibile:

- posizionare il cursore alla riga in cui è definita per la prima volta una funzione, una variabile o una costante di modulo, di gruppo o di libreria. (da menu **Debug->Vai alla definizione**)
- posizionare il cursore alle righe in cui è referenziata una funzione, un dispositivo, una variabile di modulo o di gruppo, una istruzione GPL (ad eccezione delle istruzioni FCALL e FRET). (da menu **Debug->Vai al riferimento**, per visualizzare il riferimento precedente o successivo selezionate da menu rispettivamente le voci **Debug->Precedente** o **Debug->Successivo**)

Le variabili di gruppo sono gestite solo nella finestra di edit del gruppo a cui appartengono.

Per aggiornare un browser, quando si passa ad una nuova versione, si consiglia come prima operazione di salvare le variabili globali e poi di eseguire il comando di **File->Compila Tutto**.

Durante la fase di editing delle funzioni viene persa la corrispondenza tra testo e simboli ricercati e viene ripristinata in fase di archiviazione.

8.7.2 Cerca identificatore...

Il comando è selezionabile da menu **Debug->Cerca identificatore...** Apre una finestra di dialogo per l'inserimento del nome del simbolo da ricercare nel codice GPL. A seconda del **Tipo di ricerca** impostato, viene identificata la definizione o il primo riferimento del simbolo.
















Il nome inserito può avere le seguenti caratteristiche:

- non contiene alcun carattere '.' (punto): viene ricercato in tutti i file di funzioni
- contiene un solo carattere '.' (punto): il nome che precede il punto viene considerato il nome del gruppo ed il simbolo viene ricercato solo in quel gruppo. Ad esempio, se sono definite due funzioni di nome VisError, una nel gruppo MAIN e l'altra nel gruppo ASSI, scrivendo come identificatore ASSI.VisError, il cursore verrà posizionato alla prima riga della funzione VisError del gruppo ASSI.

- contiene due caratteri '.' (punto): il nome che precede il primo punto viene considerato il nome del gruppo, quello che precede il secondo punto viene considerato il nome del sottogruppo ed il simbolo viene ricercato solo in quel sottogruppo.
- se termina con un carattere '*' (asterisco): vengono considerati tutti i simboli che iniziano con i caratteri precedenti l'asterisco.

Nel caso in cui si verifichi una ambiguità nella ricerca del simbolo, viene aperta una finestra di dialogo, dove vengono visualizzati tutti i simboli aventi il nome richiesto. Da qui è possibile scegliere il simbolo voluto.

Di seguito viene descritto il significato dei simboli particolari utilizzati nella lista per la selezione dell'identificatore.

Simbolo	Descrizione
	istruzione GPL
	costante di modulo o di gruppo o di libreria
	variabile di modulo o di gruppo
	variabile di libreria
	vettore di libreria
	matrice di libreria
	funzione di libreria
	messaggio di gruppo
	etichetta
	variabile locale
	vettore locale
	matrice locale
	parametro semplice
	parametro vettore
	parametro matrice

8.7.3 Elenco dei tasti di scelta rapida disponibili

Per attivare i comandi del Browser è possibile selezionare le voci del menu **Debug** oppure agire direttamente da tastiera.

I tasti da utilizzare sono:

Tasto	Descrizione
F2	posiziona il cursore alla riga in cui è definito il simbolo selezionato. Se nel database del browser sono definiti più simboli aventi il nome scelto, viene aperta una finestra di dialogo in cui viene data all'utente la possibilità di scegliere il simbolo richiesto.
Shift+F2	posiziona il cursore al primo riferimento del simbolo selezionato. In caso di ambiguità viene aperta la finestra di dialogo per la selezione del simbolo scelto.
Ctrl+F2	apre una finestra di dialogo per la selezione del simbolo scelto
Ctrl+'+' o Ctrl+PgUp	posiziona il cursore al riferimento successivo (usare '+' del tastierino numerico)
Ctrl+'-' o Ctrl+PgDown	posiziona il cursore al riferimento precedente (usare '-' del tastierino numerico)

9 Programmi accessori

9.1 XConfMerge: programma per il merge di file di configurazione

XConfMerge è un tool che esegue il merge dei file di configurazione. Si esegue da linea di comando dalla cartella bin, in quanto XConfMerge legge il file tpa.ini.

I file letti da XConfMerge sono:

- hardware.xconf: contiene i dati del virtuale fisico e della configurazione hardware (fisici)
- devices.xconf: contiene i dati della configurazione dei gruppi, sottogruppi e dispositivi (logici)
- devices.xmlng: contiene i messaggi traducibili. Vengono in ogni caso considerati tutti i file di lingua presenti nella cartella
- addresses.xdb: contiene gli indirizzi logici dei dispositivi.

Gli argomenti da passare sono:

- la cartella da cui leggere i nuovi file
- il numero del modulo a cui si riferisce il custom. Se non è indicato alcun numero viene considerato 0 come default.

Il percorso da cui leggere il file da aggiornare, che poi è lo stesso in cui scrivere il file ottenuto con la procedura di merge, viene dedotto dai dati impostati in tpa.ini.

Attenzione: i file da aggiornare vengono sovrascritti e non viene eseguito alcun backup automatico.

Regole di merge dei file di configurazione dei gruppi:

1. se in entrambi i file esiste lo stesso gruppo, sottogruppo o dispositivo, si mantengono i dati e l'abilitazione del vecchio file.
2. se il gruppo, sottogruppo o dispositivo esiste solo nel nuovo file, i dati e le abilitazioni vengono copiati.
3. se il gruppo, sottogruppo o dispositivo esiste solo nel vecchio file, questo viene eliminato.

Regole di merge del file di configurazione hardware e del virtuale fisico:

1. se in entrambi i file esiste lo stesso hardware, si mantengono l'hardware e il virtuale fisico del nuovo file con le abilitazioni definite nel vecchio file.
2. se l'hardware esiste solo nel nuovo file, si mantengono il nuovo hardware con la sua abilitazione e il nuovo virtuale fisico.
3. se l'hardware esiste solo nel vecchio file, viene eliminato insieme al virtuale fisico.

Regole di merge del file degli indirizzi logici:

1. se il file non si trova nella cartella dei file da importare, allora vengono mantenuti gli indirizzi logici del custom e vengono assegnati dei nuovi indirizzi ai nuovi dispositivi, se ce ne sono.
2. se il file si trova nella cartella dei file da importare, allora gli indirizzi logici contenuti in esso vengono letti ed utilizzati.

Regole di merge dei file dei messaggi:

1. Non viene eseguito alcun merge, ma il nuovo file dei messaggi viene copiato nella cartella di modulo.

Al termine dell'esecuzione il tool di merge restituisce i seguenti valori:

0	tutto OK
---	----------

1	il merge dei file non è andato a buon fine
2	non sono stati definiti argomenti
3	il numero di modulo passato come argomento è errato
4	la cartella indicata come primo argomento non esiste

9.2 XParMerge: programma per il merge di due file di parametrica

XParMerge è un tool che esegue il merge tra due file di parametrica.

Si esegue da linea di comando dalla cartella bin, in quanto XParMerge legge il file tpa.ini. Gli argomenti da passare sono: il nome del nuovo file e il numero del modulo a cui si riferisce.

Il percorso da cui leggere il file da aggiornare e quello in cui scrivere il file, ottenuto con la procedura di merge, viene dedotto dai dati impostati in tpa.ini.

Attenzione: non viene eseguito alcun backup automatico del vecchio file di parametrica, ma viene sovrascritto.

Regole di merge dei file della Parametrica Tecnologica:

- 1) se in entrambi i file esiste lo stesso controllo si tiene il valore del vecchio file, ma si aggiornano gli altri parametri che lo definiscono (disabilitato, visibile, nome variabile GPL.....) prendendoli dal nuovo file;
- 2) se nel nuovo file è definito un controllo che non esiste nel vecchio file, il controllo viene tenuto;
- 3) se nel vecchio file è definito un controllo che non esiste nel nuovo file, il controllo viene eliminato.

Regole di merge dei file della Parametrica Utensile:

- 1) se la dialog di riferimento del nuovo file è diversa dalla dialog di riferimento del vecchio file, vengono tenute tutte le dialog del nuovo file, se ce ne sono e vengono aggiornate quelle del vecchio file (tolti o aggiunti nuovi controlli);
- 2) se la dialog di riferimento del nuovo file è uguale alla dialog di riferimento del vecchio file, le dialog del nuovo si aggiungono alle dialog del vecchio;
- 3) se la dialog di riferimento è presente solo nel vecchio file, essa stessa e le sue dialog vengono eliminate.

10 Linguaggio GPL

10.1 Concetti base

10.1.1 Introduzione al linguaggio GPL

Il linguaggio GPL (General Purpose Language) è il linguaggio utilizzato per la creazione delle funzioni del sistema Albatros.

La sua struttura è in parte simile a quella del BASIC, ma si caratterizza per la presenza di molte istruzioni dedicate al controllo dei dispositivi.

Il linguaggio è composto da più di 200 istruzioni, che, per comodità, vengono suddivise in gruppi composti da istruzioni aventi funzionalità simili.

Il linguaggio è inoltre [multitasking](#), permette quindi l'esecuzione contemporanea di più task.

Sintassi tipica di un'istruzione GPL

Le istruzioni GPL hanno tutte una struttura simile, che rispecchia il seguente schema:

nomeistruzione parametro-1, parametro-2, parametro-N

Il numero di parametri effettivamente presenti dipenderà dall'istruzione e dal contesto in cui questa viene utilizzata; in generale, il numero massimo di parametri ammesso per una istruzione o funzione GPL è 120. In alcuni casi possono anche non essere presenti parametri.

Il blocco minimo di codice GPL è la [funzione](#).

Suddivisione del codice in gruppi

Il codice GPL viene suddiviso in blocchi che rispecchiano la suddivisione logica della macchina in gruppi. Per ogni gruppo avremo quindi un file contenente il codice ad esso associato. Ai file contenenti il codice dei gruppi presenti in macchina si aggiungono un file di variabili e costanti globali, visibili dal codice GPL di tutti i gruppi, e le [librerie](#). Queste contengono un codice slegato dalla configurazione della macchina e, pertanto, facilmente portabile da una macchina ad un'altra.

10.1.2 Convenzioni e terminologia

Principali termini adottati

ARGOMENTO	È uno degli argomenti dell'istruzione; può essere definito come <i>costante</i> , <i>variabile</i> , o come <i>parametro</i> , a seconda del tipo di istruzione; se è racchiuso tra parentesi quadre ([]), significa che può essere omesso, questo comporta naturalmente un diverso modo di esecuzione dell'istruzione.
PAROLA CHIAVE	È un argomento da scegliere tra argomenti con valore predefinito ed è in genere rappresentato in maiuscolo; l'elenco delle parole chiave è fornito in una pagina di aiuto dedicata.
PARAMETRO	È l'argomento di una istruzione non definito all'interno dell'istruzione stessa, ma che viene passato, come parametro appunto, alla funzione quando viene eseguita; a volte viene denominato anche con il termine di <i>argomento parametrizzato</i> .
COSTANTE	È un argomento definito in modo fisso per mezzo del metacomando CONST, oppure viene fissato rigidamente all'interno dell'istruzione.
VARIABILE	È un argomento che è stato definito come variabile globale di modulo o di gruppo o per mezzo dell'istruzione LOCAL e può essere organizzato come variabile semplice o come vettore o come matrice. Vedere Le variabili .
PARAMETRO DI CONFIGURAZIONE	È un argomento che è stato definito in configurazione, come, ad esempio, i parametri di un asse.

Argomenti che ricorrono maggiormente nella descrizione delle istruzioni

Di seguito sono elencati i termini relativi ad argomenti che ricorrono spesso nella sintassi delle istruzioni GPL, seguiti ciascuno dalla relativa descrizione. Nei casi in cui un argomento può assumere valori diversi da quelli descritti di seguito, la sua descrizione viene ripresa nella sezione *Argomenti* della pagina di aiuto dell'istruzione.

nomeinput	nome di dispositivo di ingresso digitale
nomeoutput	nome di dispositivo di uscita digitale
nomeflag	nome di dispositivo flag switch o flag bit
nomeporto	nome di dispositivo input port, output port o flag port
nometimer	nome di dispositivo timer
nomecounter	nome di dispositivo contatore
nomefunzione	nome di una funzione (vale anche come parametro dispositivo nel caso di ERRSYS.)
nomesottoprogramma	nome di un sottoprogramma, è equivalente ad <i>etichetta</i> a cui occorre fare riferimento per la spiegazione; la chiamata a sottoprogramma si effettua per mezzo dell'istruzione "CALL nome_sottoprogramma".
asse	nome di un asse
costante	un carattere o un numero intero o double oppure una parola chiave
valore	costante o variabile (il <i>tipo</i> dipende dall'istruzione)
variabile	nome di variabile, elemento di vettore o elemento di matrice
variabiledevice	nome di <i>parametro dispositivo</i>
matrice	nome di una matrice
vettore	nome di un vettore
etichetta	nome dell'etichetta di salto o il nome di un sottoprogramma.
stato	stato logico, può essere ON o OFF oppure 1 o 0
timeout	quantità di tempo, entro il quale un certo evento deve verificarsi, o un tempo di ritardo (costante o variabile)
quota	valore della quota (costante double o variabile double)
raggio	valore del raggio (costante double o variabile double)
angolo	valore dell'angolo (costante double o variabile double)
numgiri	numero di giri (costante double o variabile double)
velocità	valore di velocità (costante float o variabile float)
senso	rotazione oraria o antioraria (variabile o costante: CW o CCW)
operando	(costante o variabile o nomedevice)
risultato	risultato dell'operazione (variabile o nomedevice)
nomedevice	nome di un tipo di dispositivo qualunque (o parametro dispositivo)
costantestr	sequenza di caratteri racchiusa tra doppi apici (es. "stringa")
variabilestr	il nome di un vettore di caratteri, ovvero una stringa
operatore	operatori di confronto, anche in combinazione tra loro: > (maggiore) = (uguale a) < (minore) possono essere combinati ponendoli ravvicinati tra loro, es. >= (cioè maggiore o uguale)
tipo	tipo di costante o variabile: "char" (8 bit), "integer" (32 bit), "float" (32 bit), "double" (64 bit), "string"
parametro dispositivo	è una variabile che rappresenta un dispositivo. I dispositivi vengono definiti in Configurazione.

Principali termini utilizzati per un asse

quota teorica (o target)	Posizione corrente "teorica" imposta, istante per istante, dal controllo numerico, in base all'algoritmo di generazione del profilo di velocità.
quota reale	Posizione effettiva dell'asse come rilevato dal trasduttore di posizione. Differisce dalla quota teorica di una quantità denominata "errore di inseguimento" o "errore d'anello".
quota finale	Corrisponde alla quota programmata di arrivo di un movimento. L'algoritmo di calcolo del profilo di velocità porta la quota teorica a raggiungere esattamente il valore finale.

finestra arrivo quota	Intervallo programmabile che ha come punto centrale la quota teorica finale: quando la quota reale si colloca al suo interno, il movimento effettivo viene considerato terminato.
finestra grande di arrivo quota	finestra di arrivo in quota moltiplicato per un fattore impostabile tramite l'istruzione SETBIGWINFACTOR .
errore di inseguimento	Differenza, istante per istante, tra quota teorica e quota reale di un asse: risulta normalmente proporzionale alla velocità di traslazione ed inversamente proporzionale al "guadagno d'anello proporzionale".
guadagno [d'anello] proporzionale	Parametro di regolazione dell'asse, programmabile: determina il rapporto tra la velocità corrente e l'errore di inseguimento relativo.
feed forward	Parametro di regolazione dell'asse, programmabile: determina un contributo diretto (proporzionale alla velocità programmata) iniettato sul comando di velocità dell'azionamento. Permette di ridurre, a pari velocità e pari guadagno proporzionale, il valore dell'errore di inseguimento.
feed rate override	Percentuale della velocità programmata. Questo parametro permette di ridurre la velocità di esecuzione rispetto a quella programmata, in un valore percentuale che può variare da 0% al 100%.
tolleranza	Valore di spostamento per il quale l'asse si discosta dalla traiettoria originale in una interpolazione multiasse a cavallo di due blocchi di spostamento consecutivi.
gioco meccanico	Spazio tra cava e dente di una coppia di ingranaggi

10.1.3 Le variabili

Le variabili sono dei contenitori di informazioni che vengono utilizzati dal linguaggio GPL per depositarvi i valori necessari per lo svolgimento del programma.

Le variabili sono caratterizzate da un "tipo" che rispecchia le caratteristiche dell'informazione, che in esse verrà depositata. Ad ogni variabile è, inoltre, associata una visibilità, che definisce quali insiemi o sottoinsiemi di codice possono operare (leggere e scrivere) su di essa.

Tipi di dato

DATI SEMPLICI o SCALARI

Il GPL supporta tipi di dati semplici ed aggregati. I tipi di dato semplice sono simili a quelli disponibili nella maggior parte dei linguaggi di programmazione:

Char

È un intero con segno compreso nell'intervallo [-128 ; +127] e ha lunghezza pari ad 1 byte.

La dichiarazione di una variabile di tipo char viene effettuata con la seguente sintassi:

```
NomeVariabile as char
```

Integer

È un intero con segno compreso nell'intervallo [-2147483647 ; +2147483647] e ha lunghezza pari a 4 byte (corrisponde al tipo long del C).

La dichiarazione di una variabile di tipo Integer viene effettuata con la seguente sintassi:

```
NomeVariabile as integer
```

Float

È un numero in virgola mobile compreso nell'intervallo [-3,402 E+38 ; +3,402 E+38], ha lunghezza pari a 4 byte (utilizzato tipicamente per rappresentare le velocità).

La dichiarazione di una variabile di tipo Float viene effettuata con la seguente sintassi:

```
NomeVariabile as float
```

Double È un numero in virgola mobile compreso nell'intervallo [-1,797 E+308 ; 1,797 E+308], ha lunghezza pari a 8 byte (utilizzato tipicamente per rappresentare le quote)

La dichiarazione di una variabile di tipo Double viene effettuata con la seguente sintassi:

NomeVariabile as double

Questi tipi di dati possono essere utilizzati contemporaneamente in una stessa espressione. Il GPL esegue una conversione automatica senza dare alcun messaggio di avvertimento. Si deve, quindi, fare attenzione alla possibile perdita di informazione che si può verificare, quando si utilizzano tipi di dati diversi nella stessa espressione.

In alcune situazioni la conversione non è consentita. In questi casi si ottengono, in genere, degli avvertimenti dal compilatore o degli errori di sistema.

DATI AGGREGATI

L'array

È un insieme di variabili di tipo semplice, tutte dello stesso tipo, ottenuto associando un indice al nome della variabile. L'indice viene racchiuso tra parentesi quadre. Se l'array si chiama, ad esempio, "parametri", il primo elemento dell'insieme verrà indicato con "parametri[1]", il secondo con "parametri[2]" e così via.

L'array ha un numero fisso di elementi, che deve essere definito all'atto della dichiarazione. Una tipica dichiarazione di un array segue la sintassi:

```
parametri[10] as integer
```

Dove *parametri[10]* specifica che il nome dell'array è "parametri" e che è composto da 10 elementi, *as integer* definisce il tipo di dato semplice dei singoli elementi dell'array, che in questo caso è integer.

Gli array possono essere costituiti da dati di tipo semplice o da stringhe.

Un array può essere composto da un massimo di 262144 elementi.

I vettori possono essere inizializzati direttamente nel codice GPL al momento della loro dichiarazione. La sintassi GPL può essere:

```
[READONLY] vettore[numerorighe] as integer = 1,2,3,4
[READONLY] vettore[numerorighe] as string = "uno","due","tre","quattro"
```

Le matrici

Le matrici sono degli array bidimensionali ovvero delle variabili cui sono assegnati due indici. Possiamo associare una matrice ad una tabella di dati suddivisa in righe e colonne. Per indicare una delle caselle della tabella possiamo definire in quale riga e in quale colonna della tabella essa si trova. Il primo indice della matrice corrisponde al numero di riga, il secondo al numero di colonna.

A differenza degli array le matrici possono contenere tipi di dati differenti tra loro, ma con la seguente limitazione: possiamo usare un tipo di dati semplice, diverso per ogni colonna, ma non possiamo variare il tipo all'interno della colonna stessa.

Ad esempio, possiamo definire una matrice, in cui la prima colonna è di tipo integer e la seconda è di tipo float. Non possiamo però avere una matrice, in cui nella prima riga abbiamo un integer ed un float e nella seconda riga abbiamo un char ed un double. Le righe devono essere tutte uguali per quanto riguarda il tipo di dati degli elementi che le compongono.

La dichiarazione di una matrice può essere fatta secondo una delle seguenti sintassi:

```
offset[10] as double double double
```

```
dim_pezzo[50] as float:lunghezza float:larghezza float:spessore
```

Nel secondo tipo di dichiarazione viene assegnata un'etichetta o nome simbolico a ciascuna colonna. I nomi simbolici delle colonne risultano molto utili quando si lavora con matrici di grosse dimensioni. In queste situazioni è infatti difficile ricordare il significato delle grandezze memorizzate all'interno di ogni colonna della matrice. Il nome simbolico ci permette di identificare immediatamente il tipo di dato con cui stiamo lavorando. Ad es. "Offset[1][3]" è meno chiaro di "Offset[1].asse_X".

Le matrici possono contenere solo dati di tipo semplice. Non si possono, ad esempio, creare matrici che contengono delle stringhe. Una matrice può essere composta da un massimo di 262144 righe.

Le matrici possono essere inizializzate direttamente nel codice GPL, al momento della loro dichiarazione. La sintassi GPL può essere:

```
[READONLY] nomematrice[numerorighe] as double double integer double = _
  1.1, 2.2, 3, 0.1 _
  1.2, 3.4, 5, 0.1 _
  2.1, 5.6, 6, 0.1
```

Le stringhe

Le stringhe sono degli insiemi di caratteri ovvero di dati tipo char, che però vengono gestiti in modo particolare, in quanto rappresentano del testo leggibile.

Una stringa è molto simile ad un array di char. La maggiore differenza è data dalla presenza di un carattere terminatore, che viene aggiunto automaticamente in coda alle stringhe. Il GPL fornisce, inoltre, alcune istruzioni che permettono di manipolare le stringhe.

Tipicamente le stringhe vengono utilizzate per scrivere dei messaggi, leggibili dall'utente, sullo schermo o in un file di report.

La dichiarazione di una variabile di tipo Stringa può essere effettuata secondo una delle seguenti sintassi:

```
NomeVariabile as String
```

```
NomeVariabile as String[20]
```

Nel primo tipo di dichiarazione viene assegnata una dimensione di default di 256 byte. Nel secondo caso si definisce una dimensione massima per la stringa.

I valori stringa sono sequenze di caratteri delimitate da doppi apici. Esempio: "Premere il pulsante".

Per inserire il carattere "" (doppio apice), bisogna inserirlo due volte. Esempio: "Premere il pulsante ""Start""".

Per inserire caratteri, tramite il codice numerico, si scrivono all'interno della stringa i caratteri \u, seguiti dal valore numerico del carattere esadecimale. Esempio: \u20ac è il simbolo dell'euro. Se si scrive "\u20ac 15,6" si ottiene € 15,6.

Conversione dati

In tutte le espressioni matematiche, ad eccezione della istruzione EXPR, i tipi di dato degli operandi vengono convertiti al tipo di dato della variabile risultato e poi viene eseguita l'operazione. È importante porre molta attenzione alla dichiarazione dei tipi di dato, perché possono influenzare il risultato. La tabella seguente è un esempio di come possono variare i risultati in base al tipo di dato:

DIV	Operando 1 (Integer)	Operando 2 (Double)	Risultato (char)
	3	5.0	0
	5	1.9	5
	1200	107.2	Indefinito
	1200	250.0	Indefinito
DIV	Operando 1 (Double)	Operando 2 (Double)	Risultato (Double)
	3	5.0	0.6
	5	1.9	2.631
	1200	107.2	11.194
	1200	250.0	4.8

Nell'istruzione EXPR se gli operandi non sono tutti dello stesso tipo, viene effettuata una conversione automatica e il tipo del risultato dell'operazione è lo stesso di quello maggiore tra i due in base alla seguente regola:

- char < integer
- float < double
- char o integer < float o double.

Risolta l'espressione il risultato viene convertito al tipo della variabile risultato.

EXPR	Operando 1 (Double)	+	Operando 2 (Integer)	/	Operando 3 (Float)	Risultato (Integer)
	900.0	+	100	/	400.0	900
EXPR	Operando 1 (Double)	+	Operando 2 (Integer)	/	Operando 3 (Float)	Risultato (Double)
	900.0	+	100	/	400.0	900.25

Dichiarazione e visibilità delle variabili

Le dichiarazioni delle variabili e delle costanti possono essere fatte solo in alcuni punti particolari del codice GPL.

Si possono definire variabili:

- Globali di modulo
- Globali di gruppo
- Locali (solo variabili)
- Globali di libreria

Il numero massimo di variabili globali (di modulo e di gruppo) dichiarabili è 2048.

Si possono definire dei *modificatori*, che attribuiscono alle variabili delle caratteristiche aggiuntive.

Variabili globali di modulo

Le variabili globali di modulo sono contenute in un particolare file cui si accede selezionando la voce di menu

File->Apri Variabili globali.

La dichiarazione viene effettuata, come si è visto nei paragrafi precedenti, specificando il nome della variabile, seguito dalla parola chiave "AS", seguita dal tipo di dato (o dai tipi di dato nel caso delle matrici).

Queste variabili sono visibili direttamente dal codice di tutti i gruppi.

Variabili globali di gruppo

Le variabili globali di gruppo vengono definite all'inizio del codice relativo al gruppo. Devono essere dichiarate prima delle funzioni GPL.

Queste variabili sono visibili direttamente da tutto il codice contenuto all'interno del gruppo. Inoltre è possibile estendere la visibilità di queste variabili all'esterno del gruppo dichiarandole come variabili "Pubbliche".

Le variabili pubbliche non sono accessibili direttamente dall'esterno del gruppo. Per potervi accedere si deve utilizzare il loro nome preceduto dal nome del gruppo cui appartengono. Ad esempio se si vuole modificare la variabile pubblica "offset" del gruppo "assi" dal codice del gruppo "main" si dovrà scrivere "SETVAL 10 assi.offset".

Per dichiarare una variabile globale di gruppo si utilizza la stessa sintassi utilizzata per le variabili globali di modulo. La principale differenza sta nella definizione delle variabili pubbliche.

Per definire una o più variabili pubbliche e private si utilizzano le etichette "Public:" e "Private:". Ad esempio:

```
Public:
  offset as double
  velocità as float
Private:
  utensile as integer
```

Variabili locali

Le variabili locali vengono dichiarate all'interno del corpo di una funzione. Devono essere dichiarate prima di qualsiasi altra istruzione ad eccezione della dichiarazione dei parametri della funzione.

Le variabili locali sono accessibili solo all'interno della funzione stessa.

Queste variabili vengono create con valore 0 (viene allocata la memoria necessaria) solo all'inizio dell'esecuzione della funzione stessa e vengono distrutte (la memoria viene rilasciata) al termine dell'esecuzione. Le variabili globali vengono invece create all'inizializzazione del modulo e sono sempre visibili nella "Diagnostica".

La dichiarazione di una variabile locale segue la sintassi già vista, ma deve essere preceduta dalla parola chiave "LOCAL".

Ad esempio:

```
Function lavorazione
  local quota_centro as double
  movabs X,quota_centro
fret
```

Variabili globali di libreria

Le variabili globali di libreria sono contenute nelle [librerie di codice GPL](#). Sono analoghe alle variabili globali di gruppo.

Modificatori

Modificatori: READONLY

Le variabili globali di modulo e globali di gruppo possono essere dichiarate come READONLY.

Una variabile readonly è una variabile il cui valore non può essere modificato da codice GPL, ma può essere modificato "dall'esterno" ovvero dall'archivio dei parametri tecnologici di Albatros.

L'archivio dei parametri tecnologici è un database in cui vengono conservati dei valori che caratterizzano la macchina ma che possono variare nel lungo periodo in seguito a modifiche della macchina stessa o a manutenzione straordinaria. Queste informazioni vengono normalmente inserite in una matrice GPL durante l'inizializzazione del controllo.

Esempi di questo tipo di informazioni sono gli offset delle aree di lavorazione o le dimensioni e i parametri tecnologici degli utensili.

Dichiarando queste variabili come readonly, ci mettiamo al riparo da modifiche accidentali di informazioni che in genere non devono variare durante il normale funzionamento della macchina.

La dimensione massima di una variabile readonly è di 128 Kbyte.

La dichiarazione di una variabile readonly va effettuata secondo la seguente sintassi:

```
readonly NomeVariabile as tipo
```

Modificatori: NONVOLATILE

Le variabili dichiarate **NONVOLATILE** vengono memorizzate su RAM non volatile (dotata di batteria), anziché su RAM normale. Di conseguenza i valori immagazzinati in queste variabili non vengono persi allo spegnimento del controllo numerico.

La dichiarazione di una variabile nonvolatile va effettuata secondo la seguente sintassi:

```
nonvolatile NomeVariabile as tipo
```

Ad esempio:

```
nonvolatile OffsetAree[2] as double:offsetX double:offsetY double:offsetZ
```

Possono essere definite "nonvolatile" solo le variabili globali di gruppo e di modulo.

La dimensione massima complessiva delle variabili memorizzate su RAM non volatile è di 65536 byte. La dimensione massima di una singola matrice non volatile è di 1024 byte.

Assegnazione di RANGE

È possibile all'atto della dichiarazione di una variabile assegnarle un range di valori. Al momento non viene, tuttavia, effettuato un controllo del rispetto dei limiti in fase di esecuzione; viene solamente effettuato un controllo dal compilatore, nel caso in cui vengano assegnati dei valori costanti (ad es. per inizializzare la variabile).

Il principale vantaggio è, quindi, una sorta di auto documentazione del codice.

La definizione dei range si effettua secondo la seguente sintassi:

```
NomeVariabile Range: valmin..valmax AS tipo
```

Ad esempio:

```
NumeroUtensile Range: 1..100 as integer
```

Diritti di Lettura / Scrittura

I diritti di lettura e scrittura permettono di specificare il [livello minimo di accesso](#) al sistema necessario per visualizzarne (diritto di lettura) e modificarne (diritto di scrittura) il valore.

La sintassi è la seguente:

```
NomeVariabile Read=S Write=M AS tipo
```

Le parole chiave utilizzate per specificare i diritti sono:

- READlettura
- WRITE scrittura

I valori assegnabili sono:

- U o USER utente
- S o SERVICE assistenza
- M o MANUFACTURER costruttore
- T o TPA TPA

I default dei valori sono:

- READlettura per assistenza (S o SERVICE)
- WRITE scrittura per costruttore (M o MANUFACTURER) e Tpa (T o TPA)

10.1.4 Le costanti

Il GPL prevede quattro tipi di costanti:

- Integer
- Double
- Char
- String

Le costanti di tipo Char vengono dichiarate utilizzando gli apici come segue:

```
Const COD = 'A'
```

Le costanti di tipo String vengono dichiarate utilizzando i doppi apici come segue:

```
Const MSG = "Inizio lavorazione"
```

Le costanti di tipo Integer e di tipo Double vengono dichiarate con la seguente sintassi:

```
Const PI = 3.14  
Const MSGBOX = 12
```

Per le costanti di tipo Integer è prevista anche una notazione binaria ed esadecimale:

```
Const MASK = $11001001b ; binario  
Const MASK = $F5h ; esadecimale
```

Anche le costanti di gruppo e libreria possono essere pubbliche o private. La sintassi è analoga a quella vista per le variabili.

Esempio:

```
Public:  
Const PI = 3.14  
Const MSGBOX = 12  
Private:  
Const MASK = $11001001b
```

NOTA: non esistono costanti di tipo Float. Valori decimali dovranno necessariamente essere dichiarati come Double. A volte ciò può causare dei messaggi di avvertimento da parte del compilatore (quando si utilizzano istruzioni GPL ottimizzate per l'utilizzo di tipi Float).

È possibile definire le costanti come risultato di espressioni di calcolo, con la seguente sintassi:

```
Const a = 10  
Const b = 20  
Const c = a + b
```

Gli operatori ammessi sono gli stessi utilizzati nell'istruzione [EXPR](#).

Costanti predefinite con valore preassegnato

Il linguaggio GPL prevede alcune costanti predefinite. Queste possono essere utilizzate direttamente senza necessità di essere definite.

Le costanti predefinite ed i valori corrispondenti sono:

ON	1
OFF	0
UP	+1
DOWN	-1
POSITIVE	+1
NEGATIVE	-1
CW	1
CCW	0
TRUE	1

FALSE	0
NOWAIT	0
WAIT	1
WAITACK	2
STORE	1
NOSTORE	0
NOPLACE	0
COM1	0
COM2	1
COM3	2
COM4	3
COM5	4
COM6	5
COM7	6
COM8	7
NOPARITY	0
ODDPARITY	1
EVENPARITY	2

Costanti predefinite con valore preassegnato in avvio di Albatros

Il linguaggio GPL prevede alcune costanti predefinite il cui valore viene definito all'avvio di Albatros. Possono essere utilizzate nell'istruzione [IFDEF](#).

_ID_MODULE	numero del modulo corrente. Il numero del modulo è compreso tra 0 e 15.
_REMOTE_MODULE	tipo di modulo. La costante vale 1 se il modulo è remoto, vale 0 se il modulo è locale, non è definita se il modulo non è configurato in Configurazione di sistema.
_VER_MAJOR	numero di versione principale di Albatros. Se la versione di Albatros è 3.2.1, il valore della versione principale è 3.
_VER_MINOR	numero di versione secondario di Albatros. Se la versione di Albatros è 3.2.1, il valore della versione secondario è 2.
_VER_REVISION	numero di revisione di Albatros. Se la versione di Albatros è 3.2.1, il valore della revisione è 1.
_VER_SP	stringa che descrive il Service Pack se installato, ad esempio "Service Pack 1f", altrimenti non è definita.
_VER_FULL	numero di versione completa. Nel caso di versione 3.2.1 vale \$00030201H.

10.1.5 Parole chiave

Le parole chiave sono identificatori con uso riservato e non possono essere utilizzati in altro modo.

Le parole chiave disponibili sono:

Tutti i nomi delle istruzioni GPL	Si rimanda alla parte del manuale "Istruzioni " per la descrizione di tutte le istruzioni GPL
Tutti i tipi di dato	Vedere Le variabili
I parametri di tipo dispositivo	Vedere I parametri di tipo dispositivo
EXIST	utilizzata nell'istruzione IFDEF per verificare esistenza di un gruppo. Vedere l'istruzione IFDEF
NOTEXIST	utilizzata nell'istruzione IFDEF per verificare la non esistenza di un gruppo. Vedere l'istruzione IFDEF
LINKED	utilizzata nell'istruzione IFDEF per abilitare la compilazione di blocchi di codice se il dispositivo è collegato in virtuale-fisico. Vedere l'istruzione IFDEF
UNLINKED	utilizzata nell'istruzione IFDEF per abilitare la compilazione di blocchi di codice se il dispositivo non è collegato in virtuale-fisico. Vedere l'istruzione IFDEF
FUNCTION	Dichiarazione di una funzione. Vedere Le funzioni
AS	Utilizzata per la dichiarazione delle variabili. Vedere Le variabili
PUBLIC	È un attributo di una funzione. Vedere Le funzioni
AUTORUN	È un attributo di una funzione. Indica che la funzione è a partenza automatica. Vedere Le funzioni
R= o READ	È un attributo di una funzione o di una variabile. Segnala il livello di accesso in lettura. Vedere Le funzioni e Le variabili e Diritti di accesso
W=o WRITE	È un attributo di una funzione o di una variabile. Segnala il livello di accesso in scrittura. Vedere Le funzioni e Le variabili e Diritti di accesso
CONST	Permette di assegnare un nome significativo, chiamato costante simbolica, al posto di un numero, di un carattere o di una stringa. Vedere Le variabili
READONLY	È un attributo di una variabile globale. Vedere Le variabili
NONVOLATILE	È un attributo di una variabile globale. Vedere Le variabili
PRIVATE	È un attributo di una funzione. Vedere Le funzioni
RANGE	Utilizzata per la definizione di un intervallo di valori per una variabile. Vedere Le variabili
USER	È un attributo di una funzione o di una variabile. Segnala la tipologia di accesso, in questo caso utente. Vedere Le funzioni o Le variabili .
SERVICE	È un attributo di una funzione o di una variabile. Segnala la tipologia di accesso, in questo caso manutenzione. Vedere Le funzioni o Le variabili .
MANUFACTURER	È un attributo di una funzione o di una variabile. Segnala la tipologia di accesso, in questo caso costruttore. Vedere Le funzioni o Le variabili .
TPA	È un attributo di una funzione o di una variabile. Segnala la tipologia di accesso, in questo caso TPA. Vedere Le funzioni o Le variabili .

10.1.6 Le funzioni

Le funzioni sono il blocco minimo di codice GPL. Le istruzioni GPL non possono essere inserite sequenzialmente in un file, ma vanno raggruppate in funzioni. Il numero massimo di funzioni dichiarabili è 8191.

Dal punto di vista del compilatore sono funzioni tutti i blocchi di codice GPL, che iniziano con una riga, la cui prima parola è FUNCTION. Non c'è tuttavia una parola chiave che indica la fine del testo di una funzione: la funzione termina con la riga che precede l'inizio di un'altra funzione o con la fine del file che contiene le funzioni.

La sintassi per la definizione di una funzione è:

```
FUNCTION      NomeFunzione Attributi
                Parametri
                Variabili locali
                Elenco delle istruzioni GPL
```

Una funzione è, inoltre, un particolare tipo di dispositivo di Albatros. Condivide con i dispositivi alcune proprietà: un nome univoco (non traducibile), un indicatore di visibilità (se il dispositivo è pubblico o meno), un [livello di accesso](#) in lettura e un livello di accesso in scrittura (vedi il paragrafo seguente).

Diritti di accesso

I diritti di accesso permettono di specificare il livello di accesso minimo al sistema che ne consenta la visibilità (diritto di lettura) e l'esecuzione (diritto di scrittura).

La sintassi è la seguente:

```
Function      NomeFunzione      READ=S WRITE=M
```

I diritti sono identificati dalle parole chiave READ (lettura) e WRITE (esecuzione)

I valori associabili, che corrispondono ai livelli di accesso, sono:

U o USER	utente
S o SERVICE	assistenza
M o MANUFACTURER	costruttore
T o TPA	tpa

I default dei valori sono:

READ	lettura per assistenza (S o SERVICE) e utente (U o USER)
WRITE	scrittura per costruttore (M o MANUFACTURER) e tpa (T o TPA)

Funzioni Autorun

Una funzione di tipo autorun viene eseguita automaticamente all'inizializzazione della macchina.

Le funzioni autorun hanno la caratteristica di essere rilanciate automaticamente, quando terminano, in seguito ad un errore di sistema.

La sintassi è la seguente:

```
Function      NomeFunzione      autorun
```

Basta, quindi, apporre il modificatore "autorun" alla dichiarazione della funzione.

Funzioni Public

Una funzione normalmente può essere mandata in esecuzione (chiamata), solamente dal codice che risiede all'interno del file di gruppo. Per fare in modo che una funzione possa essere mandata in esecuzione dal codice GPL di un altro gruppo, la funzione deve essere definita di tipo **public**. La sintassi per definire una funzione public è la seguente:

```
Function      NomeFunzione      public
```

Basta, quindi, apporre il modificatore "public" alla dichiarazione della funzione.

Fanno eccezione le funzioni che appartengono all'intergruppo, le quali sono sempre di tipo **public**.

Funzioni di Sottogruppo

Una funzione può essere associata ad un sottogruppo, semplicemente antepoendo il nome del sottogruppo a quello della funzione. Il nome del sottogruppo e quello della funzione devono essere separati da un punto ".". Ad esempio, la seguente funzione appartiene al sottogruppo "X" del gruppo "Assi".

```
Function      X.azzeramento
local        vel as float
movabs      X,100
```

```
waitstill X
Fret
```

Funzioni asincrone

Una funzione asincrona viene richiamata automaticamente dal controllo numerico, quando si genera l'evento a cui è legata la funzione.

Le tipologie di eventi sono tre:

- Cambiamento di stato di un input digitale: istruzione ONINPUT
- Cambiamento di stato di un flag bit o un flag switch: istruzione ONFLAG
- Generazione di un errore di sistema: istruzione ONERRSYS

Quando si genera l'evento, viene chiamata la funzione (non come task autonomo, bensì nel contesto del task, in cui è stata eseguita l'istruzione ON... corrispondente) come FCALL implicita, subito dopo che l'istruzione corrente ha terminato l'esecuzione.

Tipicamente le funzioni asincrone servono a gestire situazioni di emergenza e devono essere molto veloci. Per questo motivo queste funzioni non possono utilizzare qualunque istruzione GPL, ma un sottoinsieme, che garantisce tempi di esecuzione brevi.

Funzioni con parametri in ingresso (parametriche)

Una funzione può avere dichiarati dei parametri in ingresso, mentre non restituisce in alcun caso un valore. I parametri possono esser visti come delle particolari variabili locali il cui valore viene inizializzato dall'esterno nell'istante in cui la funzione viene mandata in esecuzione. I parametri vengono dichiarati con la parola chiave [PARAM](#) e seguono la stessa sintassi utilizzata per le variabili locali. I parametri devono essere elencati nelle prime righe del corpo della funzione, prima di qualsiasi altra istruzione e prima delle variabili locali.

Esistono due modi in cui vengono passati i parametri:

- **per valore:** vengono passati per valore tutti i tipi di dato semplici, cioè CHAR, INTEGER, FLOAT e DOUBLE. Il passaggio per valore comporta la creazione di una copia del valore originale. Modifiche apportate al parametro hanno effetto solo nel contesto della funzione.
- **per riferimento:** vengono passati per riferimento i tipi strutturati, cioè ARRAY, MATRICI e STRINGHE. Il passaggio per riferimento comporta l'utilizzo della variabile di origine, di conseguenza le modifiche apportate al parametro hanno effetto nel contesto della funzione chiamante. Questa caratteristica può essere sfruttata per restituire valori di ritorno alla funzione chiamante.

Tipicamente una funzione viene mandata in esecuzione con l'istruzione FCALL. Se si tratta di una funzione parametrica dopo il nome della funzione si dovrà specificare l'elenco dei valori da assegnare ai parametri. Nell'esempio che segue si ha una funzione parametrica che esegue una lavorazione di foratura. Le quote del centro del foro e la velocità di avanzamento dell'asse Z vengono passate alla funzione come parametri.

```
Function Foratura
  Param Qx as Double      ; quota X del centro foro
  Param Qy as Double
  Param vel as Float      ; velocità di avanzamento

  Movabs      X, Qx, Y, Qy
  Waitstill  X,Y
  ....
Fret
```

La chiamata di questa funzione, per effettuare ad esempio un foro alle quote (12.5 , 25.7) e con velocità di avanzamento di 3 m al minuto, potrebbe essere effettuata nel seguente modo:

```
Fcall Foratura 12.5, 25.7, 3.0
```

I parametri passati alla funzione devono corrispondere per numero e tipo a quelli dichiarati nella funzione chiamata. L'esecuzione della funzione chiamante riparte al termine dell'esecuzione della funzione chiamata. Come parametro di una funzione può anche essere [dichiarato un dispositivo](#). Ciò permette di scrivere delle funzioni di utilizzo generale, come ad esempio una funzione di azzeramento utilizzabile con tutti gli assi presenti sulla macchina:

```
Function AZZERAMENTO PUBLIC
  param asse as Axis
  movabs asse,100
Fret

Function MAIN
  ....
  Assi.Azzeramento x
Fret
```

La funzione azzeramento appartiene al gruppo Assi ed è dichiarata PUBLIC in modo che possa essere vista anche dalle funzioni dichiarate in altri gruppi. La funzione Main chiama la funzione azzeramento del gruppo assi, specificando come parametro in ingresso l'asse che si vuole muovere.

10.1.7 I parametri di tipo dispositivo

I tipi Dispositivo sono delle variabili particolari che permettono di fare riferimento ad un dispositivo della macchina. Queste tipologie di dato possono essere utilizzate **esclusivamente** nella dichiarazione dei [parametri di una funzione](#). Non si possono, quindi, dichiarare delle variabili di questo tipo. La definizione dei nomi e delle altre caratteristiche dei dispositivi rimane di pertinenza della Configurazione del sistema. Nella seguente tabella sono riportati i tipi Dispositivo e le corrispondenti parole chiave da utilizzare per la dichiarazione dei corrispondenti parametri.

Tipo	Parola chiave
Input digitale	INPUTDIG
Output digitale	OUTPUTDIG
Input analogico	INPUTANALOG
Output analogico	OUTPUTANALOG
Asse	AXIS
Timer	TIMER
Contatore	COUNTER
Flag bit	FLAGBIT
Flag switch	FLAGSWITCH
Flag port	FLAGPORT
Porto di Input	INPUTPORT
Porto di Output	OUTPUTPORT
Funzione	FUNCTION
Dispositivo generico	DEVICE
Task	TASK

Nel seguente esempio viene dichiarato ed utilizzato un parametro di tipo asse:

```
Function prova
Param asse as axis

MovAbs asse,100
waitStill asse

Fret
```

10.1.8 Il Multitasking

Il sistema Albatros è di tipo multitasking e, pertanto, si possono avere più task GPL in esecuzione, ove per task si intende un processo di gestione di una entità logica (tipicamente un gruppo).

Le tipologie di task messe a disposizione sono due: i task normali e i task real-time.

Task normali

Il multitasking è basato su un algoritmo cooperativo basato su priorità. Questo garantisce a tutti i task di essere eseguiti ciclicamente e di variare la loro priorità. L'algoritmo di schedulazione prevede l'esecuzione di una istruzione per ogni task attivo (stato running). Ad ogni task è associata una priorità impostabile con l'istruzione [SETPRIORITYLEVEL](#). La priorità è identificata da un numero intero compreso tra 0 (priorità massima) e 255 (priorità minima). Per i task con priorità 0 (zero) verrà eseguita un'istruzione ad ogni ciclo di schedulazione; per i task con priorità 1 verrà eseguita un'istruzione ogni due cicli di schedulazione e così via fino ai task con priorità 255, per i quali verrà eseguita un'istruzione ogni 256 cicli di schedulazione.

L'esecuzione dei task normali è asincrona rispetto alla frequenza di rinfresco degli assi. Ciò significa che non vi è garanzia del fatto che una funzione GPL venga completata nell'arco di tempo che intercorre tra due aggiornamenti dello stato degli assi.

Un task viene identificato dal nome della funzione GPL, da cui ha inizio la sua esecuzione.

L'esecuzione di un task può avere inizio:

- automaticamente all'inizializzazione del sistema: funzione principale dell'intergruppo e funzioni autorun.
- in seguito all'esecuzione di una istruzione [STARTTASK](#).
- in seguito al lancio in modalità manuale dall'interfaccia grafica di Albatros.

Ogni task è caratterizzato da uno stato interno:

RUNNING	il task è in esecuzione
HOLD	il task è sospeso
BREAK	il task è interrotto dal debugger

La gerarchia dei task è organizzata ad albero. Ogni task è creato da un altro di cui ne è figlio, quindi comporta che se il task madre termina, tutti i task figli saranno terminati.

Il numero massimo di task in esecuzione contemporaneamente è 500.

Si tenga presente che un numero elevato di task in esecuzione comporta una diminuzione della velocità a cui ogni singolo task viene eseguito.

Se l'applicazione che si intende realizzare richiederà l'utilizzo di un numero di task maggiore di 200, è necessario utilizzare un hardware adeguato quale i moduli Cn2128.

Task Real-Time

I task real-time si differenziano dai precedenti per il fatto che non sono soggetti ad un meccanismo di schedulazione, né hanno una priorità associata, ma vengono integralmente eseguiti ad ogni aggiornamento dello stato degli assi (real-time assi).

È assolutamente necessario che l'esecuzione di questi task termini entro un tempo determinato, perché l'esecuzione dei task GPL descritti in precedenza rimane sospesa durante l'esecuzione dei task real-time. Il sistema esegue delle verifiche sul tempo di esecuzione dei task real-time e, nel caso, in cui questi superino il tempo massimo consentito, viene generato un errore di sistema.

È, quindi, sconsigliabile creare dei cicli infiniti (ad es. con delle GOTO) all'interno di questi task, cicli che, per altro, non sono necessari, dato che l'esecuzione del codice ricomincia da capo ad ogni real-time assi.

Sempre per evitare tempi di esecuzione eccessivi, i task real-time sono soggetti a delle restrizioni nell'uso di alcune istruzioni GPL. Le istruzioni, il cui utilizzo non è consentito, sono quelle [non utilizzabili su interrupt](#).

Si consiglia l'utilizzo di task real-time solo per quelle attività, che devono essere necessariamente effettuate in modo sincrono con l'aggiornamento delle quote assi. Per la maggior parte delle attività di controllo, è più opportuno l'utilizzo dei task normali.

I task real-time vengono avviati con l'istruzione [STARTREALTIMETASK](#) e possono essere interrotti con l'istruzione [ENDREALTIMETASK](#). È possibile attivare un massimo di 256 task real-time contemporaneamente.

Non vale più il meccanismo di ereditarietà, per cui, se il task, che ha avviato un task real-time, termina, quest'ultimo rimarrà in esecuzione.

Le variabili locali dichiarate nel task real-time vengono inizializzate solo in avvio del task e, in seguito, mantengono il valore dell'ultima esecuzione.

I task real-time non sono caratterizzati dagli stati previsti per i task normali. È possibile effettuare il debug di un task real-time, ma, in questo caso, il sistema declassa automaticamente il task a "task normale" per tutta la durata dell'attività di debug.

Se in un task real-time si verifica un errore di sistema, il task viene declassato a task normale e viene posto in stato HOLD per permetterne un'analisi con il debugger.

10.1.9 Le Comunicazioni

Le comunicazioni tra il GPL ed il mondo esterno avvengono tramite tre modalità distinte:

- SEND / RECEIVE
- Comunicazioni seriali
- IPC

Send / Receive

Le istruzioni [SEND](#) e [RECEIVE](#) implementano un meccanismo di comunicazione orientato al messaggio.

La comunicazione può avvenire o all'interno dello stesso modulo (poco vantaggioso), tra più moduli di una linea, oppure tra i moduli ed il supervisore Albatros oppure con applicazioni OLE.

Il funzionamento è simile a quello della posta: ad ogni messaggio è associato un destinatario, un identificatore dell'informazione inviata (o richiesta), l'informazione stessa più delle informazioni di servizio. Albatros svolge la funzione di raccolta e smistamento delle informazioni e, in alcuni casi, fornisce direttamente le informazioni richieste.

Questa modalità di comunicazione viene normalmente utilizzata per l'invio dei programmi di lavorazione tra il supervisore ed i moduli di controllo, per la sincronizzazione dell'attività delle macchine di una linea e per l'interfacciamento con applicazioni esterne (server OLE).

Comunicazione seriali

Il linguaggio GPL fornisce alcune istruzioni, ad esempio [COMREAD](#) e [COMWRITE](#), che permettono di inviare e ricevere dati attraverso le porte seriali del controllo. È, quindi, possibile interfacciare il controllo con dispositivi esterni quali inverter, terminali o PLC. Queste istruzioni, opportunamente utilizzate, permettono di implementare protocolli di comunicazione seriale quali MODBUS-RTU etc.

IPC

L'IPC o Inter Process Communication è una modalità di comunicazione tra processi. In particolare, questa modalità permette di definire un'area di memoria, condivisa tra due o più processi, che può essere utilizzata per lo scambio di informazioni.

Tipicamente viene utilizzato, quando le prestazioni fornite dall'interfaccia OLE di Albatros non sono adeguate.

Sul lato GPL la comunicazione IPC viene implementata con le istruzioni [SENDIPC](#), [WAITIPC](#) e [TESTIPC](#). Nel caso di modulo locale i processi esterni possono fare riferimento alle API fornite da RTX, oppure al componente COM **gplipc2.dll** fornito da TPA, che ne semplifica l'uso.

Nel caso di modulo remoto i processi in esecuzione nel supervisore utilizzano il componente COM **gplipc2net.dll**.

Per maggiori informazioni contattare la TPA.

10.1.10 Variabili da usare nella programmazione

La maggior parte delle istruzioni è stata realizzata in modo da lavorare con vari tipi di variabili (CHAR, INTEGER, FLOAT, DOUBLE). Tuttavia, ogni istruzione ha un tipo di variabile preferita, per la quale è stata ottimizzata. Per ottenere migliori prestazioni durante l'esecuzione del codice GPL, si consiglia di utilizzare i tipi di variabili consigliate nella descrizione di ciascuna istruzione. In generale, si consiglia di rispettare la seguente tabella, che associa alle principali grandezze utilizzate nella programmazione i rispettivi tipi ottimali:

grandezza	tipo
quota	<i>double</i>
velocità	<i>float</i>
tempo	<i>double</i>
contatore	<i>integer</i>
valore porto / flag porto	<i>integer</i>
timeout	<i>double</i>
ingresso / uscita analogica	<i>float</i>
coseni direttori	<i>double</i>
carattere di controllo stringa	<i>char</i>
accelerazioni / decelerazioni	<i>integer</i>

10.1.11 Gli Assi

Viene normalmente indicato con il nome di "asse" un sistema elettromeccanico, il cui scopo è la movimentazione controllata di un organo di una macchina utensile.

Possiamo descrivere questo sistema in termini degli elementi che lo compongono; questi possono essere suddivisi in base alle loro caratteristiche tecnologiche.

Avremo, quindi, dei componenti meccanici tra cui:

- struttura portante
- guide
- cuscinetti
- vite + cuscinetti a ricircolo di sfere

il cui compito è quello di contrastare le forze in gioco, diminuire gli attriti, trasformare il moto da rotatorio a traslatorio, etc.

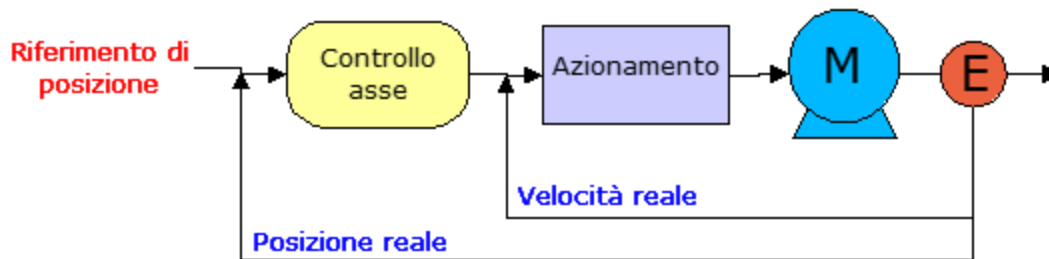
Avremo anche dei componenti elettrici ed elettronici tra cui:

- motore
- interruttori di fine corsa
- encoder
- dinamo tachimetrica

il loro compito è di fornire la potenza necessaria al movimento e rilevare lo stato del sistema.

Questi elementi sono tra loro collegati, in modo tale da permettere l'esecuzione di movimenti in modo controllato.

Schema di controllo retroazionato

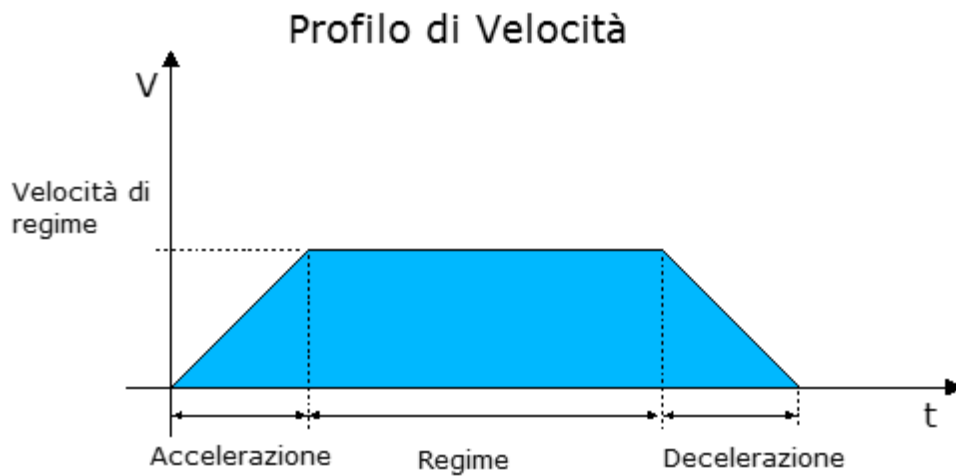


Compito del controllo numerico è quello di controllare la posizione ed il movimento di un asse.

Il movimento di un asse può essere scomposto in 5 fasi:

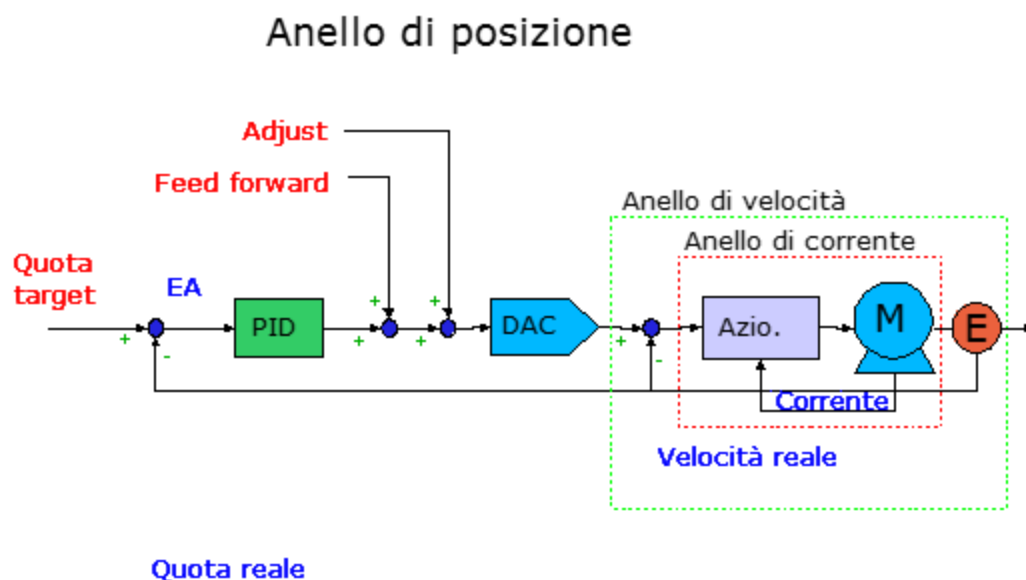
- Accelerazione** fase iniziale durante la quale l'asse aumenta progressivamente la velocità fino a raggiungere quella desiderata
- Regime** fase intermedia durante la quale l'asse si muove a velocità costante (questa fase può non esserci se lo spazio da percorrere è minore dello spazio percorso in accelerazione e decelerazione)
- Decelerazione** fase durante cui l'asse riduce progressivamente la velocità fino a 0
- Finestra** fase di attesa che l'errore d'anello si riduca al valore indicato in configurazione come "finestra di arrivo in quota"
- Quota** fine del movimento

Alla fine del movimento l'asse dovrà essere posizionato all'interno di un intervallo detto finestra di arrivo in quota (definisce la tolleranza sul posizionamento dell'asse). Se ciò non avviene entro 5 secondi dalla fine prevista del movimento, il sistema genera un errore di sistema di "movimento non finito".



Per ogni movimento il controllo numerico calcola un profilo di velocità, come quello visibile in figura, quindi, calcola le quote target suddividendo il profilo di velocità in intervalli temporali, pari al tempo di rinfresco dell'asse e calcolando l'area di ciascuna parte. L'area costituisce l'incremento di quota che l'asse deve raggiungere in quell'intervallo di tempo per rispettare il profilo di velocità visto sopra.

Il controllo dell'asse viene implementato per mezzo di un controllore PID che si occupa di "chiudere l'anello di posizione", ovvero di fornire un riferimento di velocità all'azionamento calcolato in base alla quota che si vuole raggiungere (quota target) e alla quota reale letta dall'encoder. La differenza tra quota reale e quota target è detta **Errore d'Anello**.



Schema del controllo asse di Albatros

10.1.12 Correttori di linearità

La tabella dei correttori di linearità di un asse viene vista come una matrice, il cui nome è *NomeGruppo.NomeSottogruppo.NomeAsse#correctors* o *NomeGruppo.NomeAsse#correctors* e può essere utilizzata in tutte le istruzioni, nelle quali si accede a matrici, elementi di matrice e righe di matrice.

Il numero di colonne della matrice corrisponde al numero di assi impostati nella finestra dei correttori di linearità in configurazioni assi. Gli autocorrettori sono inseriti nella prima colonna. Tutti i valori di correzione sono di tipo float. Il numero di righe totali della matrice può essere ottenuto tramite l'istruzione GPL [LASTELEM](#).

Queste matrici sono accessibili sia in lettura sia in scrittura e, se modificate, i loro valori vengono immediatamente utilizzati per la correzione della quota, solo se la correzione è stata abilitata.

Esempio:

```
Function ReadCorr
local i as integer
local j as integer
local row as integer
local column as integer
local firstvalue as float

; lettura del primo valore di autocorrezione dell'asse AX
firstvalue = X.Ax#correctors[1][1]
; numero assi nella finestra dei correttori di linearità
setval 3 column
lastelem X.Ax#correctors row
; aumenta tutte le correzioni di un valore costante
for i 1 row
  for j 1 column
    X.Ax#correctors[i][j] = X.Ax#correctors[i][j] + 0.025
  next
next
fret
```

10.1.13 Gestione dei messaggi in lingua

Come descritto nel capitolo Composizione del sistema, Albatros supporta la [visualizzazione dei messaggi di testo in più lingue](#).

Questo supporto viene fornito attraverso l'utilizzo di TpaLangs che è un programma esterno ad Albatros, che gestisce archivi di messaggi. Questo programma aiuta nell'attività di traduzione dei messaggi nelle varie lingue.

Testo associato a Errori di Ciclo e Messaggi

Un tipo particolare di testi che, normalmente, vengono visualizzati da Albatros sono i Messaggi e gli Errori di Ciclo generati dal codice GPL.

Questi vengono normalmente definiti da chi sviluppa il codice GPL durante la scrittura del codice stesso. Per facilitare il lavoro del programmatore, l'editor del GPL permette di inserire il testo di un messaggio direttamente da Albatros, ovvero senza dover utilizzare TpaLangs.

Una seconda possibilità per gestire dei messaggi in lingua è data dall'utilizzo dell'istruzione GPL [DEFMSG](#).

10.1.14 Gestione degli errori di sistema

Quando si verifica un [errore di sistema](#) (Vedi capitolo **Errori di sistema->Introduzione agli Errori di Sistema**) il comportamento normale del controllo è quello di terminare tutti i task: la gestione degli errori di sistema permette di evitare la terminazione dei task, per i quali è stata abilitata.

Errori di sistema, generati da fault, da stack underflow e da stack overflow, vengono gestiti direttamente dal controllo, senza richiamare la funzione di gestione degli errori di sistema: il task viene posto in stato HOLD.

La funzione di gestione degli errori

All'interno del codice GPL devono essere definite una o più funzioni per l'analisi dell'errore di sistema e per la conseguente definizione di opportune azioni, per mettere in sicurezza la macchina. La funzione da richiamare viene passata come parametro all'istruzione GPL [ONERRSYS](#). (Vedi capitolo **Linguaggio GPL->Istruzioni->Gestione di flusso->ONERRSYS**).

Al verificarsi di un errore di sistema, il task, che ha generato l'errore, viene posto in stato HOLD. Se i task autorun generano errori di sistema, vengono rilanciati, solo se l'errore di sistema non è un FAULT. Se l'errore di sistema viene generato senza numero di task, viene messo in HOLD il task corrente.

10.2 Funzioni speciali

10.2.1 Personalizzazione movimento assi

L'interfaccia grafica del sistema Albatros prevede la possibilità di movimentare gli assi in manuale e fornisce un supporto grafico alla taratura di questi.

Il movimento in manuale degli assi avviene per mezzo della plancia di movimentazione asse, la taratura può essere effettuata attraverso la plancia di taratura. Entrambe sono richiamabili dalla funzione di Diagnostica e dai quadri sinottici.

In entrambi i casi il movimento dell'asse viene comandato da un gruppo di funzioni GPL la cui esecuzione rimane invisibile all'utente.

Il sistema dispone di un insieme predefinito di queste funzioni che risultano adeguate nella maggior parte dei casi. Tuttavia, può essere necessario personalizzarle, ad esempio, per introdurre delle restrizioni alla movimentazione degli assi legate allo stato della macchina oppure per gestire dispositivi ausiliari come i freni asse.

La personalizzazione avviene attraverso la creazione di due funzioni GPL per ogni asse: una per la movimentazione in manuale ed una per la taratura. Queste funzioni sono opzionali, quindi se vengono rilevate dal sistema vengono utilizzate, altrimenti vengono utilizzate quelle standard. Inoltre, è prevista anche una personalizzazione parziale delle funzioni di movimento.

Movimentazione in manuale

Le funzioni di *movimento manuale* personalizzate devono rispettare le seguenti specifiche:

- La funzione dovrà appartenere allo stesso gruppo o sottogruppo cui appartiene l'asse cui fa riferimento.
- Il nome della funzione dovrà essere **MoveAx#nome_asse**, dove nome_asse va sostituito con il nome dell'asse definito in configurazione. Ad esempio, per l'asse X il nome della funzione sarà: MoveAx#X .
- La funzione dovrà prevedere i seguenti parametri:
 1. **Azione richiesta.** Può essere un movimento ad una quota assoluta, un movimento incrementale, uno stop, etc. Le azioni sono identificate da un numero intero, il compilatore GPL prevede delle costanti predefinite associate alle azioni:

_MOVAXABS	movimento a quota assoluta
_MOVAXINC	movimento a quota incrementale
_MOVAXSET	impostazione della quota
_MOVAXFREE	impostazione stato free
_MOVAXNORMAL	impostazione stato normal
_MOVAXEND	ripristino stato asse a fine movimento (non serve per fermare l'asse)
 2. **Risultato.** Serve al sistema per sapere se l'azione richiesta è gestita dalla funzione personalizzata. Se l'azione non è gestita viene utilizzata la funzione standard corrispondente. Si tratta quindi di un valore di ritorno che la funzione personalizzata deve impostare e a questo scopo è definito come un parametro passato per riferimento (array di un solo elemento).
 3. **Velocità.** È significativa solo quando l'azione richiesta è un movimento e ne imposta la velocità.
 4. **Quota.** Significativa solo per le azioni di movimento e di impostazione quota.

Esempio di funzione di movimentazione personalizzata:

```
Function MoveAx#X
  param action as integer
  param result[1] as integer
  param speed as float
  param position as double

  setval      1,result[1]
```

```

select action
case _MOVAXEND
    fcall EndMovement
case _MOVAXABS
    fcall AbsMovement X, speed, position
case _MOVAXINC
    fcall IncMovement X, speed, position
case _MOVAXSET
    fcall PositionSet X, position
case _MOVAXFREE
    fcall FreeAxis
case _MOVAXNORMAL
    fcall NormalAxis
case else
    call Unknown
endselect

fret

```

```

Unknown:
setval    0, result[1]
ret

```

Le funzioni EndMovement, AbsMovement, etc. (i nomi non sono vincolanti) dovranno implementare la gestione personalizzata delle azioni richieste. Per facilitare il lavoro del programmatore riportiamo le funzioni standard di movimentazione che possono servire come traccia per lo sviluppo di quelle personalizzate.

Taratura

Le funzioni di taratura personalizzate devono rispettare le seguenti specifiche:

- La funzione dovrà appartenere allo stesso gruppo o sottogruppo cui appartiene l'asse cui fa riferimento.
- Il nome della funzione dovrà essere **CalibAx#nome_asse**, dove nome_asse va sostituito con il nome dell'asse definito in configurazione. Ad esempio per l'asse X il nome della funzione sarà: CalibAx#X .
- La funzione dovrà prevedere i seguenti parametri:
 1. **Azione richiesta.** Può essere un movimento punto-punto o un movimento interpolato.
 2. **Risultato.** Serve al sistema per sapere se l'azione richiesta è gestita dalla funzione personalizzata. Se l'azione non è gestita viene utilizzata la funzione standard corrispondente.
 3. **Velocità.** Velocità di movimento durante la taratura
 4. **Quota Positiva.** Quota positiva del movimento alternato di taratura.
 5. **Quota Negativa.** Quota negativa del movimento alternato di taratura.
 6. **Attesa.** Attesa tra un movimento e quello successivo

NOTA: si deve tener presente che in alcuni casi azioni effettuate nella plancia di taratura comportano l'esecuzione della funzione di movimentazione asse. Ad esempio, a fine movimento di taratura (quando viene premuto il pulsante di stop) viene effettuata un'operazione di ripristino dell'asse per la quale viene chiamata la funzione di movimentazione asse personalizzata con il parametro azione impostato a _MOVAXEND. Allo stesso modo, quando si modifica la quota dell'asse dalla plancia di taratura, viene chiamata la funzione di movimentazione asse con il parametro azione impostato a _MOVAXSET.

Esempio di funzione di taratura personalizzata:

```

Function CalibAx#X
param action as integer
param result[1] as integer
param speed as float
param PosPosition as double
param NegPosition as double
param waitTime as float

setval    1,result[1]

```

```

select action
case _CALAXPP
    fcall PPCalibration X, speed, PosPosition, NegPosition, _
        waitTime
case _CALAXINT
    fcall IntCalibration X, speed, PosPosition, NegPosition, _
        waitTime
case else
    call Unknown
endselect

fret

Unknown:
setval      0, result[1]
ret

```

Le funzioni PPCalibration e IntCalibration (i nomi non sono vincolanti) dovranno implementare la gestione personalizzata delle azioni richieste. Per facilitare il lavoro del programmatore riportiamo le funzioni standard di taratura che possono servire come traccia per lo sviluppo di quelle personalizzate.

Interazione con la finestra di Movimentazione asse in manuale

Le funzioni per l'interazione con la finestra di movimento manuale devono rispettare le seguenti specifiche:

- La funzione dovrà appartenere allo stesso gruppo o sottogruppo cui appartiene l'asse cui fa riferimento.
- Il nome della funzione dovrà essere **MoveAx#nome_asse#Azione**, dove nome_asse va sostituito con il nome dell'asse definito in configurazione e Azione può assumere uno tra le seguenti definizioni:

OPEN	indica che l'utente ha appena aperto la finestra di movimentazione asse
CLOSE	indica che l'utente sta per chiudere la finestra di movimentazione asse
ACTIVE	segnala che la finestra di movimentazione asse è attiva
INACTIVE	segnala che la finestra di movimentazione asse non è attiva
JOG	indica che è impostato un movimento per spostamento gestito runtime dall'operatore
STEP	indica che è impostato un movimento con spostamento di passo predefinito
ABSOLUTE	indica che è impostato un movimento con spostamento a quota stabilita

Ad esempio se la finestra di movimentazione asse per l'asse X è stata aperta, verrà chiamata la funzione con nome MoveAx#X#Open.

Modifiche alla finestra di Movimentazione asse in manuale

È possibile aggiungere fino a 4 pulsanti alla finestra di movimentazione asse. Devono essere definite, nello stesso gruppo o sottogruppo in cui è definito l'asse interessato, delle funzioni GPL a nome fisso MoveAx#NomeAsse#BUTTONtesto. NomeAsse rappresenta il nome dell'asse interessato e testo rappresenta il testo che apparirà sul pulsante. Il testo può contenere il carattere '&' per introdurre un acceleratore di tastiera. Se testo inizia con un numero compreso tra 1 e 4, questo numero viene considerato come la posizione in cui verrà inserito il pulsante nella finestra di movimento asse. Il testo del pulsante può essere tradotto introducendo una DEFMSG, nel gruppo in cui si trova l'asse, che abbia come identificatore MOVEAX#BUTTONtesto. La pressione del pulsante personalizzato comporta l'esecuzione della funzione GPL associata. Non viene eseguita nessuna attesa di fine esecuzione, né alcuna verifica di inizio esecuzione della funzione.

10.2.2 Funzioni standard di movimentazione e taratura

Quelle che seguono sono le funzioni standard utilizzate dalla plancia di movimentazione asse e dalla plancia di taratura.

Le funzioni variano a seconda del tipo di asse interessato: conteggio, passo-passo, etc.

Queste funzioni possono essere [personalizzate](#).

Funzioni standard di movimentazione manuale

Movimento a quota assoluta

```

; per assi passo-passo
Function AbsMovement
    param axisname as axis
    param speed as float

```

```

    param position as double

    ifstill    axisname goto move
    fret

move:
    setvel    axisname, speed
    movabs    axisname, position
    waitstill axisname
    fret

; per tutti gli altri tipi di asse
Function AbsMovement
    param axisname as axis
    param speed as float
    param position as double

    iftarget  axisname goto move
    ifstill   axisname goto move
    fret

move:

    setvel    axisname, speed
    movabs    axisname, position
    waitstill axisname
    fret

```

Movimento incrementale

```

; per assi passo-passo
Function IncMovement
    param axisname as axis
    param speed as float
    param position as double

    ifstill    axisname goto move
    fret

move:
    setvel    axisname, speed
    movinc    axisname, position
    waitstill axisname
    fret

; per tutti gli altri tipi di asse
Function IncMovement
    param axisname as axis
    param speed as float
    param position as double

    iftarget  axisname goto move
    ifstill   axisname goto move
    fret

move:
    setvel    axisname, speed
    movinc    axisname, position
    waitstill axisname
    fret

```

Impostazione quota

```

; per assi di conteggio
Function PositionSet
  param axisname as axis
  param position as double

  setquote   axisname, position
  fret

```

```

; per assi passo-passo
Function PositionSet
  param axisname as axis
  param position as double

  ifstill    axisname goto set
  fret
set:
  setquote   axisname, position
  fret

```

```

; per tutti gli altri tipi di asse
Function PositionSet
  param axisname as axis
  param position as double

  iftarget   axisname goto set
  ifstill    axisname goto set
  fret
set:
  setquote   axisname, position
  fret

```

Impostazione stato free

```

Function FreeAxis
  param axisname as axis

  free      axisname
  fret

```

Impostazione stato normal

```

Function NormalAxis
  param axisname as axis

  normal    axisname
  fret

```

Funzioni standard di taratura**Taratura in punto-punto**

```

; per assi passo-passo
Function PPCalibration
  param axisname as axis
  param speed as float
  param PosPosition as double
  param NegPosition as double
  param waitTime as float

```

```

    setvel      axisname, speed
loop:
    movabs     axisname, PosPosition
    waitstill  axisname
    delay      WaitTime
    movabs     axisname, NegPosition
    waitstill  axisname
    delay      WaitTime
    goto       loop
    fret

```

; per tutti gli altri tipi di asse

```

Function PPCalibration
    param axisname as axis
    param speed as float
    param PosPosition as double
    param NegPosition as double
    param WaitTime as float

    setvel      axisname, speed
loop:
    movabs     axisname, PosPosition
    waitstill  axisname
    ifquotet   axisname,<>,PosPosition goto exit
    delay      WaitTime
    movabs     axisname, NegPosition
    waitstill  axisname
    ifquotet   axisname,<>,NegPosition goto exit
    delay      WaitTime
    goto       loop
exit:
    fret

```

Taratura in interpolazione

```

Function IntCalibration
    param axisname as axis
    param speed as float
    param PosPosition as double
    param NegPosition as double
    param WaitTime as float

    setveli    axisname, speed
loop:
    linearabs   axisname, PosPosition
    waitstill  axisname
    ifquotet   axisname,<>,PosPosition goto exit
    delay      WaitTime
    linearabs   axisname, NegPosition
    waitstill  axisname
    ifquotet   axisname,<>,NegPosition goto exit
    delay      WaitTime
    goto       loop
exit:
    fret

```


10.2.3 Funzione OnUIEnd#

La funzione "OnUIEnd#" viene eseguita, se presente, da Albatros prima di terminare tutti i task di un modulo. Deve essere definita nel file di funzioni dell'intergruppo. La funzione "OnUIEnd#" ha un tempo massimo di esecuzione di 2 secondi, dopodiché Albatros terminerà tutti i task.

Il tempo massimo di attesa che questa funzione termini l'esecuzione è configurabile in Tpa.ini, nella sezione [Albatros], alla voce Timeout.OnUIEnd=*valore*, dove *valore* è in millisecondi e non può essere maggiore di 60000.

10.2.4 Funzione OnUIPlugged#

La funzione "OnUIPlugged#" viene eseguita, quando serve sapere, ad esempio all'accensione di un impianto, se Albatros ha comunicato con il modulo remoto. Tale funzione deve essere definita nell'intergruppo.

10.2.5 Funzione OnUIUnPlugged#

La funzione "OnUIUnplugged#" viene eseguita prima di terminare l'esecuzione di Albatros (e quindi prima che Albatros si scolleghi da un modulo). Tale funzione deve essere definita nell'intergruppo. Albatros esegue questa funzione entro massimo 2 secondi, durante i quali vengono letti:

- errori di ciclo
- errori sistema
- messaggi

Alla fine dell'esecuzione Albatros si chiude.

Il tempo massimo di attesa, che questa funzione termini l'esecuzione, è configurabile in Tpa.ini, nella sezione [Albatros], alla voce Timeout.OnUIUnplugged=*valore*, dove *valore* è in millisecondi e non può essere maggiore di 60000.

10.3 Istruzioni

10.3.1 Convenzioni

Le pagine seguenti sono organizzate in forma di schede e descrivono, relativamente ad ogni istruzione:

- la sintassi
- una descrizione degli argomenti: tipo di dato e valori ammessi
- una descrizione di funzionamento
- eventuali note
- eventuali esempi

Le istruzioni dello stesso tipo sono state raggruppate per facilitare l'apprendimento e la consultazione.

10.3.2 Tipologia delle istruzioni del linguaggio GPL

Il linguaggio si compone di istruzioni che possono essere così raggruppate:

Istruzioni per la gestione degli Input/Output

GETFEED	legge il feed rate override
INPANALOG	legge un input analogico
INPFLAGPORT	legge un flag port
INPPORT	legge un porto digitale
MULTIINPPORT	legge fino a 4 porti di uscita
MULTIOUTPORT	imposta fino a 4 porti di uscita
MULTIRESETFLAG	imposta più flag a 0
MULTIRESETOUT	imposta più uscite a 0
MULTISETFLAG	imposta più flag a 1

MULTISETOUT	imposta più uscite a 1
MULTIWAITFLAG	attende lo stato di un flag bit o flag switch
MULTIWAITINPUT	attende lo stato di più ingressi
OUTANALOG	modifica un output analogico
OUTFLAGPORT	modifica un flag port
OUTPORT	modifica un porto digitale
RESETFLAG	imposta un flag a 0
RESETOUT	imposta un'uscita a 0
SETFLAG	imposta un flag a 1
SETOUT	imposta un'uscita a 1
WAITFLAG	attende lo stato di un flag bit o flag switch
WAITINPUT	attende lo stato di un ingresso
WAITPERSISTINPUT	attende lo stato persistente di un ingresso

Istruzioni per la gestione degli Assi

CHAIN	concatena un asse a un altro
CIRCABS	interpolazione circolare assoluta
CIRCINC	interpolazione circolare incrementale
CIRCLE	esegue un cerchio
COORDIN	movimento assi coordinati
DISABLECORRECTION	disabilita la correzione lineare per l'asse specificato
EMERGENCYSTOP	forza un arresto di emergenza degli assi
ENABLECORRECTION	abilita la correzione lineare per l'asse specificato
ENDMOV	termina il movimento di un asse
FASTREAD	lettura veloce quote assi
FREE	pone l'asse in free
HELICABS	interpolazione elicoidale assoluta
HELICINC	interpolazione elicoidale incrementale
JERKCONTROL	abilita o disabilita il controllo sui movimenti in interpolazione
JERKSMOOTH	raccorda con continuità di accelerazione e di velocità, i profili di velocità tenuti dagli assi durante i movimenti in contornatura
LINEARABS	interpolazione lineare assoluta
LINEARINC	interpolazione lineare incrementale
MOVABS	movimento assi in modo assoluto
MOVINC	movimento assi in modo incrementale
MULTIABS	interpolazione lineare multiasse in modo assoluto
MULTIINC	interpolazione lineare multiasse in modo incrementale
NORMAL	toglie il free da un asse
RESRIFLOC	ripristina riferimenti iniziali
SETINDEXINTERP	associa ad un asse una variabile per il conteggio dei blocchi di interpolazione eseguiti
SETLABELINTERP	associa ad un asse una variabile per l'identificazione di un blocco di spostamento
SETPFLY	azzeramento al volo
SETPFLYCHAINSTRAT	controlla il comportamento dell'asse slave a fronte di una istruzione setpfly sul master
SETPZERO	azzeramento su tacca di zero
SETPZEROCHAINSTRAT	controlla il comportamento dell'asse slave a fronte di una istruzione setpzero sul master
SETQUOTE	imposta la quota
SETQUOTECHAINSTRAT	controlla il comportamento dell'asse slave a fronte di una istruzione setquote sul master
SETRIFLOC	imposta riferimenti spaziali
SETTOLERANCE	imposta i valori di tolleranza per l'interpolazione lineare
START	riprende il movimento di un asse
STARTINTERP	forza l'inizio di un'interpolazione
STOP	sospende il movimento di un asse
SWITCHENC	permette di sostituire l'encoder di un asse con quello di un altro asse
WAITACC	attesa che l'asse sia in accelerazione
WAITCOLL	attesa che l'asse superi una quota dalla quale iniziare a verificare la presenza di una collisione
WAITDEC	attesa che l'asse sia in decelerazione
WAITREG	attesa che l'asse sia in regime
WAITSTILL	attesa che la quota finale sia uguale alla quota target
WAITTARGET	attesa che l'asse sia in target
WAITWIN	attesa che l'asse sia in finestra

Istruzioni per la gestione dei Parametri asse

Letture/Scrittura

[DEVICEID](#) scrive l'indirizzo logico associato ad un dispositivo
[GETAXIS](#) legge uno o più dati di un asse

Movimento Punto-Punto

[SETACC](#) imposta l'accelerazione
[SETDEC](#) imposta la decelerazione
[SETDERIV](#) imposta il coefficiente di azione derivata
[SETFEED](#) imposta il feed rate punto-punto
[SETFEEDF](#) imposta il feed forward
[SETFEEDFA](#) imposta il feed forward di accelerazione
[SETINTEG](#) imposta il coefficiente di azione integrale
[SETMULTIFEED](#) imposta il valore percentuale di feed rate override degli assi indicati
[SETPROP](#) imposta il coefficiente di azione proporzionale
[SETSLOPE](#) imposta il tipo di rampa nei movimenti in rapido
[SETVEL](#) imposta la velocità

Movimento interpolato

[LOOKAHEAD](#) imposta il lookahead di interpolazione
[SETACCI](#) imposta l'accelerazione per interpolazione
[SETACCLIMIT](#) abilita e disabilita il calcolo automatico della velocità di regime di interpolazione
[SETACCSTRATEGY](#) seleziona il tipo di accelerazione
[SETAXPARTYPE](#) cambia il set di parametri asse in uso
[SETCONTORNATURE](#) imposta l'angolo di contornatura
[SETDECI](#) imposta la decelerazione per interpolazione
[SETDERIVI](#) imposta il coefficiente di azione derivata d'interpolazione
[SETFEEDFAI](#) imposta il feed forward di accelerazione in interpolazione
[SETFEEDI](#) imposta il feed rate in interpolazione
[SETFEEDFI](#) imposta il feed forward in interpolazione
[SETINTEGI](#) imposta il coefficiente di azione integrale d'interpolazione
[SETPROPI](#) imposta il coefficiente di azione proporzionale d'interpolazione
[SETSLOPEI](#) imposta il tipo di rampa nei movimenti in interpolazione
[SETSLOWPARAM](#) modifica i parametri necessari al calcolo della velocità di rallentamento nel caso in cui sia attiva la funzionalità di rallentamento in contornatura
[SETVELI](#) imposta la velocità d'interpolazione
[SETVELLIMIT](#) imposta le singole componenti di velocità dell'asse specificato

Movimento coordinato

[SETFEEDCOORD](#) imposta il valore percentuale di variazione massima istantanea del feed rate dell'asse.
[SETOFFSET](#) imposta un offset di quota

Movimento concatenato

[RATIO](#) imposta il rapporto di concatenamento di un asse slave rispetto al proprio master
[SETDYNRATIO](#) varia in modo dinamico, durante il movimento dell'asse master, il rapporto di concatenamento

Parametri generici

[DYNLIMIT](#) abilita o disabilita il test sul superamento dei limiti asse in modo dinamico.
[ENABLESTARTCONTROL](#) abilita e imposta il timeout per il controllo della mancata partenza o improvviso stop dell'asse
[NOTCHFILTER](#) imposta la frequenza di taglio del filtro notch per l'asse specificato
[RESLIMNEG](#) disabilita il limite negativo dell'asse
[RESLIMPOS](#) disabilita il limite positivo dell'asse
[SETADJUST](#) imposta l'adjust di un asse
[SETBACKLASH](#) diminuisce o elimina gli effetti dei giochi meccanici sulla traiettoria dell'asse

SETBIGWINFACTOR	modifica il fattore di moltiplicazione per il calcolo della finestra grande sull'asse scelto
SETDEADBAND	imposta i parametri di tensione minima per l'asse indicato
SETENCLIMIT	imposta i limiti di collegamento encoder errato
SETINDEXEN	attiva o disattiva sull'asse l'azzeramento della quota in corrispondenza della tacca di zero
SETINTEGTIME	imposta il numero di campioni dell'errore d'anello utilizzati per il calcolo della componente integrale
SETIRMP	imposta la velocità di inizio rampa
SETLIMNEG	imposta il limite negativo dell'asse
SETLIMPOS	imposta il limite positivo dell'asse
SETMAXER	imposta il valore massimo di inseguimento tollerato
SETMAXERNEG	imposta il valore massimo di inseguimento tollerato (direzione negativa)
SETMAXERPOS	imposta il valore massimo di inseguimento tollerato (direzione positiva)
SETMAXERTYPE	imposta il tipo di test sul servo error
SETPHASESINV	attiva o disattiva sull'asse indicato l'inversione fasi
SETREFINV	abilita o disabilita sull'asse indicato l'inversione di riferimento di velocità
SETRESOLUTION	cambia la risoluzione di un asse

Istruzioni per la gestione dei Contatori

DECOUNTER	decrementa un contatore
INCOUNTER	incrementa un contatore
SETCOUNTER	imposta un contatore

Istruzioni per la gestione dei Timer

HOLDTIMER	blocca un timer
SETTIMER	imposta un timer
STARTTIMER	fa partire il timer

Istruzioni per la gestione delle Matrici

CLEAR	azzeramento di variabile, vettore, matrice
FIND	ricerca di un elemento
FINDB	ricerca di un elemento in vettore o matrice ordinati in modo crescente
LASTELEM	ultimo elemento di un vettore o matrice
LOCAL	dichiarazione di variabile, vettore, matrice locale
MOVEMAT	copia una riga di matrice in un'altra
PARAM	dichiarazione di un parametro di funzione
SETVAL	modifica una variabile
SORT	ordinamento di vettore o matrice

Istruzioni per la gestione delle Stringhe

ADDSTRING	concatena due stringhe
CONTROLCHAR	imposta un carattere di controllo in una variabile stringa
LEFT	estrae i primi caratteri
LEN	legge la lunghezza di una stringa
MID	estrae alcuni caratteri
RIGHT	estrae gli ultimi caratteri
SEARCH	ricerca in una stringa
SETSTRING	modifica una variabile stringa
STR	converte da numero a stringa
VAL	converte da stringa a numero

Istruzioni per la gestione delle Comunicazioni

CLEARRECEIVE	svuota la lista di RECEIVE da soddisfare
COMCLEARRXBUFFER	svuota il buffer di ricezione di una porta seriale

COMCLOSE	chiude una porta seriale
COMGETERROR	legge il codice di errore
COMGETRXCOUNT	legge il numero di byte presenti nel buffer di ricezione
COMOPEN	apre una porta seriale
COMREAD	legge dalla porta seriale
COMREADSTRING	legge una stringa dalla porta seriale
COMWRITE	scrive sulla porta seriale
COMWRITESTRING	scrive una stringa sulla porta seriale
RECEIVE	ricezione dati dall'esterno
SEND	invio dati all'esterno
SENDIPC	invia un'informazione IPC
WAITIPC	attende l'arrivo di un'informazione IPC
WAITRECEIVE	ricezione dati dall'esterno con attesa

Istruzioni matematiche

ABS	valore assoluto
ADD	somma
AND	AND binario
ARCCOS	arcocoseno
ARCSIN	arcoseno
ARCTAN	arcotangente
COS	coseno
DIV	divisione
EXP	esponenziale
EXPR	risolve espressioni matematiche
LOG	logaritmo naturale
LOGDEC	logaritmo in base 10
MOD	modulo
MUL	moltiplicazione
NOT	NOT binario
OR	OR binario
RANDOM	genera un numero casuale
RESETBIT	imposta a 0 un bit
ROUND	arrotondamento
SETBIT	imposta a 1 un bit
SHIFTL	ruota i bit a sinistra
SHIFTR	ruota i bit a destra
SIN	seno
SQR	radice quadrata
SUB	sottrazione
TAN	tangente
TRUNC	troncamento
XOR	XOR binario

Istruzioni per la gestione Multitask

ENDMAIL	segnala la fine dell'esecuzione di un comando
ENDREALTIMETASK	termina un task real-time
ENDTASK	termina un task
GETPRIORITYLEVEL	legge il livello di priorità del task corrente
GETREALTIME	ritorna il tempo passato dall'inizio del real-time assi
GETREALTIMECOUNT	ritorna il numero di real-time trascorsi
HOLDTASK	sospende l'esecuzione di un task
RESUMETASK	riprende l'esecuzione di un task
SENDMAIL	invia un comando al mailbox 'mail'
SETPRIORITYLEVEL	imposta il livello di priorità per il task corrente
STARTREALTIMETASK	avvia un task real-time
STARTTASK	avvia l'esecuzione di un task
STOPTASK	sospende l'esecuzione di un task e ferma il movimento degli assi associati
WAITMAIL	riceve un comando dal mailbox 'mail'
WAITTASK	attende che un task termini l'esecuzione

Istruzioni per la gestione di Flusso

CALL	chiamata a sottoprogramma
DELONFLAG	disabilita la gestione dell'emergenza su flag bit o flag switch
DELONINPUT	disabilita la gestione dell'emergenza su input digitale
ENDREP	fine ripetizione blocco con REPEAT
FCALL	chiamata a funzione
FOR	estensione di REPEAT
FRET	ritorno da chiamata a funzione
GOTO	salto a etichetta
IF	test su una variabile
IFACC	test se l'asse è in accelerazione
IFAND	test su operazione di AND
IFBIT	test su bit
IFBLACKBOX	test se la registrazione dell'attività dei dispositivi logici è attiva.
IFCHANGEVEL	test se l'asse è in cambio velocità
IFCOUNTER	test su un contatore
IFDEC	test se l'asse è in decelerazione
IFDIR	test su direzione asse
IFERRAN	test su errore d'anello
IFERROR	test sull'errore di ciclo attivo
IFFLAG	test su un flag
IFINPUT	test su un input
IFMESSAGE	test sul messaggio attivo
IFOR	test su operazione di OR
IFOUTPUT	test su un output
IFQUOTER	test su quota reale
IFQUOTET	test su quota target
IFRECEIVED	test su ricezione dati
IFREG	test se l'asse è in regime
IFSAME	verifica che i due argomenti si riferiscano allo stesso dato
IFSTILL	test se l'asse è fermo
IFSTR	test su stringa
IFTARGET	test se l'asse è in target
IFTASKHOLD	test se funzione parallela è sospesa
IFTASKRUN	test se funzione parallela è in esecuzione
IFTIMER	test su un timer
IFVALUE	test su una variabile
IFVEL	test su velocità asse
IFWIN	test se l'asse è in finestra
IFXOR	test su operazione di XOR
NEXT	fine ripetizione blocco con FOR
ONERRSYS	imposta la chiamata a una funzione su errore di sistema
ONFLAG	emergenza su flag bit o flag switch
ONINPUT	emergenza su input digitale
REPEAT	ripetizione di un blocco di istruzioni
RET	ritorno da sottoprogramma
SELECT	selezione multipla con salto
TESTIPC	verifica la presenza di un'informazione IPC
TESTMAIL	test e ricezione di un comando

Istruzioni Varie

CLEARERRORS	cancella tutti gli errori di ciclo del modulo
CLEARMESSAGES	cancella tutti i messaggi del modulo
DEFMSG	definisce un messaggio di gruppo
DELAY	blocca la funzione corrente per un periodo di tempo
DELERROR	cancella un precedente errore di ciclo
DELMESSAGE	cancella un precedente messaggio
ERROR	invia al PC un errore di ciclo
IFDEF/ELSEDEF/ENDDF	test per la compilazione condizionale
MESSAGE	invia al PC un messaggio
SYSFAULT	disattiva il segnale di SYSOK
SYSOK	attiva il segnale di SYSOK
TYPEOF	tipo dell'argomento
WATCHDOG	abilita, aggiorna, disabilita il watchdog da GPL sul modulo hardware TMSWD

Istruzioni per la gestione MECHATROLINK-II

MECCOMMAND	invia un comando all'azionamento dell'asse
MECGETPARAM	legge un parametro dell'asse indicato
MECGETSTATUS	legge i valori di STATUS, ALARM e IO_MON
MECSETPARAM	scrive un parametro nell'asse indicato

Istruzioni per la gestione dei bus di campo standard

AXCONTROL	imposta un valore per la ControlWord
AXSTATUS	restituisce il valore contenuto nella StatusWord
CNBYDEVICE	restituisce il numero di scheda e nodo di un dispositivo
READDICTIONARY	legge il contenuto di un oggetto del dizionario
WRITEDICTIONARY	scrive il contenuto di un oggetto del dizionario

Istruzioni per la gestione EtherCAT

ACTIVATEMODE	imposta una modalità operativa
ECATGETREGISTER	restituisce il contenuto di un registro dell'ESC (EtherCAT Slave Controller)
ECATSETREGISTER	scrive il contenuto di un registro dell'ESC (EtherCAT Slave Controller)
GETPDO	restituisce un oggetto all'interno di un PDO EtherCAT
SETEOF	attiva o disattiva lo Sniffer
SETPDO	imposta un oggetto all'interno di un PDO EtherCAT

Istruzioni per la gestione del bus CAN

GETCNSTATE	restituisce lo stato del protocollo NMT per un nodo di una scheda.
GETSDOERROR	restituisce l'ultimo errore che si è verificato
GETMNSTATE	restituisce lo stato del protocollo NMT per il nodo master della scheda.
RECEIVEPDO	legge il contenuto di un PDO asincrono
SENDPDO	scrive il contenuto di un PDO asincrono
SETNMTSTATE	imposta lo stato del protocollo NMT per il nodo della scheda

Istruzioni per la Simulazione

DISABLE	disabilita uno o più assi
DISABLEFORCEDINPUT	disabilita la forzatura degli ingressi
ENABLE	abilita uno o più assi
ENABLEFORCEDINPUT	abilita la forzatura degli ingressi
RESETFORCEDINPUT	forza un ingresso ad OFF
SETFORCEDANALOG	forza un ingresso analogico
SETFORCEDINPUT	forza un ingresso ad ON
SETFORCEDPORT	forza un porto di ingressi

Istruzioni per le funzionalità di "Blackbox"

ENDBLACKBOX	termina la funzionalità di registrazione
PAUSEBLACKBOX	sospende la funzionalità di registrazione
STARTBLACKBOX	avvia la funzionalità di registrazione

Istruzioni per la gestione ISO

ISOG0	imposta il movimento in rapido
ISOG1	imposta il movimento interpolato
ISOG9	imposta l'arresto forzato del movimento
ISOG90	imposta l'interpretazione delle quote come quote assolute
ISOG91	imposta l'interpretazione delle quote come quote relative
ISOG93	imposta l'interpretazione della velocità come inverso del tempo
ISOG94	imposta l'interpretazione della velocità come unità di misura al minuto
ISOG216	definisce le matrici per la parametrizzazione della macchina
ISOG217	descrive gli assi fisici e gli assi virtuali che costituiscono la macchina
ISOM2	libera gli assi dal movimento ISO
ISOM6	seleziona gli indici delle matrici di parametrizzazione

[ISOSETPARAM](#)

imposta alcuni parametri caratterizzanti la fluidità del movimento interpolato ISO

[KINEMATICEXPR](#)

imposta le singole espressioni di cinematica inversa e diretta

10.3.3 Input/Output

GETFEED

Sintassi

GETFEED **variabile**

Argomenti

variabile feed rate

Descrizione

Copia il valore del feed rate, letto dalla scheda di I/O remoto, nella **variabile** specificata. Il valore del feedrate è compreso tra 0 e 100 ed è un valore percentuale.

Opera su un ingresso analogico non visibile in configurazione. Tutte le schede TPA che gestiscono il feed rate hanno un connettore dedicato.

INPANALOG

Sintassi

INPANALOG **nomeinpanalogico, variabile**

Argomenti

nomeinpanalogico nome dispositivo di tipo input analogico
variabile variabile

Descrizione

Copia il valore dell'input analogico specificato da **nomeinpanalogico** nella **variabile** specificata.

INPFLAGPORT

Sintassi

INPFLAGPORT **nomeflagport, variabile**

Argomenti

nomeflagport nome di dispositivo tipo porto di flag
variabile variabile

Descrizione

Copia lo stato del porto di flag specificato da **nomeflagport** nella **variabile** specificata.

Il porto di flag viene visto come una maschera di bit. Ad ogni flag del porto viene associato un bit. Se un flag si trova nello stato "ON", il corrispondente bit viene impostato a 1.

INPPORT

Sintassi

INPPORT **nomeporto, variabile**

Argomenti

nomeporto nome di dispositivo tipo porto di input
variabile variabile integer o char

Descrizione

Copia lo stato del porto di input **nomeporto** nella **variabile** specificata.

Il porto di input viene visto come una maschera di bit. Se un input del porto si trova in stato "ON" il corrispondente bit viene impostato ad 1.

MULTIINPPORT

Sintassi

MULTIINPPORT **porto1[,...],porto4],variabile**

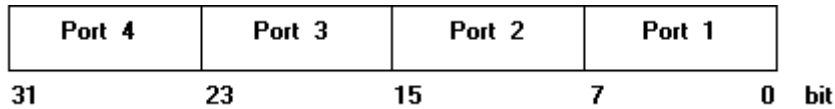
Argomenti

porto1 fornisce i bit da 0 a 7
porto2 fornisce i bit da 8 a 15

porto3 fornisce i bit da 16 a 23
porto4 fornisce i bit da 24 a 31
variabile variabile integer che riceve i porti di ingresso

Descrizione

Legge fino a quattro porti di ingresso contemporaneamente e li scrive in una **variabile**. La lettura dei porti avviene in maniera atomica, questo garantisce che la lettura sia eseguita all'interno dello stesso real-time. Il **porto1** corrisponde al byte più basso, mentre il **porto4** a quello più alto.

**MULTIOUTPUT****Sintassi**

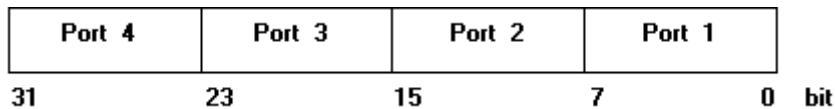
MULTIOUTPUT **valore, nomeporto1[,...,nomeporto4]**

Argomenti

valore numero o variabile integer da scrivere nei porti di uscita
nomeporto1 riceve i bit da 0 a 7
nomeporto2 riceve i bit da 8 a 15
nomeporto3 riceve i bit da 16 a 23
nomeporto4 riceve i bit da 24 a 31

Descrizione

Scrive **valore** in quattro porti di uscita contemporaneamente. La scrittura dei porti indicati avviene in maniera atomica, questo garantisce che la scrittura sia eseguita all'interno dello stesso real-time. Se **nomeporto2**, **nomeporto3**, **nomeporto4** non sono specificati, il relativo byte ha valore 0.

**MULTIRESETFLAG****Sintassi**

MULTIRESETFLAG **maschera, nomeflag1[, ..., nomeflag32]**

Argomenti

maschera maschera dei flag interessati - costante o variabile
nomeflag1 nome di dispositivo tipo flag

Descrizione

Di tutti i flag **nomeflag** (1÷32), disattiva, ossia pone ad "OFF", quelli il cui bit nell'argomento **maschera** è posto a 1.
 Il bit 0 di **maschera** (quello di peso più basso) corrisponde a **nomeflag1**.

MULTIRESETOUT**Sintassi**

MULTIRESETOUT **maschera, nomeoutput1[, ..., nomeoutput32]**

Argomenti

maschera maschera delle uscite interessate - costante o variabile
nomeoutput1 nome di dispositivo tipo output

Descrizione

Di tutte le uscite **nomeoutput** (1÷32), disattiva quelle il cui bit nell'argomento **maschera** è posto a 1. Il bit 0 di **maschera** (quello di peso più basso) corrisponde a **nomeoutput1**.

MULTISETFLAG

Sintassi

MULTISETFLAG **maschera, nomeflag1[, ..., nomeflag32]**

Argomenti

maschera maschera dei flag interessati - costante o variabile
nomeflag1 nome di dispositivo tipo flag

Descrizione

Di tutti i flag **nomeflag** (1÷32), attiva, ossia pone ad "ON", quelli il cui bit nell'argomento **maschera** è posto a 1. Il bit 0 di **maschera** (quello di peso più basso) corrisponde a **nomeflag1**.

MULTISETOUT

Sintassi

MULTISETOUT **maschera, nomeoutput1[, ..., nomeoutput32]**

Argomenti

maschera maschera delle uscite interessate - costante o variabile
nomeoutput1 nome di dispositivo tipo output

Descrizione

Di tutte le uscite **nomeoutput** (1÷32), attiva quelle il cui bit nell'argomento **maschera** è posto a 1. Il bit 0 di **maschera** (quello di peso più basso) corrisponde a **nomeoutput1**. Se l'uscita è di tipo monostabile viene disattivata automaticamente allo scadere di un timeout fisso di 200 millisecondi.

MULTIWAITFLAG

Sintassi

MULTIWAITFLAG **maschera, flag1[, ..., flag32], stato [, timeout [, GOTO etichetta]]**
MULTIWAITFLAG **maschera, flag1[, ..., flag32], stato [, timeout [, CALL**
 nomesottoprogramma]]
MULTIWAITFLAG **maschera, flag1[, ..., flag32], stato [, timeout [, nomefunzione]]**

Argomenti

maschera costante o variabile. Maschera dei flag interessati
flag1[,...flag32] nome di dispositivo tipo flag
stato costante predefinita. I valori ammessi sono:
 - **ON** stato flag attivo
 - **OFF** stato flag disattivo
timeout costante o variabile. È il tempo limite di attesa
etichetta etichetta di salto (GOTO)
nomesottoprogra etichetta di sottoprogramma (CALL)
mma
nomefunzione nome della funzione

Descrizione

Attende finché i flag specificati da **flag1...flag32** si trovino nello stato indicato dal parametro **stato** (ON/OFF).
 Di tutti i flag verifica quelli il cui bit nell'argomento **maschera** è abilitato (ON). Il bit 0 dell'argomento **maschera** (quello di peso più basso) corrisponde al bit definito da **flag1**, il bit 1 corrisponde al bit definito da **flag2** e così fino al bit definito dal **flag32**.
 Il parametro **timeout** permette di impostare un timeout diverso da quello di default che è di un secondo. Quando è presente **etichetta** o **nomesottoprogramma** o **nomefunzione**, allo scadere del timeout il programma salta a **etichetta** o chiama **nomesottoprogramma** o **nomefunzione**.

MULTIWAITINPUT

Sintassi

MULTIWAITINPUT **maschera, input1[, ..., input32], stato [, timeout [, GOTO etichetta]]**
MULTIWAITINPUT **maschera, input1[, ..., input32], stato [, timeout [, CALL**
 nomesottoprogramma]]
MULTIWAITINPUT **maschera, input1[, ..., input32], stato [, timeout [, nomefunzione]]**

Argomenti

maschera costante o variabile. Maschera degli input interessati
input1[,...input32
] nome ingresso
stato costante predefinita. I valori ammessi sono:

Descrizione

Disattiva (pone ad OFF) il flag **nomeflag**.

RESETOUT**Sintassi**

RESETOUT **nomeoutput**

Argomenti

nomeoutput nome di dispositivo tipo uscita digitale

Descrizione

Disattiva (pone ad OFF) l'uscita **nomeoutput**.

SETFLAG**Sintassi**

SETFLAG **nomeflag**

Argomenti

nomeflag nome di dispositivo tipo flag

Descrizione

Attiva (pone ad ON) il flag **nomeflag**.

SETOUT**Sintassi**

SETOUT **nomeoutput**

Argomenti

nomeoutput nome di dispositivo tipo uscita digitale

Descrizione

Attiva (pone ad ON) l'uscita **nomeoutput**.

Se l'uscita è configurata di tipo monostabile, viene disattivata automaticamente allo scadere di un timeout fisso di 200 millisecondi.

WAITFLAG**Sintassi**

WAITFLAG **nomeflag, stato [, timeout [, GOTO etichetta]]**
WAITFLAG **nomeflag, stato [, timeout [, CALL nomesottoprogramma]]**
WAITFLAG **nomeflag, stato [, timeout [, nomefunzione]]**

Argomenti

nomeflag nome di dispositivo flag
stato costante predefinita. I valori ammessi sono:
ON stato flag attivo
OFF stato flag disattivo
timeout costante o variabile. È il tempo limite di attesa
etichetta etichetta di salto (GOTO)
nomesottoprogramma etichetta di sottoprogramma (CALL)
nomefunzione nome della funzione

Descrizione

Attende finché il flag **nomeflag** non assume lo stato indicato dal parametro **stato** (ON/OFF).

Se degli argomenti opzionali è presente solo **timeout**, l'errore di ciclo "Flag **nomeflag** in attesa di **stato**" viene generato allo scadere del timeout.

Se la condizione si verifica dopo che il timeout è scaduto, viene automaticamente cancellato l'errore di ciclo inviato in precedenza per quel task.

Quando è presente **etichetta** o **nomesottoprogramma** o **nomefunzione**, allo scadere del timeout il programma salta a **etichetta** o chiama **nomesottoprogramma** o **nomefunzione** senza generare nessuna visualizzazione automatica.

Nota

Al fine di evitare una situazione di attesa di un flag durante un ciclo di lavoro, si consiglia di impostare un timeout.

WAITINPUT

Sintassi

```

WAITINPUT           nomeinput, stato [, timeout [, GOTO etichetta]]
WAITINPUT           nomeinput, stato [, timeout [, CALL nomesottoprogramma]]
WAITINPUT           nomeinput, stato [, timeout [, nomefunzione]]

```

Argomenti

nomeinput nome ingresso

stato costante predefinita. I valori ammessi sono:
 - **ON** stato input attivo
 - **OFF** stato input disattivo

timeout costante o variabile. Tempo limite di attesa

etichetta etichetta di salto (GOTO)

nomesottoprogramm etichetta di sottoprogramma (CALL)

a

nomefunzione nome della funzione

Descrizione

Attende finché l'ingresso **nomeinput** non assume lo stato indicato dal parametro **stato** (ON/OFF). Se non sono specificati argomenti opzionali, passati 20 secondi dall'inizio dell'esecuzione dell'istruzione, viene automaticamente generato l'errore di ciclo: "Input digitale **nomeinput** in attesa di **stato**". Se degli argomenti opzionali è presente solo **timeout**, la visualizzazione sopracitata viene generata allo scadere dello stesso. Se la condizione si verifica dopo che il **timeout** è scaduto, viene automaticamente cancellato l'errore di ciclo inviato in precedenza per quel task. Quando è presente **etichetta** o **nomesottoprogramma** o **nomefunzione**, allo scadere del timeout il programma salta a **etichetta** o chiama **nomesottoprogramma** o **nomefunzione** senza generare nessuna visualizzazione automatica.

Nota

Al fine di evitare una situazione di attesa di un segnale d'ingresso, durante un ciclo di lavoro, si consiglia di impostare un timeout inferiore al tempo di default (20 secondi).

Esempio

[Routine di Azzeramento di un asse](#)

WAITPERSISTINPUT

Sintassi

```

WAITPERSISTINPUT   nomeinput, stato, timepersist [, timeout [, GOTO etichetta]]
WAITPERSISTINPUT   nomeinput, stato, timepersist [, timeout [, CALL
nomesottoprogramma]]
WAITPERSISTINPUT   nomeinput, stato, timepersist [, timeout [, nomefunzione]]

```

Argomenti

nomeinput nome di dispositivo tipo ingresso digitale

stato costante predefinita. I valori ammessi sono:
 - **ON** stato input attivo
 - **OFF** stato input disattivo

timepersist costante o variabile

timeout costante o variabile. Tempo limite di attesa

etichetta etichetta di salto (GOTO)

nomesottoprogram etichetta di sottoprogramma (CALL)

ma

nomefunzione nome della funzione

Descrizione

Attende che l'ingresso **nomeinput** si trovi nello stato indicato dal parametro **stato** (ON/OFF) e rimanga in quello stato per il tempo specificato in **timepersist**. Se non sono specificati argomenti opzionali, passati 20 secondi dall'inizio dell'esecuzione dell'istruzione, viene automaticamente generato l'errore di ciclo: "Input digitale **nomeinput** in attesa di **stato**". Se degli argomenti opzionali è presente solo **timeout**, la visualizzazione sopracitata viene generata allo scadere dello stesso. Se la condizione si verifica dopo che il **timeout** è scaduto, viene automaticamente cancellato l'errore di ciclo inviato in precedenza per quel task. Quando è presente **etichetta** o **nomesottoprogramma** o **nomefunzione**, allo scadere del timeout il programma salta a **etichetta** o chiama **nomesottoprogramma** o **nomefunzione** senza generare nessuna visualizzazione automatica.

Nota

Al fine di evitare una situazione di attesa di un segnale d'ingresso, durante un ciclo di lavoro, si consiglia di impostare un timeout inferiore al tempo di default (20 secondi).

10.3.4 Assi**CHAIN****Sintassi**

CHAIN **asse_master, asse_slave1 [, ...asse_slave5]**

Argomenti

asse_master nome del dispositivo tipo asse che funzionerà da master
asse_slave1...asse_slave5 nome del dispositivo tipo asse che funzionerà da slave

Descrizione

Dopo l'esecuzione di questa istruzione, gli **asse_slave** (1÷5) eseguiranno dei movimenti legati a quelli dell'asse master dal rapporto di concatenamento impostato con l'istruzione **RATIO**. Risulteranno concatenati sia i movimenti punto-punto che quelli interpolati.

L'**asse_slave1** non è un parametro opzionale ma deve sempre essere definito.

L'asse slave, per poter essere concatenato, non deve essere impegnato in una interpolazione e non può essere a sua volta master di altri slave.

L'asse master non può essere a sua volta slave di altri assi.

Il concatenamento può essere effettuato sia con gli assi in quota sia con gli assi in movimento.

Per disabilitare il concatenamento assi è sufficiente eseguire l'istruzione **NORMAL** sull'asse master. Quest'ultima operazione può essere effettuata sia con gli assi in quota che con gli assi in movimento. Quando il concatenamento viene disabilitato durante un movimento degli assi, l'asse slave effettua una rampa di decelerazione e si ferma.

È possibile definire un massimo di 8 assi master contemporaneamente.

L'istruzione è eseguibile anche con assi passo passo (stepper) purchè questi siano tutti gestiti tramite TRS-AX.

Inoltre tutti gli assi devono avere encoder reale e non simulato. In caso opposto viene generato l'errore di sistema "4101- Gestione incongruente dell'asse nomeasse".

Vedere anche [RATIO](#).

Esempio

```

; concateno l'asse Y all'asse X
CHAIN          X, Y
; muovo l'asse X.
; Y replica il movimento di X
MOVINC        X, 100

```

CIRCABS**Sintassi**

CIRCABS **[etichetta],asse1, quota1, asse2, quota2, senso, ±raggio [, angolo]**

Argomenti

etichetta costante o variabile integer. Etichetta che identifica un blocco di spostamento
asse1, asse2 nomi di dispositivi tipo asse
quota1, quota2 costante o variabile. Rappresenta la quota di spostamento assoluto
senso variabile integer. Specifica il tipo di rotazione, sono ammessi i valori:
CW oraria
CCW antioraria
raggio costante o variabile. Rappresenta il valore del raggio del cerchio
angolo costante o variabile. Rappresenta l'angolo di partenza

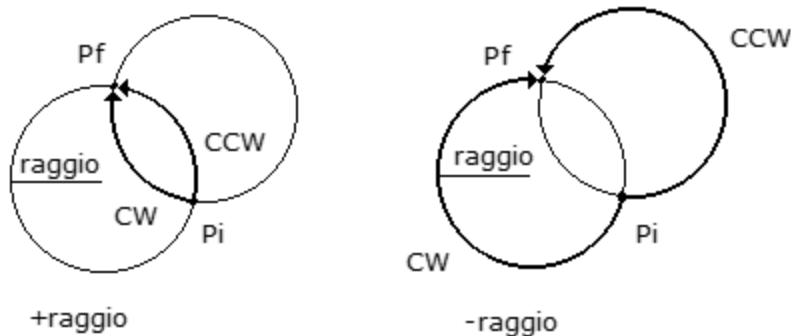
Descrizione

Interpolazione circolare a 2 assi con *spostamento assoluto* basato sulle quote **quota1** e **quota2** programmate.

L'arco è determinato dal punto iniziale (punto corrente), dal punto finale, dal valore del **raggio** e dal **senso** di percorrenza.

Il segno applicato al **raggio** permette di selezionare l'arco minore (+raggio) o l'arco maggiore (-raggio).

Nel caso particolare di quota iniziale dell'asse 1 coincidente con la quota finale **quota1** e la quota iniziale dell'asse2 coincide con la quota finale **quota2** si individua un cerchio completo: in questo caso è necessario indicare l'argomento **angolo**, con lo stesso significato dell'istruzione [CIRCLE](#) (a cui fare riferimento). Il parametro angolo serve per determinare in maniera univoca il centro del cerchio, con lo stesso significato dell'istruzione [CIRCLE](#). Viene considerato solo quando, prima dell'esecuzione dell'istruzione, **quota1** e **quota2** corrispondono alle quote correnti degli assi. Il parametro opzionale **etichetta** è utilizzato in associazione alla istruzione [SETLABELINTERP](#) per identificare in modo univoco il blocco di spostamento. Assi di tipo passo-passo possono essere utilizzati in questa istruzione solo se gestiti da un remoto TRS-AX. In questo caso è da tenere presente che col termine interpolazione si intende il movimento coordinato di più assi affetto da un errore discreto dovuto al metodo di pilotaggio dell'asse.



CIRCINC

Sintassi

CIRCINC

[**etichetta**],asse1, **quota1**, asse2, **quota2**, **senso**, ±**raggio** [, **angolo**]

Argomenti

etichetta	costante o variabile integer. Etichetta che identifica un blocco di spostamento
asse1, asse2	nomi di dispositivi tipo asse
quota1, quota2	costante o variabile. Rappresenta la quota di spostamento incrementale
senso	variabile integer. Specifica il tipo di rotazione, sono ammessi i valori: CW oraria CCW antioraria
raggio	costante o variabile. Rappresenta il valore del raggio del cerchio
angolo	costante o variabile. Rappresenta l'angolo di partenza

Descrizione

Interpolazione circolare a 2 assi con *spostamento incrementale* basato sulle quote **quota1** e **quota2** programmate.

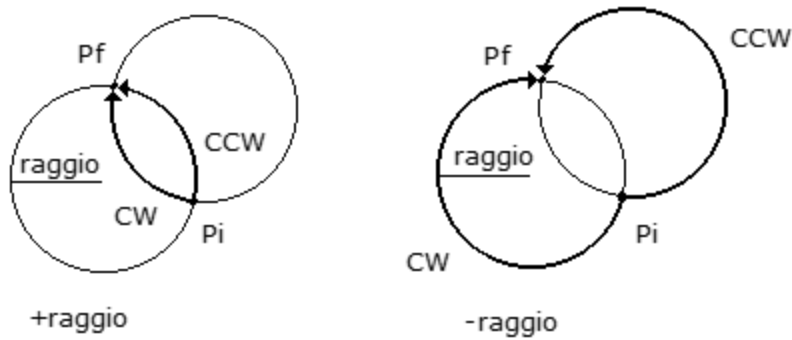
L'arco è determinato dal punto iniziale (punto corrente), dal punto finale, dal valore del **raggio** e dal **senso** di percorrenza.

Il segno applicato al **raggio** permette di selezionare l'arco minore (+raggio) o l'arco maggiore (-raggio).

Nel caso particolare di $quota1 = quota2 = 0$, si individua un cerchio completo: in questo caso è necessario indicare l'argomento **angolo**, con lo stesso significato dell'istruzione [CIRCLE](#) (a cui fare riferimento).

Il parametro angolo serve per determinare in maniera univoca il centro del cerchio, con lo stesso significato dell'istruzione [CIRCLE](#). Il parametro opzionale **etichetta** è utilizzato in associazione alla istruzione [SETLABELINTERP](#) per identificare in modo univoco il blocco di spostamento.

Assi di tipo passo-passo possono essere utilizzati in questa istruzione solo se gestiti da un remoto TRS-AX. In questo caso è da tenere presente che col termine interpolazione si intende il movimento coordinato di più assi affetto da un errore discreto dovuto al metodo di pilotaggio dell'asse.



CIRCLE

Sintassi

CIRCLE

[etichetta],asse1, asse2, senso, raggio, angolo

Argomenti

etichetta	costante o variabile integer. Etichetta che identifica un blocco di spostamento
asse1, asse2	nomi di dispositivi tipo asse
senso	variabile integer. Specifica il tipo di rotazione, sono ammessi i valori: CW oraria CCW antioraria
raggio	costante o variabile. Rappresenta il valore del raggio del cerchio
angolo	costante o variabile. Rappresenta l'angolo di partenza

Descrizione

Interpolazione circolare completa.

Genera un cerchio con **asse1** ed **asse2**, nel **senso** indicato, con **raggio** pari al valore assegnato ed in base all'**angolo** di partenza imposto.

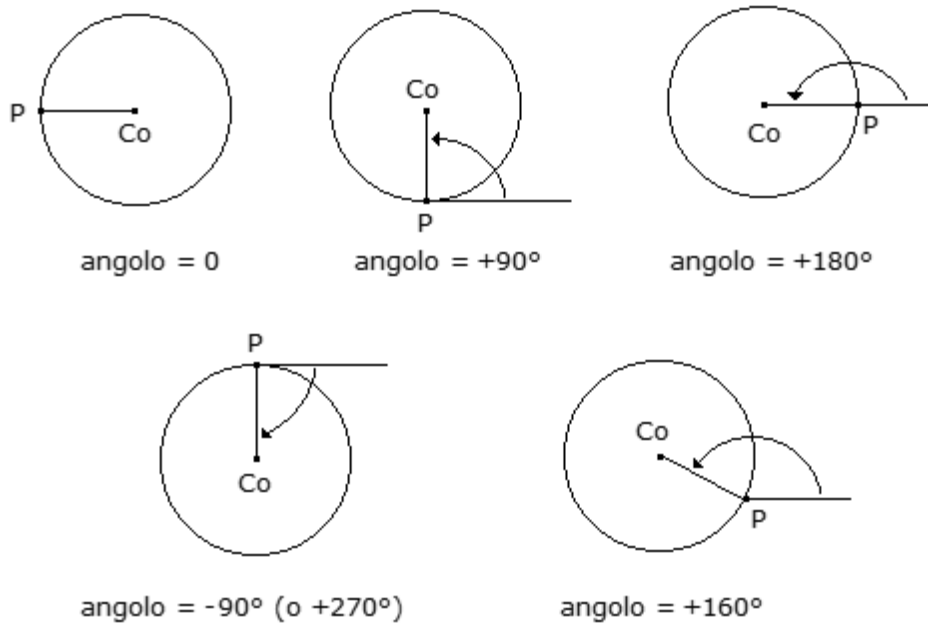
Raggio può assumere solo valori positivi.

L'**angolo** va indicato con convenzione trigonometrica positivo in senso antiorario partendo dall'asse X.

Rimane infatti determinata la posizione del centro C_0 del cerchio, specificando l'angolo formato dal raggio passante per il punto iniziale P programmato (punto corrente) e la direzione orizzontale X+. Il parametro opzionale **etichetta** è utilizzato in associazione alla istruzione [SETLABELINTERP](#) per identificare in modo univoco il blocco di spostamento.

Assi di tipo passo-passo possono essere utilizzati in questa istruzione solo se gestiti da un remoto TRS-AX.

In questo caso è da tenere presente che col termine interpolazione si intende il movimento coordinato di più assi affetto da un errore discreto dovuto al metodo di pilotaggio dell'asse.



COORDIN

Sintassi

COORDIN
 _colonna_asse1

matrice, valore deltaT, direzione, ini, fin, mask, asse1, n°

[, (asse2, n°_colonna_asse2) ÷ (asse32, n°_colonna_asse32)]

Argomenti

matrice	matrice di dati
valore deltaT	costante o variabile. Base tempi
direzione	costante predefinita. Direzione di lettura dati in matrice UP da riga più bassa a più alta DOWN da riga più alta a più bassa
ini	variabile globale integer. È il numero di riga iniziale
fin	variabile globale integer. È il numero di riga finale
mask	maschera assi da abilitare
asse1 [...asse32]	nomi di dispositivi tipo asse
n°	numero colonna matrice riferita all' asse
_colonna_asse1[...n_colonna_asse32]	

Descrizione

Questa istruzione consente di effettuare movimenti sincronizzati degli assi **asse1**, **asse2**, etc., attraverso spostamenti incrementali (microvettori) definiti da una **matrice** di dati.

I parametri **asse1** e **n_colonna_asse1** devono essere sempre definiti.

I valori contenuti nella **matrice** indicano le quote assolute via via raggiunte dai vari assi.

I relativi spostamenti incrementali (differenza quote tra riga (n) e riga (n-1)) vengono eseguiti in un intervallo temporale pari ad un **multiplo** della base tempi (1 ms = real-time di rinfresco assi), definito dall'argomento

valore Δt che deve, quindi, essere espresso da un numero intero.

Prefissato tale valore di tempo, l'entità dei singoli spostamenti di ogni asse ne determina la velocità. Questa istruzione consente la movimentazione coordinata di un massimo di 32 assi lungo qualsivoglia percorso curvilineo nello spazio, come generati da tecniche SPLINE.

L'istruzione è passante e non necessita della istruzione STARTINTERP per partire. Bisogna, tuttavia, porre una istruzione WAITSTILL al suo termine per attendere il corretto arrivo in quota degli assi. Eventuali modifiche del feedrate override devono essere fatte tramite l'istruzione SETFEEDI e gestite tramite l'istruzione SETFEEDCOORD.

Il parametro **direzione** permette di determinare la direzione di percorrenza della matrice, consentendo l'esecuzione della traiettoria in entrambi i versi.

Le colonne della matrice da scandire possono essere di tipo float o di tipo double ma non entrambi contemporaneamente.

Oltre alla movimentazione degli assi lungo un percorso finito (definito dal numero di righe della matrice) è possibile realizzare una movimentazione infinita utilizzando:

- una matrice di una sola riga. In questa modalità di funzionamento il controllo legge sempre l'unica riga della matrice ed applica agli assi le quote in essa riportate. Affinché gli assi si muovano sarà necessario modificare opportunamente la riga della matrice, preferibilmente utilizzando un task real-time che garantisce la sincronizzazione dell'aggiornamento delle quote con la frequenza di rinfresco degli assi. In questo modo si possono realizzare camme elettroniche o concatenamenti con rapporto diverso da 1:1. Per attivare questa modalità di funzionamento si dovranno impostare **ini** = 1, **fin** = 0 e **direzione** = UP. Quando si utilizza questa modalità di funzionamento NON va utilizzata l'istruzione STOP.
- una matrice di più righe. È possibile scandire la matrice con cicli dalla prima all'ultima riga indefinitamente impostando i valori **ini** = 1, **fin** = 0 e **direzione** = UP. Nel caso in cui si voglia eseguire una sola riga di una matrice multirighe, è necessario impostare i parametri **ini**, **fin** e **direzione** nel seguente modo: **ini** = numero riga che si vuole eseguire, **fin** = numero riga precedente quella che si vuole eseguire, **direzione** = UP. In altro caso viene generato un errore di sistema.

Assi di tipo passo-passo possono utilizzati in questa istruzione solo se gestiti da un remoto TRS-AX.

DISABLECORRECTION

Sintassi

DISABLECORRECTION **asse [, asse1, ..., asse6]**

Argomenti

asse nome di dispositivo tipo asse
asse1, ..., asse6 nome di dispositivo tipo asse

Descrizione

Disabilita la correzione lineare per l'**asse** specificato.

Il primo parametro è l'asse per cui si vuole disattivare la correzione, se è l'unico parametro specificato verranno disattivate tutte le correzioni presenti in configurazione. I successivi parametri permettono di specificare quali correzioni disattivare, se uno di questi coincide con il primo parametro verrà disattivata l'autocorrezione.

Per una descrizione più dettagliata vedere [ENABLECORRECTION](#).

Esempio

; disabilita la sola autocorrezione per l'asse X
DISABLECORRECTION X, X

; disabilita la correzione incrociata (verso X e Y) per l'asse Z, ma non l'autocorrezione
DISABLECORRECTION Z, X, Y

EMERGENCYSTOP

Sintassi

EMERGENCYSTOP **asse, tempo**

Argomenti

asse nome di dispositivi tipo asse
tempo costante o variabile integer. Tempo di rampa in ms

Descrizione

Arresta l'asse specificato e eventualmente con lui tutti quelli coinvolti nel movimento interpolato. Il movimento viene fermato con una rampa di decelerazione nel tempo indicato dalla variabile **[tempo]**.

Nei movimenti punto punto se il tempo impostato è superiore al tempo di decelerazione indicato in configurazione, viene utilizzato quest'ultimo.

Nei movimenti interpolati se il tempo impostato è superiore al massimo dei tempi di decelerazione di tutti gli assi coinvolti, viene utilizzato il tempo massimo impostato in configurazione.

Il movimento potrà essere ripreso da una istruzione [START](#).

L'istruzione non può essere utilizzata se **[asse]** è un asse slave.

L'istruzione può generare i seguenti errori di sistema:

- "4101 - Gestione incongruente dell'asse" quando **[asse]** sta eseguendo un movimento sincronizzato o un'interpolazione multilineare o un movimento ISO.
- "4105 - istruzione non eseguibile sull'asse" quando **[asse]** è un asse di conteggio.
- "4399 - parametro fuori range" se il **[tempo]** indicato è uguale o inferiore a 0.

ENABLECORRECTION

Sintassi

ENABLECORRECTION **asse [, asse1, ..., asse6]**

Argomenti

asse nome di dispositivo tipo asse
asse1, ..., asse6 nome di dispositivo tipo asse

Descrizione

Abilita la correzione lineare per l'**asse** specificato. La correzione è composta dall'autocorrezione e dalla correzione incrociata. L'autocorrezione è una correzione della quota reale di un asse in funzione della propria posizione, la correzione incrociata è una correzione della quota reale di un asse in funzione della posizione di altri assi. È possibile definire un massimo di cinque correttori incrociati.

Il primo parametro è l'asse per cui si vuole attivare la correzione, se è l'unico parametro specificato verranno attivate tutte le correzioni presenti in configurazione.

I successivi parametri permettono di specificare quali correzioni attivare, se uno di questi coincide con il primo parametro verrà attivata l'autocorrezione.

Vedere anche [DISABLECORRECTION](#).

NOTA: Affinché l'istruzione abbia effetto la correzione deve essere abilitata anche in configurazione.

Esempio

; abilita tutte le correzioni previste in configurazione per l'asse X
ENABLECORRECTION X

; abilita la sola autocorrezione per l'asse X
ENABLECORRECTION X, X

; abilita l'autocorrezione e la correzione incrociata (verso X eY) per l'asse Z
ENABLECORRECTION Z, X, Y, Z

ENDMOV

Sintassi

ENDMOV **asse [, quota]**

Argomenti

asse nome di dispositivi tipo asse
quota costante o variabile.

Descrizione

Arresta il movimento dell'asse specificato. Si differenzia da una istruzione [STOP](#), in quanto il movimento viene terminato e non potrà più essere ripreso da una eventuale istruzione [START](#). Se specificato il parametro **quota** permette di impostare la posizione in cui l'asse terminerà il movimento, altrimenti il punto in cui l'asse si ferma dipende dalla velocità corrente e dall'ultima decelerazione programmata. Se necessario, per raggiungere il punto di fine movimento, il controllo effettua una inversione di moto dell'asse.

Nota

Il parametro quota viene utilizzato solo se si tratta di un movimento punto-punto. In caso di movimento interpolato, il movimento dell'asse si arresta senza tener conto del valore della **quota**.

Esempio

; ferma il movimento corrente e porta l'asse a quota 0.0
ENDMOV X, 0.0

FASTREAD

Sintassi

FASTREAD **asse1, stato, variabile1 [,asse2, variabile2],[..., asse8, variabile8]**

Argomenti

asse1...[...asse8] nomi di dispositivi tipo asse. Asse1 è l'asse master.
stato costante predefinita. Assume i seguenti valori:
ON fronte di salita
OFF fronte di discesa
variabile1...[...variabile8] variabile o elemento di matrice/vettore di tipo double. Quota memorizzata

le eventuali quote **asse4**, **asse5**, **asse6**) si riferisce allo spostamento totale (non quindi allo spostamento/giro). Il parametro opzionale **etichetta** è utilizzato in associazione alla istruzione [SETLABELINTERP](#), per identificare in modo univoco il blocco di spostamento.

Assi di tipo passo-passo possono essere utilizzati in questa istruzione, solo se gestiti da un remoto TRS-AX. In questo caso, è da tenere presente che con il termine interpolazione si intende il movimento coordinato di più assi affetto da un errore discreto dovuto al metodo di pilotaggio dell'asse.

Nota

1. La condizione di contornatura viene valutata solo sui primi tre assi, quelli che formano il sistema di riferimento. Aggiungendone un successivo, ed eventualmente, modificandolo, si ha una errata gestione del profilo di velocità. Per ottenere un movimento corretto, occorre interporre una istruzione [WAITSTILL](#) tra una istruzione HELICABS e l'altra.
2. Se si imposta un sistema di riferimento locale utilizzando l'istruzione [SETRIFLOC](#), i tre assi, che definiscono il nuovo sistema di riferimento, devono sempre essere indicati tra i parametri della istruzione HELICABS, anche se non eseguono spostamenti.

HELICINC

Sintassi

HELICINC **[etichetta], asse1, quota1, asse2, quota2, asse3, quota3, senso, ± raggio [,angolo [, numgiri [, asse4, quota4 [, ..., asse6, quota6]]]]**

Argomenti

etichetta	costante o variabile integer. Etichetta che identifica un blocco di spostamento
asse1...asse3[...asse6]	nomi di dispositivi tipo asse
quota1...quota3[...quota6]	costante o variabile. Quota di spostamento incrementale
senso	variabile integer. Tipo di rotazione oraria/antioraria (CW/CCW)
raggio	costante o variabile. Raggio dell'elica
angolo	costante o variabile. Angolo di partenza
numgiri	costante o variabile. Numero di giri

Descrizione

Interpolazione elicoidale con spostamento incrementale pari alle quote **quota1**, **quota2** e **quota3** programmate.

Il movimento è costituito da una interpolazione circolare associata agli assi **asse1** e **asse2** (con le stesse regole sintattiche di [CIRCABS](#) /[CIRCINC](#), relativamente agli argomenti **senso**, **±raggio** e **angolo**), e da una lineare associata all'**asse3** (eventualmente, **asse4**, **asse5** e **asse6**). Il movimento elicoidale può essere sviluppato su più giri indicati dall'argomento **numgiri**.

La quota relativa all'asse con movimento lineare (come le eventuali quote **asse4**, **asse5**, **asse6**) si riferisce allo spostamento totale (non quindi allo spostamento/giro). Il parametro opzionale **etichetta** è utilizzato in associazione alla istruzione [SETLABELINTERP](#) per identificare in modo univoco il blocco di spostamento. Assi di tipo passo-passo possono essere utilizzati in questa istruzione, solo se gestiti da un remoto TRS-AX. In questo caso, è da tenere presente che con il termine interpolazione si intende il movimento coordinato di più assi affetto da un errore discreto dovuto al metodo di pilotaggio dell'asse.

Nota

1. La condizione di contornatura viene valutata solo sui primi tre assi, quelli che formano il sistema di riferimento. Aggiungendone un successivo, ed eventualmente modificandolo, si ha una errata gestione del profilo di velocità. Per ottenere un movimento corretto, occorre interporre una istruzione [WAITSTILL](#) tra una istruzione HELICINC e l'altra.
2. Se si imposta un sistema di riferimento locale utilizzando l'istruzione [SETRIFLOC](#), i tre assi, che definiscono il nuovo sistema di riferimento, devono sempre essere indicati tra i parametri della istruzione HELICINC, anche se non eseguono spostamenti.

JERKCONTROL

Sintassi

JERKCONTROL **asse, stato**

Argomenti

asse	nomi di dispositivi tipo asse
stato	costante predefinita. I valori ammessi sono: ON stato flag attivo OFF stato flag disattivo

Descrizione

Abilita, se al parametro **stato** viene assegnato valore ON, o disabilita, se al parametro **stato** viene assegnato valore OFF, il controllo del jerk sui movimenti in interpolazione e punto-punto dell'**asse**. Il controllo del jerk


```

setvel X, 20
setvel X, 20
movabs X, 100, Y, 200
movabs X, 150, Y, 180
setvel X, 5
movabs X, 80, Y, 100
waitstill X, Y
fret

```

MOVINC

Sintassi

MOVINC **asse1, valore1 [, asse2, valore2 [, ..., asse6, valore6]]**

Argomenti

asse1...[...asse6] nomi di dispositivi tipo asse
valore1...[...valore6] costante o variabile. Valore di spostamento incrementale

Descrizione

Fa eseguire ad ogni asse un *movimento incrementale* pari al corrispondente **valore**. Per eseguire il movimento l'asse non deve essere impegnato in un movimento interpolato e si deve trovare in quota o in finestra. Il movimento dell'asse ha inizio non appena viene eseguita l'istruzione. Se vengono eseguite più istruzioni di movimentazione punto-punto in un medesimo task, queste vengono concatenate. Se un secondo task cerca di eseguire delle istruzioni punto-punto su di un asse che è già impegnato in un movimento, questo task rimarrà in attesa che il movimento comandato dal primo task termini. È inoltre possibile modificare la velocità tra un movimento punto-punto ed il successivo con una istruzione [SETVEL](#). I due movimenti verranno raccordati con una rampa di velocità senza fermare gli assi. Se non viene utilizzata l'istruzione [SETVEL](#), la velocità massima possibile è rappresentata dal valore di velocità manuale definito in configurazione. Un movimento punto-punto può essere interrotto con l'istruzione [STOP](#) e successivamente ripreso con l'istruzione [START](#). Durante l'interruzione del movimento l'asse rimane in stato "a regime" anche se fisicamente non si muove. Un movimento può essere interrotto con l'istruzione [ENDMOV](#). In questo caso non può più essere ripreso.

Nota

- 1) In precedenza i movimenti punto-punto:
 - non permettevano la variazione della velocità se non ad asse fermo. Il comportamento attuale è simile a quello dei movimenti interpolati.
 - quando interrotti da una STOP l'asse corrispondente assumeva lo stato "in quota".
- 2) È consigliato utilizzare istruzioni di interpolazione lineare invece di istruzioni di movimento punto punto, quando il numero dei blocchi di spostamento è superiore a 32 e i blocchi sono costituiti da microtratti. Per approfondimenti richiedere a TPA il documento "Limiti Firmware Movimento Punto Punto.doc".

Esempio 1

[Routine di Azzeramento di un asse](#)

Esempio 2

```

; cambio di velocità
Function cambiovel
  setvel X, 20
  setvel Y, 20
  movinc X, 100, Y, 200
  movinc X, 150, Y, 180
  setvel X, 5
  movinc X, 80, Y, 100
  waitsti X, Y
}
fret

```

MULTIABS

Sintassi

MULTIABS **[etichetta],asse1, valore1, [asse2, valore2 [, asse3, valore3 [, ..., asse16, valore 16]]]**

Argomenti

etichetta costante o variabile integer. Etichetta che identifica un blocco di spostamento

asse1...asse16 nomi di dispositivi tipo asse
valore1...[...valore16] costante o variabile. Valore della quota teorica di fine blocco di spostamento

Descrizione

Interpolazione multi-lineare assoluta fino a 16 assi. Questo movimento d'interpolazione permette l'anticipo dei profili di velocità, impostando sugli assi le rispettive tolleranze, tramite l'istruzione [SETTOLERANCE](#) (per tolleranza di un asse si intende la porzione di percorso nella quale può non esistere un rapporto di interpolazione costante). L'ordine di inserimento degli assi nella istruzione MULTIABS **deve** sempre essere il medesimo e **tutti** gli assi coinvolti nel movimento devono essere presenti. I blocchi di spostamento vengono accodati nel normale lookahead e il movimento parte all'esecuzione di una istruzione [WAITSTILL](#), [STARTINTERP](#) o al riempimento del lookahead stesso. Tra gli assi coinvolti nel movimento uno può essere usato come collider tramite l'istruzione [WAITCOLL](#). Il parametro opzionale **etichetta** è utilizzato in associazione alla istruzione [SETLABELINTERP](#) per identificare in modo univoco il blocco di spostamento. Assi di tipo passo-passo possono essere utilizzati in questa istruzione solo se gestiti da un remoto TRS-AX. In questo caso è da tenere presente che col termine interpolazione si intende il movimento coordinato di più assi affetto da un errore discreto dovuto al metodo di pilotaggio dell'asse.

Nota

Con questo tipo d'interpolazione non è possibile utilizzare sistemi di riferimento virtuali (istruzione [SETRIFLOC](#) e [RESRIFLOC](#)). È possibile eseguire dei movimenti con assi concatenati (in [CHAIN](#)). Gli assi coinvolti nel moto interpolato multiasse devono essere dichiarati master di altri assi non coinvolti nel movimento. Inoltre è possibile applicare il FeedRate Override.

Esempio

```

setquote      x, 0
setquote      y, 0
setquote      z, 0
; primo blocco
setveli       x, velx1
setveli       y, vely1
setveli       z, velz1
multiabs      x, quotax1, y,quotay1, z,quotaz1
; secondo blocco
settolerance  x,tollx2, y,tolly2, z,tollz2
setveli       x, velx2
setveli       y, vely2
setveli       z, velz2
multiabs      x,quotax2, y,quotay2, z,quotaz2
;terzo blocco
settolerance  x,tollx3, y,tolly3, z,tollz3
setveli       x, velx3
setveli       y, vely3
setveli       z, velz3
multiabs      x,quotax3, y,quotay3, z,quotaz3
;quarto blocco
settolerance  x,tollx4, y,tolly4, z,tollz4
setveli       x, velx4
setveli       y, vely4
setveli       z, velz4
multiabs      x,quotax4, y,quotay4, z,quotaz4
waitstill     x, y, z

```

MULTIINC

Sintassi

MULTIINC **[etichetta],asse1, valore1, [asse2, valore2 [, asse3, valore3 [, ..., asse16, valore 16]]]**

Argomenti

etichetta costante o variabile integer. Etichetta che identifica un blocco di spostamento
asse1...asse16 nomi di dispositivi tipo asse
valore1...[...valore16] costante o variabile. Valore di incremento della quota teorica di fine blocco di spostamento

Descrizione

Interpolazione multi-lineare incrementale fino a 16 assi. Questo movimento d'interpolazione permette l'anticipo dei profili di velocità, impostando opportunamente tramite l'istruzione [SETTOLERANCE](#) sugli assi le rispettive

SETQUOTECHAINSTRAT

Sintassi

SETQUOTECHAINSTRAT **asse, [valore]**

Argomenti

asse nome di dispositivo tipo asse
valore Variabile tipo integer. I valori ammessi sono:
 0 = solo l'asse master inizializza alla nuova quota, l'asse slave mantiene la quota precedente
 diverso da 0 = le quote dell'asse slave vengono inizializzate in modo sincrono con le quote dell'asse master

Descrizione

Questa istruzione permette di impostare il modo in cui l'asse slave indicato si comporterà a fronte di una istruzione [SETQUOTE](#) sul master.

L'istruzione va eseguita sull'asse slave.

Se la variabile **valore** è omessa, viene assunto un valore di default pari a 0.

SETRIFLOC

Sintassi

SETRIFLOC **quota1_ax1, quota2_ax1, quota3_ax1,
 quota1_ax2, quota2_ax2, quota3_ax2,
 quota1_ax3, quota2_ax3, quota3_ax3,
 asse1, asse2, asse3**

Argomenti

quota1_ax1...quota3_ax3 coseni direttori dei tre assi
asse1...asse3 nomi di dispositivi. Tipo asse

Descrizione

Consente di attivare un sistema di riferimento cartesiano X' Y' Z' rototraslato rispetto al sistema di riferimento assoluto di macchina X Y Z, rappresentato dagli assi fisici **asse1**, **asse2** e **asse3**.

I nove argomenti indicano i Coseni Direttori dei tre assi locali rispetto a quelli assoluti,

$\cos\alpha_1$	$\cos\beta_1$	$\cos\gamma_1$
$\cos\alpha_2$	$\cos\beta_2$	$\cos\gamma_2$
$\cos\alpha_3$	$\cos\beta_3$	$\cos\gamma_3$

che costituiscono la matrice di trasformazione delle coordinate.

L'origine del nuovo sistema di riferimento viene presa nel punto corrente.

Tutte le istruzioni di movimento interpolato, interessanti gli assi X, Y e Z, vengono riferite a questo sistema di riferimento, fino all'esecuzione dell'istruzione [RESRIFLOC](#).

SETTOLERANCE

Sintassi

SETTOLERANCE **asse1, valore1, [asse2, valore2 [, asse3, valore3 [, ..., asse16, valore16]]]**

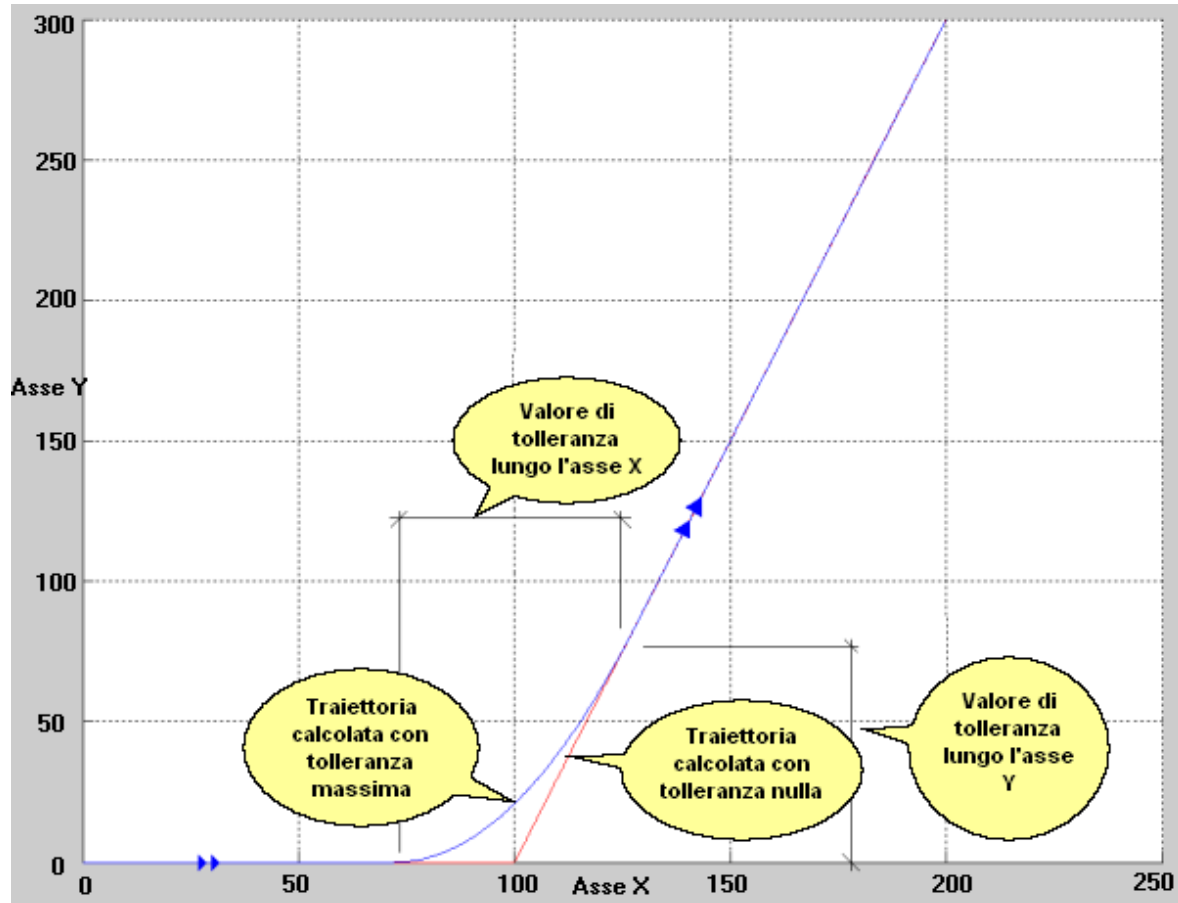
Argomenti

asse1...asse16 nomi di dispositivi tipo asse
valore1...[...valore16] costante o variabile. Valore della tolleranza massima applicabile all'asse

Descrizione

Imposta per ogni **asse** definito il **valore** di tolleranza da applicare nei movimenti di interpolazione multiasse. Il valore di tolleranza è il valore di spostamento per il quale l'asse si discosta dalla traiettoria originale in una interpolazione multiasse.

La tolleranza va impostata per ogni **asse** coinvolto nell'interpolazione, e il sistema provvederà all'anticipo dei profili di velocità rispettando le tolleranze su tutti gli assi e non oltrepassando lo spazio di rampa, che rappresenta il limite massimo per l'anticipazione dei profili. La mancata assegnazione della tolleranza prima di una istruzione multiasse comporta l'applicazione dell'ultima tolleranza impostata sull'asse stesso. Qualora non fosse mai stato assegnato un valore di tolleranza ad un determinato asse, lo stesso viene considerato a tolleranza nulla. In questo caso ogni movimento multiasse che coinvolgerà quell'asse non prevederà nessuna anticipazione di rampa. Si consiglia quindi il sistematico utilizzo di questa istruzione per ogni asse coinvolto, prima di una istruzione di un movimento di interpolazione multiasse.



Nella figura qui sopra è rappresentata una classica traiettoria multi-asse, composta da due blocchi di movimento il primo dei quali consiste in uno spostamento di 100 dell'asse X, mentre il secondo consiste in un movimento dell'asse Y di 300 e uno sempre di 100 dell'asse X. La linea rossa evidenzia la traiettoria in caso di tolleranza nulla, mentre quella blu è la traiettoria in caso di tolleranza massima.

La tolleranza può essere vista anche come l'area sottesa dal profilo di velocità durante l'intervallo di anticipazione, come mostra la figura di seguito.

Argomenti

asse nome di dispositivo tipo asse

Descrizione

Arresto del movimento dell'**asse**. L'asse eseguirà una rampa di decelerazione la cui durata dipende dalla velocità attuale e dai parametri di configurazione.

Esempio

[Routine di Azzeramento di un asse](#)

SWITCHENC**Sintassi**

SWITCHENC **asse1, [asse2, [direzione, quota]]**

Argomenti

asse1 nome del dispositivo tipo asse
asse2 nome del dispositivo tipo asse. Indica un asse di conteggio costante predefinita.
direzione **UP**=scambio encoder al superamento della quota in direzione positiva
DOWN=scambio encoder al superamento della quota in direzione negativa
quota costante double o variabile double

Descrizione

Permette di sostituire all'encoder di **asse1** l'encoder di **asse2**. Lo scambio dell'encoder viene effettuato al superamento in **direzione** positiva (UP) o negativa (DOWN) della **quota** indicata.

Se i parametri **direzione** e **quota** sono omessi, lo scambio dell'encoder viene subito eseguito, indipendentemente dalla posizione degli assi.

Se viene dichiarato solo **asse1** viene ripristinato il funzionamento a singolo encoder.

Asse1 non può essere di tipo passo-passo, conteggio e virtuale, **Asse2** può essere solo un asse di conteggio. Inoltre, sia **asse1** sia **asse2** non possono essere impegnati in movimenti in chain come asse slave.

L'istruzione genera l'errore di sistema 4101 - Gestione incongruente dell'asse NomeAsse, quando **asse1** o **asse2** è dichiarato slave in un movimento in chain oppure quando **asse1** sta eseguendo una istruzione FASTREAD o una istruzione SETPFLY. Inoltre, può essere generato l'errore di sistema 4105 - Istruzione non eseguibile sull'asse NomeAsse, quando il tipo di asse dichiarato non è uno tra quelli possibili.

WAITACC**Sintassi**

WAITACC **asse1 [, ..., asse6]**

Argomenti

asse1 [, ...,asse6] nome del dispositivo tipo asse

Descrizione

Attesa dello stato di accelerazione o uno degli stati successivi su tutti gli **assi** (1÷6) specificati.

Il task, in cui viene eseguita l'istruzione, viene messo in attesa, fino a quando l'asse si trova nello stato di accelerazione o in uno degli stati successivi.

Gli stati dell'asse sono identificati da un numero intero:

- accelerazione = 1
- regime = 2
- decelerazione = 3
- quota = 4
- attesa finestra grande = 5
- attesa asse fermo = 6
- attesa finestra piccola = 7

WAITCOLL**Sintassi**

WAITCOLL **asse, valore, timeout, delta**

Argomenti

asse	nome del dispositivo tipo asse
valore	costante o variabile. Valore di quota assoluta
timeout	costante o variabile. È il tempo di attesa ad asse fermo
delta	costante o variabile. È il valore di finestra per ottenere asse fermo

Descrizione

Durante il movimento di un asse il raggiungimento di una quota programmata può essere impedito da un ostacolo di natura meccanica, rappresentato a volte anche dallo stesso pezzo in lavorazione. Il sistema in questo caso genera un errore di sistema di "servoerror" o "movimento non finito". Con questa istruzione si definisce un **valore** di quota in corrispondenza della quale il sistema inizia a verificare la presenza di una collisione, il tempo di attesa (**timeout**) prima che l'**asse**, dopo l'avvenuta collisione, venga posto in stato "quota" e il **delta** che definisce la tolleranza sul posizionamento dell'asse. Quando l'asse supera la quota definita in **valore**, viene verificato se l'asse sia ancora in movimento. Intercettato l'ostacolo, viene individuata la situazione critica e, pur garantendo la spinta del motore, non viene più verificato il superamento del limite dell'errore d'anello. La direzione di moto su cui viene verificata l'avvenuta collisione è la stessa direzione dell'ultimo movimento accodato. Il **timeout** è espresso in secondi, il valore **delta** deve essere maggiore di 0.001 mm e minore della differenza tra la quota di arrivo programmata e la quota **valore**. L'istruzione può essere utilizzata con l'interpolatore multiasse dal momento che in tale interpolatore si accetta la perdita temporanea del legame di interpolazione.

L'istruzione è applicabile anche ad assi Master di un movimento in Chain.

Viene generato un errore di sistema quando:

- l'**asse** sta eseguendo un movimento interpolato classico (vedi istruzioni LINEARBS, LINEARINC, CIRCABS, CIRCINC, HELICABS, HELICINC) o in moto coordinato o movimenti interpolati ISO.
- l'**asse** è un asse slave
- l'**asse** è un asse di conteggio o un asse passo-passo o un asse virtuale
- il **valore** impostato è superiore alla quota di fine movimento

Esempio

```

; imposta la quota dell'asse X
SETQUOTE          X, 0.0
; muove l'asse X alla quota assoluta 1000
MOVABS           X, 1000.0
; attende la quota di collisione , attende 2 secondi
; prima di porre l'asse in stato "quota" dopo
; aver intercettato una collisione con una precisione di 0.01 mm
WAITCOLL         X, 980.0,2.0,0.01

```

WAITDEC**Sintassi**

WAITDEC **asse1 [, ..., asse6]**

Argomenti

asse1 [...,asse6] nome del dispositivo tipo asse

Descrizione

Attesa dello stato di decelerazione o uno degli stati successivi su tutti gli **assi** (1÷6) specificati.

Il task, in cui viene eseguita l'istruzione, viene messo in attesa, fino a quando l'asse si trova negli stati di decelerazione, quota, attesa finestra grande, attesa finestra piccola e attesa asse fermo.

Gli stati dell'asse sono identificati da un numero intero:

- accelerazione = 1
- regime = 2
- decelerazione = 3
- quota = 4
- attesa finestra grande = 5
- attesa asse fermo = 6
- attesa finestra piccola = 7

WAITREG**Sintassi**

WAITREG **asse1 [, ..., asse6]**

Argomenti

asse1 [...,asse6] nome del dispositivo tipo asse

costante	descrizione	tipo	valore di ritorno
_CFGVMAXD	velocità massima in manuale	f	m/1' o inch/1" o gradi/1" o giri/1'
_CFGVMAXI	velocità massima di interpolazione	f	m/1' o inch/1" o gradi/1" o giri/1'
_CFGPHINV	inversione fasi encoder	b	0=no inversione, 1=inversione
_CFGRFINV	inversione riferimento	b	0=no inversione, 1=inversione
_CFGZIND	abilitazione reset quota tacca di zero	b	0=disabilitato, 1=abilitato
_CFGSLOPE	tipo di rampa per accelerazioni/decelerazioni in punto-punto	b	0=lineare, 1=a 'S' , 2= a doppia esse
_CFGKFFA	feed forward di accelerazione	f	
_CFGKFFAI	feed forward d'interpolazione di accelerazione	f	
_CFGSRPP	velocità inizio rampa passo passo	f	m/1' o inch/1" o gradi/1" o giri/1'
_CFGACC	tempo di accelerazione da 0 a _CFGVMAX	i	msec
_CFGDEC	tempo di decelerazione da _CFGVMAX a 0	i	msec
_CFGACCI	tempo di accelerazione da 0 a _CFGVMAXI	i	msec
_CFGDECI	tempo di decelerazione da _CFGVMAXI a 0	i	msec
_CFGQLP	limite asse in positivo	d	quota
_CFGQLN	limite asse in negativo	d	quota
_CFGKPP	coefficiente proporzionale	f	
_CFGKI	coefficiente integrativo	f	
_CFGKD	coefficiente derivativo	f	
_CFGKFF	feed forward	f	percentuale
_CFGKPS	coefficiente proporzionale asse slave	f	
_CFGKIS	coefficiente integrativo asse slave	f	
_CFGKDS	coefficiente derivativo asse slave	f	
_CFGQEAP	errore d'anello in positivo	d	quota
_CFGQEAN	errore d'anello in negativo	d	quota
_CFGKPI	coefficiente proporzionale d'interpolazione	f	
_CFGKII	coefficiente integrativo d'interpolazione	f	
_CFGKDI	coefficiente derivativo d'interpolazione	f	
_CFGTMINP	tensione minima positiva	f	volt
_CFGTMINN	tensione minima negativa	f	volt
_CFGSTMINP	tensione di soglia positiva	f	volt
_CFGSTMINN	tensione di soglia negativa	f	volt
_CFGESC	timeout movimento asse	i	msec
_CFGDSE	abilitazione servoerro dinamico	b	0=disabilitato, 1=abilitato
_CFGHEN	abilitazione adjust automatico	b	0=disabilitato, 1=abilitato
_CFGOFFSET	tensione di adjust - offset iniziale	f	volt
_CFGCEE	quota di collegamento encoder errato	d	quota
_CFGNOTCH	frequenza per filtro di notch	i	Hz

costante	descrizione	tipo	valore di ritorno
_CFGBUFI	dimensione buffer di calcolo integrativo	i	[1, 200]
_CFGQAP	finestra arrivo in quota positivo	d	
_CFGQAN	finestra arrivo in quota negativo	d	
_CFGSLOPEI	tipo di rampa per accelerazioni/decelerazioni in interpolazione	f	0=lineare, 1=a 'S' , 2= a doppia esse
_CFGKFFI	feed forward interpolazione	f	percentuale
_CFGAAF	attesa asse fermo	b	0=disabilitato, 1 =abilitato
_CFGENTYPE	tipo di encoder	i	0=simulato o assente, 1=reale
_SRPP	velocità inizio rampa passo passo	f	m/1' o inch/1" o gradi/1" o giri/1'
_ACC	tempo di accelerazione da 0 a VMAX	i	msec
_DEC	tempo di decelerazione da VMAX a 0	i	msec
_ACCI	tempo di accelerazione da 0 a VMAXI in interpolazione	i	msec
_DECI	tempo di decelerazione da VMAXI a 0 in interpolazione	i	msec
_QLP	limite asse in positivo	d	quota
_QLN	limite asse in negativo	d	quota
_KP	coefficiente proporzionale	f	
_KI	coefficiente integrativo	f	
_KD	coefficiente derivativo	f	
_KFF	feed forward	f	percentuale
_KPS	coefficiente proporzionale asse slave	f	
_KIS	coefficiente integrativo asse slave	f	
_KDS	coefficiente derivativo asse slave	f	
_QEAP	errore d'anello in positivo	d	quota
_QEAN	errore d'anello in negativo	d	quota
_VEL	velocità punto-punto	f	m/1' o inch/1" o gradi/1" o giri/1'
_VELI	velocità di interpolazione	f	m/1' o inch/1" o gradi/1" o giri/1'
_MODE	modo di funzionamento asse	b	1=normal, 2=free, 8=interpol., 10=coord.
_PHINV	inversione fasi encoder	b	0=no inversione, 1=inversione
_RFINV	inversione riferimento	b	0=no inversione, 1=inversione
_ZIND	abilitazione reset quota tacca di zero	b	0=disabilitato, 1=abilitato
_KPI	coefficiente proporzionale interpolazione	f	
_KII	coefficiente integrativo interpolazione	f	
_KDI	coefficiente derivativo interpolazione	f	
_KFFI	feed forward interpolazione	f	percentuale
_KFFA	feed forward di accelerazione	f	percentuale
_KFFAI	feed forward di accelerazione d'interpolazione	f	percentuale
_ESC	timeout movimento asse	i	msec

costante	descrizione	tipo	valore di ritorno
_CEE	quota di collegamento encoder errato	d	quota
_NOTCH	frequenza filtro di notch	i	Hz
_BUFI	dimensione buffer di calcolo integrativo	i	[1,200]
_QAP	finestra arrivo in quota positivo	d	
_QAN	finestra arrivo in quota negativo	d	
_QEAPINV	limite errore d'anello positivo in inversione	d	
_QEANINV	limite errore d'anello negativo in inversione	d	
_OFSCoord	quota offset movimento coordinato	d	
MS	tipo asse master o slave	b	0=non in chain,4=master,5=slave
QENC	quota encoder	d	quota
QR	quota reale	d	quota
_RIS	risoluzione utilizzata dall'asse	d	
_ST	stato asse	b	1=accel.,2=regime,3=decel.,4=quota,5=attesa finestra grande, 6=attesa asse fermo,7=attesa finestra piccola,8=partenza
QT	quota teorica	d	quota
EA	errore d'anello	d	quota
_FF	feed forward	i	
_VC	velocità corrente	f	
_P	correttore proporzionale	i	
_I	correttore integrativo	i	
_D	correttore derivativo	i	
_FLGS	flags assi	b	
_VCR	velocità corrente reale	f	
_ADJUST	compensazione offset asse	i	valore intero, che rappresenta la tensione, da passare all'azionamento, vista dal lato scheda assi del dac. Il fondoscala per l'azionamento è 10 volt e per il dac è di 32767
_DAC	DAC	i	valore intero, che rappresenta la tensione, da passare all'azionamento, vista dal lato scheda assi del dac. Il fondoscala per l'azionamento è 10 volt e per il dac è di 32767.
_ACCINST	valore di accelerazione istantanea	f	
_FFA	feed forward di accelerazione	i	
_GONETIME	tempo trascorso dall'inizio del movimento	f	sec (0 per assi slave e assi passo-passo)
_RESTIME	tempo residuo alla fine del movimento. I valori sono relativi al movimento allocato nel buffer al momento della richiesta	f	sec (0 per assi slave, assi in movimento coordinato, assi passo-passo e movimento ISO)
_TARGETTIME	tempo impiegato per generare la quota target	f	microsecondi
_GONESPACE	spazio percorso dall'inizio del movimento. I valori sono relativi al movimento allocato nel buffer al momento della richiesta	f	percentuale (100 per assi slave,assi in movimento coordinato e assi passo-passo)

costante	descrizione	tipo	valore di ritorno
_RESSPACE	spazio residuo alla fine del movimento. I valori sono relativi al movimento allocato nel buffer al momento della richiesta	f	percentuale (0 per assi slave assi in movimento coordinato e assi passo-passo)
_AXESJERK	abilitazione sull'asse del controllo del jerk	b	1=controllo abilitato , 0=controllo non abilitato
_MOVEJERK	abilitazione del controllo del jerk sul movimento in cui è impegnato l'asse	b	1=controllo abilitato, 0=controllo non abilitato
_MOVETYPE	tipo di movimento in cui è impegnato l'asse	b	1=movimento interpolato classico, 2= movimento interpolato multiasse , 3=movimento coordinato, 4= movimento punto punto, 5=movimento in chain (solo assi slave)
_PARTYPESET	tipo di parametri asse in uso durante il movimento	i	1=interpolazione, 0=punto punto
_AXINRIFLOC	asse in presa in un sistema di riferimento locale	i	1=sì, 0=no
_QTARGETTOOL	quota target dell'asse. Se interpolazione ISO quota target della coordinata della punta utensile corrispondente all'asse	d	
_QREALTOOL	quota reale dell'asse. Se interpolazione ISO quota reale della coordinata della punta utensile corrispondente all'asse	d	
_BACKLASH	valore del gioco meccanico definito per l'asse	d	
DISABLED	disabilitazione di un asse	b	1=asse disabilitato,0=asse abilitato
_DYNLIMIT	abilitazione del controllo dinamico dei limiti asse	b	1=controllo abilitato, 0=controllo non abilitato
_AXESFEED	valore di feedrate override applicato correntemente all'asse	f	
_CORRLIN	tipo di correttore di linearità in uso	i	0=nessun correttore in uso, 1=autocorrettore, 2=correttore incrociato, 3=autocorrettore insieme a correttore incrociato
_VELISO	velocità della punta dell'utensile durante il movimento 5 assi	f	
_ISOSTOPS	numero di arresti forzati del movimento interpolato dovuti a situazioni limite nella gestione del lookahead	i	
_CURRATIO	valore del rapporto di concatenamento correntemente in uso	d	
_DYNRATIO	ritorna, se è in atto un cambio dinamico di rapporto di concatenamento	i	0=no, 1 = sì
_RESBLOCK	numero di blocchi di spostamento ancora da eseguire	i	
_EXECBLOCK	numero di blocchi di spostamento eseguiti	i	
_TOTALBLOCK	numero di blocchi di spostamento accodati in totale nel movimento (valore attuale)	i	

SETACCLIMIT

Sintassi

SETACCLIMIT **asse,[valore]**

Argomenti

asse nome del dispositivo tipo asse
valore costante di tempo dell'azionamento

Descrizione

Abilita e disabilita il calcolo automatico della velocità di regime di interpolazione in funzione delle accelerazioni tollerate dagli assi. Il parametro **valore** è una costante di tempo utilizzata per determinare il gradino di velocità tollerato dall'**asse**, in millisecondi. Questo parametro è opzionale. Se il parametro è omesso, l'istruzione avrà l'effetto di disabilitare il calcolo automatico. Un valore standard per questo parametro è 30 millisecondi. Abbassando questo tempo si avrà l'effetto di rallentare il profilo, rendendo più blando il movimento. Aumentandolo si otterrà l'effetto opposto. Questa istruzione non può essere applicata alle interpolazioni elicoidali.

SETACCSTRATEGY

Sintassi

SETACCSTRATEGY **asse, [valore]**

Argomenti

asse nome del dispositivo tipo asse
[valore] variabile o costante integer

Descrizione

Consente di selezionare il tipo di accelerazione che si desidera per i successivi movimenti di interpolazione. L'istruzione va eseguita per tutti gli assi impegnati nell'interpolazione. I valori ammessi per il parametro **valore** sono 0, 1 e 2. Nel caso venga passato il valore 0, si adotterà la consueta strategia di accelerazione (verrà scelta come accelerazione di profilo quella minore tra tutti gli assi impegnati nell'interpolazione). In caso di valore uguale a 2 e interpolazione lineare, verrà presa l'accelerazione massima che i singoli assi possono sopportare (considerando le singole componenti di tutti gli assi, lineari e/o rotanti), la gestione delle accelerazioni in caso di interpolazione circolare rimane invariata. Il caso di valore 1 richiama una gestione obsoleta che viene mantenuta per compatibilità.

SETAXPARTYPE

Sintassi

SETAXPARTYPE **asse, [valore]**

Argomenti

asse nome del dispositivo tipo asse
[valore] variabile o costante integer

Descrizione

Quando si sta eseguendo una interpolazione multilineare, questa istruzione permette di cambiare il set di parametri asse in uso, passando da quelli tipici dell'interpolazione (**valore** = 1) a quelli usati per il movimento punto punto (**valore** = 0). Se la variabile **valore** è omessa, i parametri usati sono quelli di interpolazione. Il cambio di set di parametri può essere fatto solo con asse fermo in stato QUOTA, altrimenti l'istruzione genera l'errore di sistema 4101- Gestione incongruente dell'asse NomeAsse.

SETCONTORNATURE

Sintassi

SETCONTORNATURE **[valore1[,valore2]]**

Argomenti

valore1 costante o variabile. Massimo angolo di contornatura
valore2 costante o variabile. Massimo angolo di rallentamento

Descrizione

Imposta l'angolo minimo tra le tangenti di due traiettorie eseguite in interpolazione, oltre il quale la macchina non eseguirà una contornatura, ovvero gli assi verranno fermati al termine della prima traiettoria e, quindi, fatti ripartire lungo la seconda. Per questo si definisce un *angolo massimo di contornatura* **valore1**, che rappresenta l'angolo massimo tra due tratti di spostamento, sotto il quale il movimento non viene arrestato.


```

setFeedCoord      X, 20, 80
setFeedCoord      Y, 10, 1
setFeedCoord      Z, 3, 3

coordin           matrix, deltaT, UP, rigaInit, rigaEnd,mask,_
                  X,columnX, Y,columnY, Z,columnZ

waitstill         X,y,z

```

fret

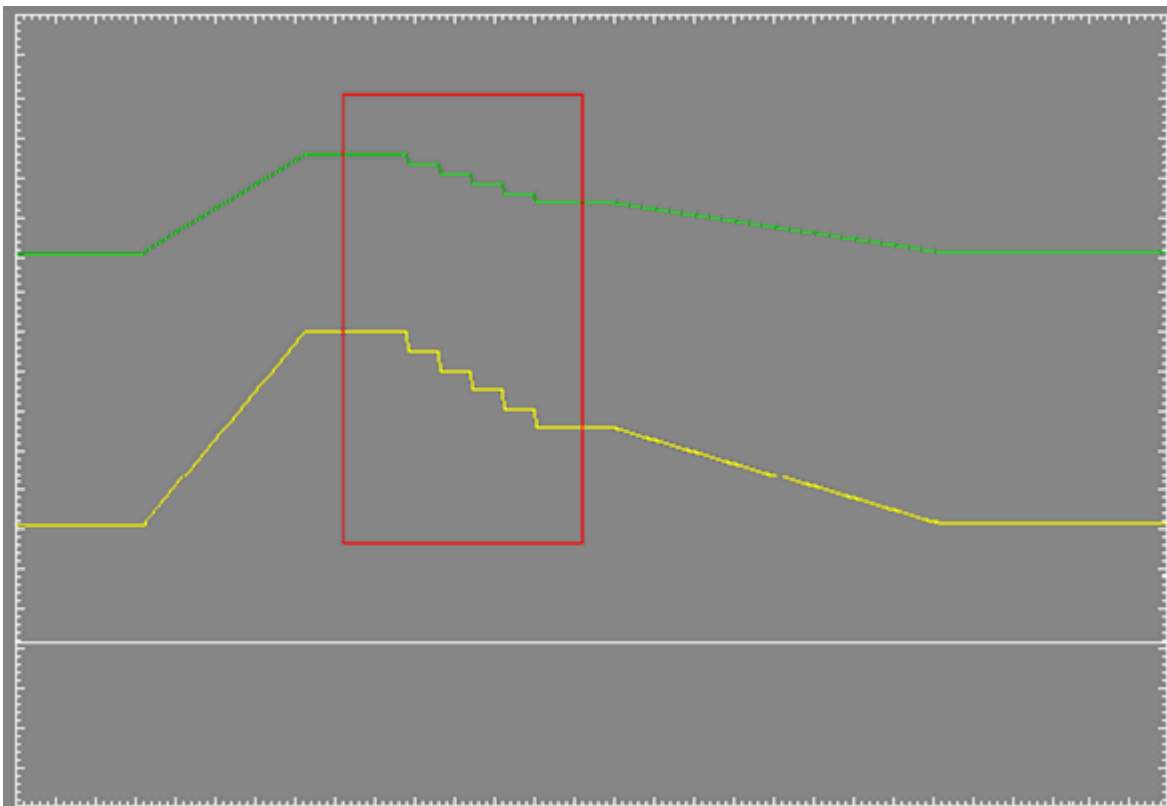
Supponiamo che in un determinato passaggio del movimento coordinato, l'asse Z non si muova. I parametri impostati risultano essere

```

Variazione_Max      = 10
Delta_T             = 10 / 0.25 = 40

```

Abbiamo quindi la seguente tracciata di oscilloscopio con in verde il profilo di velocità dell'asse X e in giallo quello dell'asse Y.



SETOFFSET

Sintassi

```
SETOFFSET      asse, quota
```

Argomenti

```

asse          nome del dispositivo tipo asse
quota        costante o variabile. Offset per movimento coordinato

```

Descrizione

Permette di applicare un offset alle quote di un movimento coordinato. L'offset specificato dal parametro **quota** verrà utilizzato nei movimenti coordinati successivi, sommando la quota indicata a tutte le quote presenti in tabella. Vedere anche l'istruzione [COORDIN](#).

DYNLIMIT deve essere impostata una istruzione [SETLIMPOS](#) e una istruzione [SETLIMNEG](#) per definire i nuovi limiti assi, altrimenti i valori limiti considerati, diminuiti dello spazio di arresto minimo, sono uguali alla quota 0.0 per entrambe le direzioni di moto.

Esempio

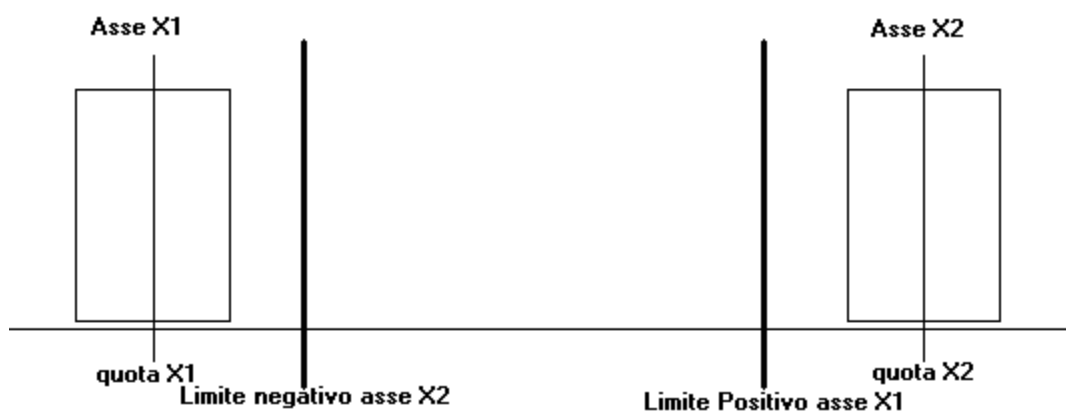
Verifica dei limiti assi secondo le due tipologie di test statico e dinamico, con assi sulla stessa direttrice di moto.

Test statico

In un generico movimento l'**Asse X1** non può essere superiore al limite positivo iniziale dato dalla quota dell'**Asse X2**. La verifica dei limiti assi genera un errore di sistema n.4108 "Asse X1: quota finale oltre il limite software".

Test dinamico

In un generico movimento si verifica che la quota istantanea **quota X1** stia entro i limiti asse diminuiti, con il segno opportuno secondo la direzione di moto dell'asse, dello spazio minimo di arresto dell'asse stesso. Lo spazio minimo di arresto viene calcolato in funzione della velocità istantanea e della decelerazione impostata in configurazione per il movimento punto-punto. Inoltre, viene disabilitato il controllo iniziale del superamento dei limiti imposti da parte delle quote passate tramite le istruzioni di movimento.



ENABLESTARTCONTROL

Sintassi

ENABLESTARTCONTROL **asse, [timeout]**

Argomenti

asse nome del dispositivo tipo asse
timeout variabile o costante integer. È il tempo limite di attesa espresso in real-time.

Descrizione

Questa istruzione permette di abilitare e impostare il **timeout** per il controllo della mancata partenza o improvviso stop dell'asse.

Se l'asse non si muove di almeno 2 passi in 200 real-time a fronte di una richiesta di movimento, viene generato l'errore di sistema n. 3 "Servoerror".

Se il parametro **timeout** è impostato a zero viene disabilitato il controllo. L'istruzione non ha effetto se la velocità teorica è minore di due passi in 200 real-time o se il movimento si conclude in meno di 200 real-time.

Esempio

```
; timeout di partenza assi pari a 10 real-time  
ENABLESTARTCONTROL    X, 10
```

NOTCHFILTER

Sintassi

NOTCHFILTER **asse, [valore]**

Argomenti

asse nome del dispositivo tipo asse
valore costante o variabile. Valore di frequenza [Hz]. Ammessi valori compresi tra **0** e **500**.

Descrizione

Imposta la frequenza di taglio del filtro notch per l'asse specificato. Se **valore** è uguale a 0 il filtro viene disabilitato. Nel caso in cui il parametro **valore** è omissso verrà usato il valore impostato in configurazione.

Esempio

```
; taglio della frequenza 97 Hz
NOTCHFILTER X, 97
```

RESLIMNEG**Sintassi**

RESLIMNEG **asse**

Argomenti

asse nome del dispositivo tipo asse

Descrizione

Disabilita il test su limite negativo dell'**asse** specificato. Queste istruzioni sono utilizzate di solito nelle routine di azzeramento per la ricerca degli switch di azzeramento, permettendo così agli assi di superare i valori di configurazione impostati. Vedere anche le istruzioni [SETLIMNEG](#), [SETLIMPOS](#), [RESLIMPOS](#).

Esempio

[Routine di Azzeramento di un asse](#)

RESLIMPOS**Sintassi**

RESLIMPOS **asse**

Argomenti

asse nome del dispositivo tipo asse

Descrizione

Disabilita il test su limite positivo dell'**asse** specificato. Queste istruzioni sono utilizzate di solito nelle routine di azzeramento per la ricerca degli switch di azzeramento, permettendo così agli assi di superare i valori di configurazione impostati. Vedere anche le istruzioni [RESLIMNEG](#), [SETLIMPOS](#), [SETLIMNEG](#).

Esempio

[Routine di Azzeramento di un asse](#)

SETADJUST**Sintassi**

SETADJUST **asse, stato, [valore]**

Argomenti

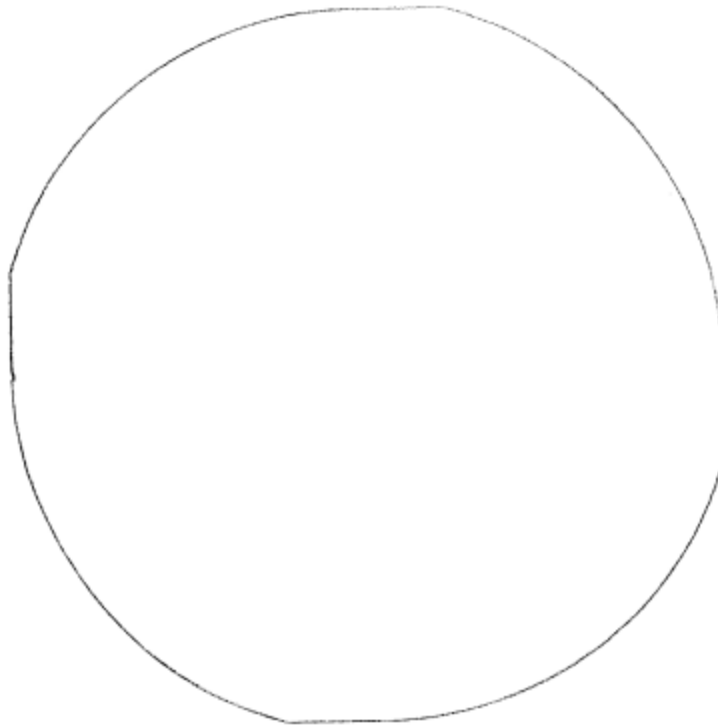
asse nome del dispositivo tipo asse
stato costante predefinita. Può assumere i seguenti valori:
ON attiva
OFF disattiva
[valore] variabile o costante float. Tensione [Volt]

Descrizione

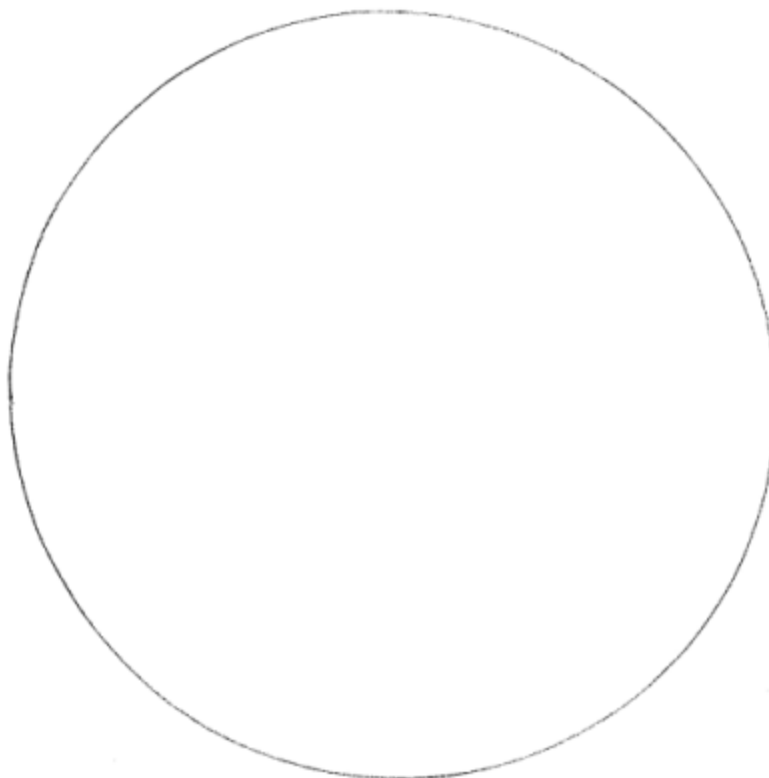
Attiva o disattiva, sull'**asse** indicato, il calcolo di recupero offset automatico, cioè l'adjust. L'adjust permette di compensare dei leggeri offset di posizione a fine movimento dell'asse. Normalmente l'adjust è abilitato.

Può essere conveniente disabilitare l'adjust per assi movimentati da motori che presentano un'elevata isteresi di posizione che quindi non otterrebbero alcun beneficio dall'utilizzo di questa funzionalità del controllo.

Quando viene riattivato l'adjust dopo una disabilitazione, il controllo non tiene conto del valore calcolato in precedenza. L'istruzione può quindi essere utilizzata per azzerare l'adjust accumulato da un asse senza dover reinizializzare il controllo.



La seconda figura rappresenta la stessa interpolazione, ma con l'utilizzo dell'istruzione di recupero giochi.



SETBIGWINFACTOR

Sintassi

SETBIGWINFACTOR

asse, valore

Argomenti

asse nome del dispositivo tipo asse
valore variabile o costante double. Fattore di moltiplicazione per il calcolo della finestra grande.

Descrizione

Questa istruzione permette di modificare il fattore di moltiplicazione per il calcolo della finestra grande sull'**asse** scelto. Per il calcolo della finestra grande si moltiplica la variabile **valore** per il parametro definito in configurazione degli assi di Finestra di arrivo in quota. Il **valore** che può essere impostato deve essere compreso tra 1 e 257 estremi esclusi. Il default è 4.0.

SETDEADBAND**Sintassi**

SETDEADBAND **asse, VMinPos, VMinNeg, VSogliaPos, VSogliaNeg**

Argomenti

asse nome di dispositivo tipo asse
VMinPos variabile o costante float. Tensione minima positiva [Volt]
VMinNeg variabile o costante float. Tensione minima negativa [Volt]
VSogliaPos variabile o costante float. Soglia positiva [Volt]
VSogliaNeg variabile o costante float. Soglia negativa [Volt]

Descrizione

Imposta i parametri di tensione minima per l'asse indicato. I valori di tensione minima (positiva/negativa) vengono sommati alla tensione di riferimento teorica (positiva/negativa) se il valore di questa supera il valore di soglia (positiva/negativa) impostato. Se la tensione di riferimento teorica è all'interno dei valori di soglia, la tensione di riferimento effettiva viene forzata a zero. È possibile disabilitare la gestione della tensione minima impostando tutti i valori a zero. I valori di soglia devono essere sempre minori o uguali ai rispettivi valori di tensione minima.

All'inizializzazione del sistema la gestione della tensione minima è disabilitata.

SETENCLIMIT**Sintassi**

SETENCLIMIT **asse [, valore]**

Argomenti

asse nome del dispositivo tipo asse
valore costante o variabile double

Descrizione

Modifica il limite di collegamento encoder errato. Il parametro è espresso nell'unità di misura dell'asse. I valori ammessi devono essere compresi in un intervallo equivalente a 128 - 16384 passi encoder. Se il parametro è omesso viene ripristinato il valore di default pari a 1024 passi.

Ad esempio, per un asse con risoluzione pari a 1000 impulsi/mm i valori ammessi saranno compresi tra 0,128 e 16,384 mm.

Se il parametro **valore** è impostato a zero viene disabilitato il controllo del limite collegamento encoder errato.

Esempio

; imposto un limite di collegamento encoder pari a 3.5

SETENCLIMIT X, 3.5

SETINDEXEN**Sintassi**

SETINDEXEN **asse, stato**

Argomenti

asse nome di dispositivo tipo asse
stato costante predefinita. I valori ammessi sono:
ON stato tacca di zero attivo
OFF stato tacca di zero disattivo

Descrizione

Attiva o disattiva sull'**asse** indicato l'azzeramento della quota in corrispondenza della tacca di zero. Per poter eseguire questa istruzione l'asse deve essere di tipo conteggio.

SETINTEGTIME

Sintassi

SETINTEGTIME **asse [, valore]**

Argomenti

asse nome del dispositivo tipo asse
valore costante integer o variabile

Descrizione

Imposta il numero di campioni dell'errore d'anello utilizzati per il calcolo della componente integrale. I valori validi sono tra 1 e 200. La variazione di tale parametro al volo è possibile ma può causare dei gradini sul riferimento di velocità dell'asse. Si consiglia quindi di agire su tale parametro ad assi fermi e disabilitati o preferibilmente in free.

SETIRMPP

Sintassi

SETIRMPP **asse, velocità**

Argomenti

asse nome del dispositivo tipo asse
velocità costante float o variabile float. Velocità di inizio rampa

Descrizione

Assegna all'**asse** il valore di **velocità di inizio rampa**. È la velocità minima del motore passo-passo. Questa istruzione è utilizzata per assi movimentati con motori passo-passo.

SETLIMNEG

Sintassi

SETLIMNEG **asse [, quota]**

Argomenti

asse nome del dispositivo tipo asse
quota costante o variabile. Limite negativo

Descrizione

Assegna all'**asse** la **quota** limite negativo.
Se il parametro **quota** è omissso, viene abilitato il limite negativo di configurazione.
Queste istruzioni sono utilizzate di solito nelle routine di azzeramento per la ricerca degli switch di azzeramento, permettendo così agli assi di superare i valori di configurazione impostati.
Vedere anche le istruzioni [RESLIMNEG](#), [SETLIMPOS](#), [RESLIMPOS](#).

Esempio

[Routine di Azzeramento di un asse](#)

SETLIMPOS

Sintassi

SETLIMPOS **asse [, quota]**

Argomenti

asse nome del dispositivo tipo asse
quota costante o variabile. Limite positivo

Descrizione

Assegna all'**asse** la **quota** limite positivo.
Se il parametro **quota** è omissso, viene abilitato il limite positivo di configurazione.
Queste istruzioni sono utilizzate di solito nelle routine di azzeramento per la ricerca degli switch di azzeramento, permettendo così agli assi di superare i valori di configurazione impostati.
Vedere anche le istruzioni [RESLIMNEG](#), [RESLIMPOS](#), [SETLIMNEG](#).

Esempio

[Routine di Azzeramento di un asse](#)

SETMAXER

Sintassi

SETMAXER **asse, valore [, direzione]**

Argomenti

asse nome del dispositivo tipo asse
valore costante o variabile. Massimo errore d'inseguimento
direzione costante predefinita. Direzione asse. Può assumere i seguenti valori:
POSITIVE, NEGATIVE

Descrizione

Assegna all'**asse**:

- il **valore** massimo di inseguimento tollerato nel caso di test statico del servoerror
- il **valore**, che sommato all'errore teorico proporzionale alla velocità, determina il valore massimo di inseguimento tollerato nel caso di test dinamico del servoerror.

Se la **direzione** è omessa, il valore massimo di inseguimento viene imposto per entrambe le direzioni.

SETMAXERNEG

Sintassi

SETMAXERNEG **asse, ritardo, anticipo**

Argomenti

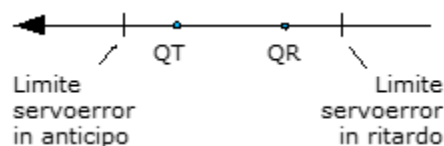
asse nome del dispositivo tipo asse
ritardo costante o variabile. Massimo errore d'inseguimento
anticipo costante o variabile. Massimo errore di anticipo

Descrizione

Assegna all'**asse** i valori massimi di errore di inseguimento (**ritardo**) e di anticipo (**anticipo**) ammessi dal controllo, per la sola direzione negativa, prima di generare la segnalazione di "servoerror".

- **ritardo**: è il valore massimo di inseguimento tollerato nel caso di test statico del servoerror oppure nel caso di test dinamico del servoerror è il valore che sommato all'errore teorico proporzionale alla velocità determina il valore massimo di inseguimento tollerato.
- **anticipo**: è il valore massimo di inseguimento tollerato durante l'inversione di movimento da movimento negativo a movimento positivo

L'errore di inseguimento è la differenza tra la quota teorica (in cui l'asse dovrebbe trovarsi) e la quota reale. Quando l'asse si muove in direzione negativa, un errore d'inseguimento con segno negativo indica una condizione di ritardo dell'asse, un errore d'inseguimento con segno positivo indica una condizione di anticipo. Se non si utilizza questa istruzione, il controllo utilizza i valori massimi di errore di inseguimento presenti in configurazione asse, in tal caso il limite di anticipo è pari ad 1/4 di quello di ritardo.



Esempio

SETMAXERNEG Assi.X, 10, 5

;Il ritardo massimo dell'asse è di 10mm, l'anticipo massimo di 5mm

SETMAXERPOS

Sintassi

SETMAXERPOS **asse, ritardo, anticipo**

Argomenti

asse nome del dispositivo tipo asse
ritardo costante o variabile. Massimo errore d'inseguimento
anticipo costante o variabile. Massimo errore di anticipo

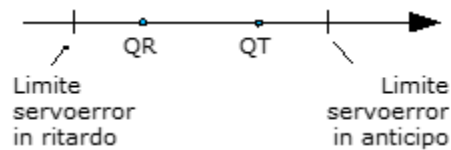
Descrizione

Assegna all'**asse** valori massimi di errore di inseguimento (**ritardo**) e di anticipo (**anticipo**) ammessi dal controllo, per la sola direzione positiva, prima di generare la segnalazione di "servoerror".

- **ritardo**: è il valore massimo di inseguimento tollerato nel caso di test statico del servoerror, oppure nel caso di test dinamico del servoerror, è il valore che, sommato all'errore teorico proporzionale alla velocità, determina il valore massimo di inseguimento tollerato.
- **anticipo**: è il valore massimo di inseguimento tollerato durante l'inversione di movimento da movimento negativo a movimento positivo.

L'errore di inseguimento è la differenza tra la quota teorica (in cui l'asse dovrebbe trovarsi) e la quota reale. Quando l'asse si muove in direzione positiva, un errore d'inseguimento con segno positivo indica una condizione di ritardo dell'asse, mentre un errore d'inseguimento con segno negativo indica una condizione di anticipo.

Se non si utilizza questa istruzione, il controllo utilizza i valori massimi di errore di inseguimento presenti in configurazione asse, in tal caso il limite di anticipo è pari ad 1/4 di quello di ritardo.

**Esempio**

`SETMAXERPOS` Assi.X, 10, 5

;Il ritardo massimo dell'asse è di 10mm, l'anticipo massimo di 5mm

SETMAXERTYPE**Sintassi**

`SETMAXERTYPE` **asse, tipo**

Argomenti

asse nome del dispositivo tipo asse
tipo Costante tipo integer. I valori ammessi sono:
 0 = imposta servoerror a soglia (valore di default)
 1 = imposta servoerror dinamico

Descrizione

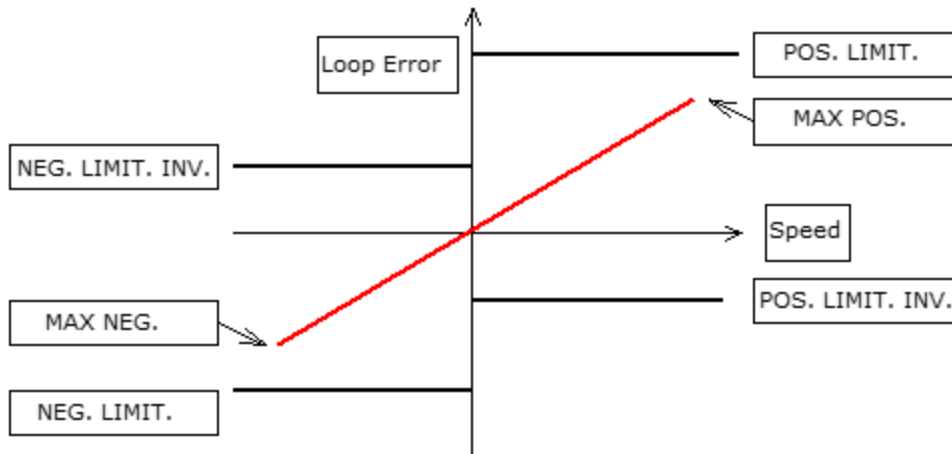
Questa istruzione permette di impostare il **tipo** di test sul servoerror. La gestione tradizionale del servoerror prevede una coppia di limiti (positivo e negativo) costanti al variare della velocità dell'asse. Questo tipo di gestione porta a dimensionare i limiti in funzione della velocità massima dell'asse, ovvero si imposta un limite tale da non far scattare l'errore in condizioni di normale funzionamento. Tuttavia, alle basse velocità l'errore d'anello ha valori generalmente molto più bassi rispetto al limite impostato, questo comporta un ritardo nell'identificazione di una condizione di errore.

La gestione del servoerror a finestra si basa sul calcolo dell'errore di anello teorico. I limiti di servoerror positivo e negativo vengono calcolati in funzione di questo, sommando e sottraendo loro un valore di soglia. Se l'errore d'anello reale esce da questa soglia, si genera l'errore di servoerror.

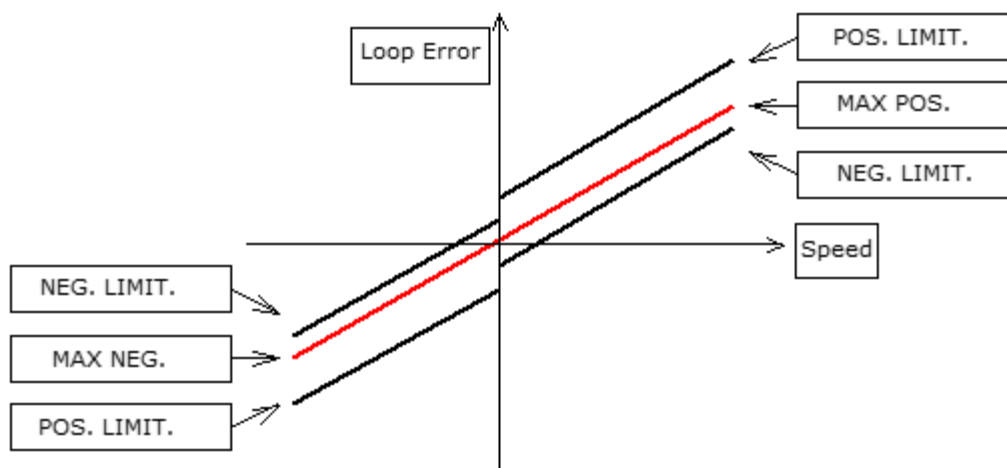
Nota

Se si imposta il test sul servoerror dinamico, è generalmente necessario modificare i valori di Limite servoerror positivo e Limite servoerror negativo impostati in Configurazione assi per il servoerror a soglia. Questo perché i suddetti valori vengono utilizzati come valori iniziali per il calcolo dell'errore d'anello.

Limite_ServoError a soglia



Limite ServoError dinamico



SETPHASESINV

Sintassi

SETPHASESINV **asse, stato**

Argomenti

asse nome di dispositivo tipo asse
stato costante predefinita. I valori ammessi sono:
ON stato inversione fasi attivo
OFF stato inversione fasi disattivo

Descrizione

Attiva o disattiva sull'**asse** indicato l'inversione fasi. Permette di compensare via software un'eventuale inversione di cablaggio delle fasi encoder. Se utilizzato assieme all'inversione riferimento, permette di invertire il verso dell'asse (se il cablaggio è corretto).
 Per poter eseguire questa istruzione, l'asse deve essere in stato FREE.

SETREFINV

Sintassi

SETREFINV **asse, stato**

Argomenti

asse nome di dispositivo tipo asse
stato costante predefinita. I valori ammessi sono:
ON abilita l'inversione di riferimento di velocità
OFF disabilita l'inversione di riferimento di velocità

Descrizione

Attiva o disattiva sull'asse indicato l'inversione del riferimento di velocità. Se utilizzato assieme all'inversione fasi, permette di invertire il verso dell'asse (se il cablaggio è corretto).
Per poter eseguire questa istruzione l'asse deve essere in stato FREE.
Vedere anche [SETPHASESINV](#).

SETRESOLUTION**Sintassi**

SETRESOLUTION **asse [, valore]**

Argomenti

asse nome di dispositivo tipo asse
valore costante o variabile double

Descrizione

Cambia la risoluzione dell'asse specificato. Se **valore** è omissso, viene utilizzato il valore di risoluzione impostato in configurazione. La modifica del valore di risoluzione può essere eseguita solo con asse fermo (stato asse = quota), altrimenti viene generato l'errore di sistema n. 4101 "Gestione incongruente dell'asse".

10.3.5 Contatori**DECOUNTER****Sintassi**

DECOUNTER **nomecounter [, valore]**

Argomenti

nomecounter nome di dispositivo tipo contatore
valore costante o variabile o dispositivo tipo contatore

Descrizione

Decrementa il contatore **nomecounter** del **valore** specificato. Se **valore** è omissso, viene assunto il valore 1. Vedere anche le istruzioni [SETCOUNTER](#) e [INCOUNTER](#).

INCOUNTER**Sintassi**

INCOUNTER **nomecounter [, valore]**

Argomenti

nomecounter nome di dispositivo tipo contatore
valore costante o variabile o dispositivo tipo contatore

Descrizione

Incrementa il contatore **nomecounter** del **valore** specificato. Se **valore** è omissso, viene assunto il valore 1. Vedere anche le istruzioni [SETCOUNTER](#) e [DECOUNTER](#).

SETCOUNTER**Sintassi**

SETCOUNTER **nomecounter, valore**

Argomenti

nomecounter nome di dispositivo tipo contatore
valore costante o variabile o dispositivo tipo contatore

Descrizione

Imposta il contatore **nomecounter** al **valore** specificato.
Vedere anche [INCOUNTER](#) e [DECOUNTER](#).

10.3.6 Timer**HOLDTIMER****Sintassi**

HOLDTIMER **nometimer**

Argomenti
nometimer nome di dispositivo tipo timer

Descrizione
 Blocca l'aggiornamento del timer **nometimer**.
 Vedere anche [STARTTIMER](#) e [SETTIMER](#)

SETTIMER

Sintassi
SETTIMER **nometimer, tempo**

Argomenti
nometimer nome di dispositivo tipo timer
tempo costante o variabile o dispositivo tipo timer

Descrizione
 Imposta **nometimer** al **tempo** specificato (in secondi).
 Sono ammessi solo valori maggiori di zero. La precisione massima dei timer è di 4 ms.
 Vedere anche [STARTTIMER](#) e [HOLDTIMER](#).

Esempio

```

;La Function imposta un timer
;Imposto il timer Timeout
;al valore 20 secondi
SETTIMER Timeout,20
; parte il timer in modalità decrescente. Quando arriva a 0 si ferma

STARTTIMER Timeout,DOWN

```

STARTTIMER

Sintassi
STARTTIMER **nometimer [, direzione]**

Argomenti
nometimer nome di dispositivo tipo timer
direzione costante predefinita. I valori impostabili sono:
UP crescente
DOWN decrescente

Descrizione
 Fa partire il timer **nometimer** con modalità eventualmente specificata da **direzione**.
 Se il parametro **direzione** è omissso, viene assunta la modalità **DOWN**.
 Quando un timer (avviato in modalità decrescente) arriva a zero, si ferma automaticamente.
 Vedere anche [HOLDTIMER](#) e [SETTIMER](#).

10.3.7 Variabili, Vettori e Matrici

CLEAR

Sintassi
CLEAR **nomevar o vettore o matrice[rigamatrice]**

Argomenti
nomevar nome variabile
vettore nome vettore
matrice nome matrice
rigamatrice costante o variabile o contatore. Riga della matrice

Descrizione
 Inizializza al valore 0 lo spazio di memoria riservato a variabili (**nomevar**), vettori (**vettore**), matrici (**matrice**) o agli elementi di una riga di matrice.

FIND

Sintassi

FIND **matrice, colonna, limite_min, limite_max, valore, variabile**
FIND **vettore, limite_min, limite_max, valore, variabile**

Argomenti

matrice nome della matrice. Matrice in cui eseguire la ricerca
vettore nome del vettore. Vettore in cui eseguire la ricerca
colonna costante o variabile intera o nomecounter. Numero di colonna della matrice in cui eseguire la ricerca
limite_min costante o variabile. Indice minimo del vettore o della matrice da cui iniziare la ricerca
limite_max costante o variabile. Indice massimo del vettore o della matrice dove terminare la ricerca
valore costante o variabile. Valore da ricercare
variabile variabile. Risultato della ricerca

Descrizione

Esegue una ricerca sequenziale di un valore all'interno di un **vettore** o di una **colonna** della **matrice** e pone l'indice dell'elemento in **variabile**.
 Se il valore non è stato trovato, **variabile** conterrà il valore -1.

FINDB

Sintassi

FINDB **matrice, colonna, limite_min, limite_max, valore, variabile**
FINDB **vettore, limite_min, limite_max, valore, variabile**

Argomenti

matrice nome della matrice. Matrice in cui eseguire la ricerca
vettore nome del vettore. Vettore in cui eseguire la ricerca
colonna costante o variabile intera o nomecounter. Numero di colonna della matrice in cui eseguire la ricerca
limite_min costante o variabile. Indice minimo del vettore o della matrice da cui iniziare la ricerca
limite_max costante o variabile. Indice massimo del vettore o della matrice dove terminare la ricerca
valore costante o variabile. Valore da ricercare
variabile variabile. Risultato della ricerca

Descrizione

Esegue una ricerca veloce di un valore all'interno di un **vettore** o di una **colonna** della **matrice** e pone l'indice dell'elemento in **variabile**. Perché la ricerca abbia successo, è necessario che il **vettore** o la **colonna** della **matrice** siano stati ordinati precedentemente con l'istruzione SORT secondo un ordinamento crescente.
 Se il valore non è stato trovato, **variabile** conterrà il valore -1.

LASTELEM

Sintassi

LASTELEM **vettore, elementivet**
LASTELEM **matrice, righemat**

Argomenti

matrice nome di matrice
vettore nome di vettore
elementivet variabile. Numero di elementi del vettore
righemat variabile. Numero di righe della matrice

Descrizione

Scriva nella variabile **elementivet** il numero di elementi del **vettore** o nella variabile **righemat** il numero di righe della **matrice**.

LOCAL

Sintassi

LOCAL **nomevar AS tipo**
LOCAL **vettore[n° elementi] AS tipo**
LOCAL **matrice[n° righe] AS tipo, tipo, tipo, etc.**

LOCAL **matrice[n° righe] AS tipo:nomecol1, tipo:nomecol2, tipo:nomecol3, etc.**

Argomenti

nomevar nome variabile
[n° elementi] variabile o costante (argomento obbligatorio). Numero di elementi del vettore
[n° righe] costante o variabile (argomento obbligatorio). Numero di righe della matrice
tipo char, integer (32 bit), float (32 bit), double (64 bit), string, timer
nomecol1...nomecolN nome colonna. Etichetta

Descrizione

Dichiarazione di una variabile locale. Prima di questa istruzione può comparire solo l'istruzione PARAM che definisce i parametri della funzione.
 Per maggiori informazioni sulle variabili locali vedere [Le variabili locali](#).

MOVEMAT**Sintassi**

MOVEMAT **nomematsorg, nomematdest**

MOVEMAT **nomematsorg[rigasorg], nomematdest[rigadest]**

MOVEMAT **nomematsorg[rigasorg], nomematdest[rigadest], numrighe**

Argomenti

nomematsorg nome matrice sorgente
rigasorg numero di riga di partenza per la copia della matrice sorgente (argomento obbligatorio)
nomematdest nome matrice destinataria
rigadest numero di riga di partenza per la copia nella matrice destinataria (argomento obbligatorio)
numrighe numero di righe da copiare

Descrizione

Copia il contenuto di una intera matrice **nomematsorg** in un'altra matrice **nomematdest** o una o più righe **numrighe** della riga di matrice **nomematsorg[rigasorg]** nella riga della matrice **nomematdest[rigadest]**.
 Se il parametro **numrighe** non è specificato, viene copiata una sola riga. Le due matrici devono avere la stessa tipologia di struttura (numero di colonne e tipo di dato di ciascuna colonna) e, nel caso di matrici intere, anche lo stesso numero di righe. Si possono spostare righe di dati all'interno della stessa matrice.

Esempio

```
Movemat Mx1, Mx2 ; copia la matrice Mx1 nella Mx2

; copia la riga 10 della matrice Mx1 nella riga 3 di Mx2
Movemat Mx1[10], Mx2[3]

; copia la riga 1 della matrice Mx1 nella riga 7 di Mx1
Movemat Mx1[1], Mx1[7]
; copia 6 righe a partire dalla riga 2 della matrice Mx1 nella matrice
; Mx2 a partire dalla riga 8
Movemat Mx1[2], Mx2[8],6

; copia 4 righe a partire dalla riga 2
; della matrice Mx1 nella stessa matrice
; Mx1 a partire dalla riga 10
Movemat Mx1[2], Mx1[10],4
```

PARAM**Sintassi**

[PARAM] **nomevar AS tipo**
[PARAM] **vettore[n° elementi] AS tipo**
[PARAM] **matrice[n° righe] AS tipo, tipo, tipo, etc.**
[PARAM] **matrice[n° righe] AS tipo:alias, tipo:alias, tipo:alias, etc.**

Argomenti

nomevar	nome variabile
[n° elementi]	costante (argomento obbligatorio)
[n° righe]	costante (argomento obbligatorio)
tipo	char, integer (32 bit), float (32 bit), double (64 bit), string

Descrizione

I parametri si comportano come le variabili locali (vedi [LOCAL](#)), ma vengono inizializzati da chi chiama la funzione. La sintassi per la dichiarazione dei parametri è la stessa usata per le variabili locali. I parametri vengono passati per valore o per riferimento in funzione del loro tipo. Vedere "[Le funzioni](#)".
Vanno dichiarati prima di ogni altra istruzione.
Per maggiori informazioni vedere [Le variabili locali](#).

SETVAL**Sintassi**

SETVAL **valore, nomevar**

Argomenti

valore costante o variabile o nome device
nomevar variabile o nome device

Descrizione

Assegna il **valore** specificato alla variabile **nomevar** o all'i-esimo elemento di un vettore o di una matrice.

SORT**Sintassi**

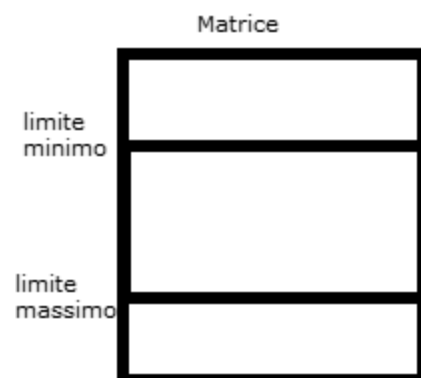
SORT **matrice, colonna [, ordine], limite_min, limite_max**
SORT **vettore [,ordine], limite_min, limite_max**

Argomenti

matrice	nome della matrice
vettore	nome del vettore
colonna	costante o variabile. Numero di colonna della matrice
ordine	costante predefinita. Indica la modalità di ordinamento I valori ammessi sono: UP ordinamento crescente DOWN ordinamento decrescente
limite_min	costante o variabile. Indice minimo del vettore o della matrice da cui iniziare l'ordinamento
limite_max	costante o variabile. Indice massimo del vettore o della matrice dove terminare l'ordinamento

Descrizione

Effettua un ordinamento dei valori presenti all'interno di un **vettore** o di una **matrice**, con senso definito dalla costante **ordine**.
Nel caso di una matrice l'ordinamento delle righe è dettato dalla disposizione crescente (UP) o decrescente (DOWN) dei valori contenuti nella **colonna** selezionata.
Se l'argomento **ordine** è omesso, viene assunta automaticamente la modalità UP.



10.3.8 Stringhe

ADDSTRING

Sintassi

ADDSTRING **nomestringa1, nomestringa2, nomestringa3**

Argomenti

nomestringa1 costante stringa o variabile stringa. Stringa sorgente
nomestringa2 costante stringa o variabile stringa. Stringa da aggiungere
nomestringa3 variabile stringa. Stringa risultato

Descrizione

Concatenamento di due stringhe.
 Aggiunge alla stringa identificata da **nomestringa1** la stringa identificata da **nomestringa2** e pone il risultato nella stringa identificata da **nomestringa3**.
 La dimensione massima di una stringa è di 255 caratteri + il terminatore; pertanto, il risultato del concatenamento delle prime due stringhe non dovrà superare questo limite.

Esempio

Vedi esempio [Operazioni sulle stringhe](#)

CONTROLCHAR

Sintassi

CONTROLCHAR **valore, nomestringa**

Argomenti

valore costante char o integer o variabile char o integer. Valore da convertire
nomestringa variabile stringa. Stringa risultato

Descrizione

Converte in carattere ASCII il valore identificato da **valore** e pone il risultato nella stringa **nomestringa** (in pratica nel primo byte).
 Il precedente contenuto della stringa è perso. Questa istruzione è utile, se si devono inserire in una stringa dei caratteri di controllo o non stampabili (ad esempio il carattere NULL = 0x00)
 Accetta stringhe di almeno 2 caratteri: 1 carattere + il terminatore. Se la stringa è di un solo carattere array[1] as char, viene segnalato un errore di sistema "Argomento macro errato".

Esempio

Vedi esempio [Operazioni sulle stringhe](#)

LEFT

Sintassi

LEFT **nomestringasorg, numcaratteri, nomestringadest**

Argomenti

nomestringasorg costante stringa o variabile stringa. Stringa sorgente
numcaratteri costante o variabile. Numero di caratteri da copiare
nomestringadest variabile stringa. Stringa destinazione

Descrizione

Copia i primi **numcaratteri** della stringa **nomestringasorg** nella stringa **nomestringadest**.
 In pratica viene prelevata la parte a sinistra della stringa sorgente. Vedere anche le istruzioni [MID](#) e [RIGHT](#).

Esempio

Vedi esempio [Operazioni sulle stringhe](#)

LEN

Sintassi

LEN **nomestringa, variabile**

Argomenti

nomestringa variabile stringa. Stringa
variabile variabile

Descrizione

Calcola il numero di caratteri contenuti nella stringa **nomestringa** (a esclusione del terminatore) e lo pone in **variabile**.

Esempio

Vedi esempio [Operazioni sulle stringhe](#)

MID

Sintassi

MID **nomestringasorg, primocar [, numcaratteri], nomestringadest**

Argomenti

nomestringasorg	costante stringa o variabile stringa. Stringa sorgente
numcaratteri	costante o variabile. Numero di caratteri da copiare
nomestringadest	variabile stringa. Stringa destinazione
primocar	costante o variabile. Posizione carattere partenza copia

Descrizione

Estrae dalla stringa identificata da **nomestringasorg** un numero di caratteri identificati da **numcaratteri**, a partire dalla posizione **primocar**.

La sottostringa estratta è posta nella stringa identificata da **nomestringadest**.

Se **numcaratteri** è omissso, viene copiata **stringasorg**, a partire dalla posizione **primocar** fino alla fine della stessa. In pratica viene prelevata la parte centrale della stringa sorgente. Vedere anche le istruzioni [LEFT](#) e [RIGHT](#).

Esempio

Vedi esempio [Operazioni sulle stringhe](#)

RIGHT

Sintassi

RIGHT **nomestringasorg, numcaratteri, nomestringadest**

Argomenti

nomestringasorg	costante stringa o variabile stringa. Stringa sorgente
numcaratteri	costante o variabile. Numero di caratteri da copiare
nomestringadest	variabile stringa. Stringa destinazione

Descrizione

Copia gli ultimi **numcaratteri** della stringa **nomestringasorg** nella stringa **nomestringadest**.

In pratica viene prelevata la parte destra della stringa sorgente. Vedere anche le istruzioni [LEFT](#) e [MID](#)

Esempio

Vedi esempio [Operazioni sulle stringhe](#)

SEARCH

Sintassi

SEARCH **nomestringa, carattere, variabile**

Argomenti

nomestringa	variabile stringa.
carattere	costante char o costante stringa o variabile stringa. Carattere o stringa da ricercare
variabile	variabile

Descrizione

Ricerca la posizione del carattere ASCII identificato da **carattere** (che può essere anche una stringa) all'interno della stringa **nomestringa** e pone l'indice del risultato nella **variabile**.

Se **carattere** non è stato trovato, **variabile** conterrà il valore -1.

Esempio

Vedi esempio [Operazioni sulle stringhe](#)

SETSTRING

Sintassi

SETSTRING **"valore", nomestringa**

Argomenti

valore costante stringa o variabile stringa (racchiusa tra doppi apici)
nomestringa stringa destinazione

Descrizione

Copia di una stringa.
 Esegue una copia dei caratteri ASCII presenti nella stringa identificata da "**valore**" nella stringa identificata da **nomestringa**.
 Per inserire caratteri non stampabili in una stringa vedere l'istruzione [CONTROLCHAR](#).

Esempio

Vedi esempio [Operazioni sulle stringhe](#)

STR**Sintassi**

STR **valore, nomestringa**

Argomenti

valore costante o variabile. Valore sorgente da convertire
nomestringa variabile stringa. Stringa destinataria

Descrizione

Converte in caratteri ASCII il **valore** e pone il risultato nella stringa **nomestringa**. Può essere utilizzata per trasformare una variabile di tipo integer in una stringa. Ad esempio, il numero 10 viene trasformato nella stringa "10".

Esempio

Vedi esempio [Operazioni sulle stringhe](#)

VAL**Sintassi**

VAL **nomestringa, risultato**

Argomenti

nomestringa variabile stringa. Stringa da convertire
risultato variabile. Stringa trasformata

Descrizione

Trasforma il contenuto della stringa **nomestringa** in un numero decimale e pone il risultato nella **variabile**. Ad esempio la stringa "123" viene trasformata nel valore 123.

Esempio

Vedi esempio [Operazioni sulle stringhe](#)

10.3.9 Comunicazioni

CLEARRECEIVE

Sintassi

CLEARRECEIVE

Argomenti

nessun argomento

Descrizione

Svuota la lista di RECEIVE eseguite, ma non ancora soddisfatte.

COMCLEARRXBUFFER

Sintassi

COMCLEARRXBUFFER **numeroCOM**

Argomenti

numeroCOM costante predefinita. Numero della porta seriale. I valori che possono essere assegnati sono: da **COM1** a **COM8**.

parity costante predefinita. Parità. I valori che possono essere assegnati sono: **NOPARITY**, **ODDPARITY** e **EVENPARITY**

Descrizione

Aprire una linea seriale. Va eseguita prima di qualunque altra istruzione che gestisca la linea seriale. Nel caso venga eseguita un'altra istruzione relativa alla stessa linea seriale prima di COMOPEN, verrà generato un errore di sistema. I parametri passati devono essere dei valori compresi tra quelli sopraindicati. Il canale di comunicazione con linea seriale è associato al task che ha eseguito l'istruzione COMOPEN. Se il task finisce, il canale di comunicazione viene chiuso in modo automatico. Vedere anche [COMCLOSE](#), [COMREAD](#), [COMWRITE](#), [COMREADSTRING](#), [COMWRITESTRING](#).

Nota

Il numero di linee seriali disponibili dipende dall'ambiente hardware del controllo numerico (consultare la documentazione del controllo). Nell'ambiente RTX sono disponibili solo COM1 e COM2.

COMREAD

Sintassi

COMREAD **numeroCOM**, **buffer**, **numchardaleggere**, **numcharletti** [,**timeout**]

Argomenti

numeroCOM costante predefinita. Numero della porta seriale. I valori che possono essere assegnati sono: da **COM1** a **COM8**.
buffer vettore di char. È il vettore dove vengono memorizzati i caratteri letti
numchardaleggere numero di caratteri che si desidera leggere dalla linea seriale
numcharletti numero di caratteri effettivamente letti
timeout timeout di attesa (in secondi)

Descrizione

L'istruzione legge dei caratteri dalla seriale **NumeroCOM**. I caratteri letti vengono memorizzati nella variabile **buffer**. Il campo **numchardaleggere** indica il numero di caratteri che l'istruzione dovrà leggere. Se nel buffer di ricezione della seriale ci sono meno caratteri ed il parametro **timeout** non è specificato, l'istruzione terminerà immediatamente, specificando nel parametro **numcharletti** il numero di caratteri effettivamente letti. Se il parametro **timeout** è specificato, allora l'istruzione dovrà attendere al massimo il numero di secondi indicato nella variabile, in attesa che arrivino altri caratteri. Se il **timeout** scade, allora l'istruzione uscirà, sempre specificando in **numcharletti** il numero di caratteri effettivamente copiato in **buffer**.

COMREADSTRING

Sintassi

COMREADSTRING **numeroCOM**, **buffer**, **numcharletti** [,**terminatore** [,**timeout**]]

Argomenti

numeroCOM costante predefinita. Numero della porta seriale. I valori che possono essere assegnati sono: da **COM1** a **COM8**.
buffer vettore di char. È il vettore dove vengono depositati i dati.
numcharletti numero di caratteri effettivamente letti
terminatore carattere di terminazione trasmissione
timeout timeout di attesa (in secondi)

Descrizione

L'istruzione legge dei caratteri dalla seriale **NumeroCOM**. Si differenzia dalla [COMREAD](#), perché legge dalla seriale, fino a che non trova il carattere terminatore. I caratteri letti vengono memorizzati nella variabile **buffer**. Questa variabile deve essere di tipo vettore di char. Il campo **numcharletti** indica il numero di caratteri che l'istruzione ha effettivamente letto dalla linea seriale e quindi copiato nel **buffer**. Il parametro **terminatore** indica il carattere che fungerà da terminatore della trasmissione. In pratica, l'istruzione dovrà leggere caratteri dalla seriale, fino a che non incontrerà un carattere uguale a quello specificato in questo parametro. Questo parametro è opzionale. Se non c'è, si intende che il carattere di terminazione è lo zero. Il carattere zero non sarà copiato nel buffer passato per parametro, mentre un eventuale carattere di terminazione specificato nell'istruzione verrà copiato nel buffer. Il **timeout** è un altro parametro che indica quanti secondi l'istruzione deve attendere che arrivino nuovi caratteri, se ha svuotato il buffer di ricezione, senza aver trovato il carattere di terminazione. Se il parametro **timeout** non è specificato, allora l'istruzione terminerà appena svuotato il buffer di ricezione.

Il parametro **flags** permette di specificare come l'informazione richiesta va trattata da Albatros. I valori ammessi e gli effetti che comportano sono i seguenti:

valore	comando	descrizione
\$0008H	CancelAfter	L'informazione verrà cancellata dopo esser stata letta
\$0800H	UpdateFlags	Modifica lo stato dell'informazione (letta / da leggere) senza modificare i dati
\$8000H	Delete	Cancella l'informazione

Il parametro **contenitore** è infine la variabile (o il dispositivo), in cui verrà memorizzata l'informazione che si sta richiedendo. Questa può essere omessa, in tal caso si sta richiedendo la notifica di un evento (può essere utilizzata per sincronizzare l'esecuzione del codice GPL su moduli differenti).

Elenco delle **sorgenti** gestite da Albatros e dei relativi comandi:

"@List"

Permette la gestione dei comandi Simulato e Setpoint.

Sono ammessi i seguenti comandi (parametro **identificatore**):

- Sim,0,contenitore: richiede lo stato del pulsante Simulato, che viene scritto sul flagSwitch Simulato. La variabile di ritorno **contenitore** vale 1 se non sono stati riscontrati errori, altrimenti vale 0.
- Setp,0,contenitore: richiede lo stato del pulsante Setpoint, che viene scritto sul flagSwitch CmdSetP. La variabile di ritorno **contenitore** vale 1 se non sono stati riscontrati errori, altrimenti vale 0.
- Esc,0,contenitore: richiede lo stato del pulsante Setpoint, che viene scritto sul flagSwitch Escluso. La variabile di ritorno **contenitore** vale 1 se non sono stati riscontrati errori, altrimenti vale 0.

"@Environ"

Permette di ottenere informazioni sullo stato del sistema: [livello di accesso](#) dell'utente, moduli collegati al supervisore, etc. L'informazione richiesta viene memorizzata nel parametro **contenitore**. I valori ammessi per il parametro **identificatore** e le corrispondenti risposte sono:

- "AccessLevel" livello di accesso al sistema 0=utente, 1=assistenza, 2=costruttore, 3=tpa
- "MaskConfModules" maschera dei moduli configurati
- "MaskActiveModules" maschera dei moduli collegati
- "CurrentModule" modulo da cui proviene la richiesta
- "mod:NamePC" nome del PC corrispondente al modulo "mod". (mod compreso tra 0 e 15)
- "LocalDateTime":

Il parametro **contenitore** riceverà la data e l'ora del PC in un formato legato al suo tipo:

- char: numero del giorno della settimana
- integer: numero di secondi dal 1/1/1970
- float: numero di giorni e frazioni di giorno dal 1/1/1900
- double: numero di giorni e frazioni di giorno dal 1/1/1900
- string: testo "AAAA/MM/GG hh:mm:ss"

Le maschere dei moduli collegati e configurati sono delle maschere di bit. Il bit di peso più basso corrisponde al modulo 0. Il bit di ogni modulo sarà ad 1, se il modulo è collegato o configurato. Nel caso di "NamePC" il numero di modulo è facoltativo, se omesso viene assunto il numero del modulo, da cui proviene la richiesta.

"@Syn"

Comunicazione tra GPL e visualizzatore dei quadri sinottici. Permette l'apertura e chiusura di quadri sinottici comandata da GPL e la richiesta di informazioni da una casella di un sinottico. Sono ammessi i seguenti comandi (parametro **identificatore**):

- "Open:nomefile" apertura del sinottico *nomefile.xsyn*
- "Close:nomefile" chiusura del sinottico *nomefile.xsyn*
- "nomecasella" casella da cui viene letta l'informazione richiesta

È possibile avere informazioni sulla finestra di movimentazione degli assi, secondo le specifiche definite anche per il parametro **sorgente** ["@Devices"](#) sottoriportato.

"@FileName"

Memorizza un'associazione tra una stringa costante e un nome di file componibile con variabili stringa. Da quando Albatros riceve la comunicazione dell'associazione, sostituisce tutti i successivi nomi di file con il nome ricevuto tramite questa istruzione. Il parametro **identificatore** è il nome del file. Il nome del file è una variabile stringa. Se nel parametro identificatore non è indicato il path completo in cui archiviare il file, Albatros considera path quello definito in tpa.ini nella sezione [tpa] alla voce dirreport. Il valore del parametro identificatore viene memorizzato in tpa.ini nella sezione [GPLFileName] alla voce Log, in modo da poterlo riutilizzare anche alle successive esecuzioni di Albatros. Per eliminare l'associazione, bisogna impostare come parametro identificatore una stringa vuota. L'associazione così definita vale per tutti i moduli.

"@FileDelete"

Cancellazione di un file. Il parametro **identificatore** è il nome del file che verrà cancellato (path completo). Se nel parametro **identificatore** non è indicato il path completo in cui archiviare il file, Albatros considera path quello definito in tpa.ini nella sezione [tpa] alla voce dirreport. Il nome del file può essere definito secondo le regole descritte in corrispondenza del parametro **sorgente** [@FileRead](#). Il parametro **contenitore** conterrà il valore:

- 1 se il file è stato cancellato
- 0 in caso contrario

"@FileRead"

Legge il contenuto di un file. Il parametro **identificatore** è il nome del file che verrà letto (percorso completo). Se nel parametro **identificatore** non è indicato il path completo in cui archiviare il file, Albatros considera path quello definito in tpa.ini nella sezione [tpa] alla voce dirreport. Se l'identificatore inizia e termina con un carattere di %, la stringa al suo interno viene cercata in tpa.ini nella sezione [tpa] e usata come nome del file. All'interno del nome possono essere inseriti dei caratteri che verranno sostituiti durante l'esecuzione dell'istruzione:

- %n numero del modulo che esegue l'istruzione RECEIVE
- %h ora corrente (formato 00-23)
- %d giorno corrente (formato 01-31)
- %m mese corrente (formato 01-12)
- %y anno corrente (formato a quattro cifre)

Se il parametro **contenitore** è definito come variabile char, conterrà un byte letto dal file; se è definito come stringa, conterrà un'intera riga del file testo; se è definito come integer, conterrà il numero di byte che mancano alla fine del file (0 = fine del file).

Per posizionare il puntatore a inizio file, il parametro **contenitore** deve essere omissivo.

"@FileExist"

Verifica l'esistenza di un file. Il parametro **identificatore** è il nome del file che verrà letto (percorso completo). Se nel parametro **identificatore** non è indicato il path completo in cui archiviare il file, Albatros considera path quello definito in tpa.ini nella sezione [tpa] alla voce dirreport. Il nome del file può essere definito secondo le regole descritte in corrispondenza del parametro **sorgente** [@FileRead](#). Il parametro **contenitore** conterrà il valore:

- diverso da 0 se il file esiste
- 0 se il file non esiste

"@FileLastWrite"

Ottiene la data dell'ultima modifica effettuata su un file. Il parametro **identificatore** è il nome del file (percorso completo). Se nel parametro **identificatore** non è indicato il percorso completo in cui archiviare il file, Albatros considera il percorso definito in tpa.ini nella sezione [tpa] alla voce dirreport. Il nome del file può essere definito secondo le regole descritte in corrispondenza del parametro **sorgente** [@FileRead](#). Il parametro **contenitore** conterrà la data dell'ultima modifica al file in un formato che è legato al tipo del parametro:

- char: numero del giorno della settimana
- integer: numero di secondi dal 1 gennaio 1970
- float: numero di giorni e frazioni di giorno dal 1 gennaio 1900
- double: numero di giorni e frazioni di giorno dal 1 gennaio 1900
- string: testo nel formato "AAAA/MM/GG hh:mm:ss"

"@FileInfo"

Legge alcune informazioni di un file. Il parametro **identificatore** deve essere espresso nella forma "proprietà:nomefile", dove **proprietà** indica il nome della proprietà da leggere e **nomefile** è il nome del file. Il nome del file può essere impostato tramite "Name". Il parametro **contenitore** conterrà il dato letto dal file.

Elenco delle proprietà:

- "version:": restituisce in contenitore un dato di tipo integer. I quattro numeri che identificano la versione si trovano nei 4 byte della variabile contenitore. Se si verifica un errore, il valore della variabile contenitore è 0.
- "size:": restituisce in contenitore un dato di tipo integer o float o double. Il dato è la dimensione del file. Se si verifica un errore, il valore della variabile contenitore è -1.

"@Devices"

Richiesta di apertura o chiusura della finestra Diagnostica relativa al modulo che invia l'informazione. Il parametro **identificatore** può assumere i seguenti valori:

- "Open" apertura della Diagnostica
- "Close" chiusura della Diagnostica

Il parametro **identificatore**, quando si vuole interagire con la finestra di movimentazione asse, può assumere i seguenti valori:

"MoveAX#nome_asse#HasFocus"	il parametro contenitore conterrà 1, se la la finestra di movimentazione asse specificata è attiva, altrimenti conterrà 0.
"MoveAX#nome_asse#Jog"	il parametro contenitore conterrà 1, se è impostato il movimento per spostamenti gestiti runtime dall'operatore, altrimenti conterrà 0.
"MoveAX#nome_asse#Step"	il parametro contenitore conterrà 1, se è impostato il movimento per spostamenti di passo predefinito, altrimenti conterrà 0.
"MoveAX#nome_asse#Absolute"	il parametro contenitore conterrà 1, se è impostato il movimento con spostamento a quota stabilita, altrimenti conterrà 0.

dove nome_asse rappresenta il nome dell'asse visualizzato nella finestra. Ad esempio, se si vuole verificare che la finestra di movimentazione asse X sia attiva, il parametro **identificatore** sarà "@MoveAX#X#HasFocus". Il nome dell'asse può essere in una delle seguenti forme:

1. Nome_Gruppo.Nome_Sottogruppo.Nome_Asse o Nome_Gruppo.Nome_Asse: viene fornito il percorso completo dell'asse.
2. Nome_Asse: per identificare l'asse corretto, vengono fatte nell'ordine le verifiche:
 - se il task da cui arriva il comando è una funzione di sottogruppo, l'asse viene cercato in quel sottogruppo.
 - se il task da cui arriva il comando è una funzione del sottogruppo principale, l'asse viene cercato in tutto il gruppo. Se è presente più di un asse con quel nome, la ricerca fallisce.
 - se le verifiche precedenti sono fallite, l'asse viene cercato in tutti i gruppi del modulo. Se è presente più di un asse con nome Nome_Asse, la ricerca non ha esito positivo.

"@Vars"

Richiede l'aggiornamento di una variabile globale GPL. Permette di effettuare un rinfresco dei dati delle Parametriche tecnologiche e utensili. I dati di parametrica vengono normalmente inviati al GPL durante l'inizializzazione della macchina. Il parametro **identificatore** assume il significato di nome della variabile globale (di macchina o di gruppo), di cui viene richiesto l'aggiornamento. Il parametro **contenitore** conterrà il valore:

- 1 se la variabile è stata aggiornata correttamente
- 0 in caso contrario

"@Application"

Interazione con Albatros. Permette di visualizzare dei "message box" sullo schermo e di chiudere Albatros. I valori ammessi per il parametro **identificatore** sono:

"Quit" chiude Albatros

"IsLocked" Verifica se l'uscita da Albatros è bloccata. Il parametro **contenitore** conterrà 1, se l'interfaccia è bloccata, 0 se è possibile uscire da Albatros.

"MsgBox" legge la risposta di un message box aperto precedentemente con una SEND

Il parametro **contenitore** permette di sapere, nel caso di un message box, quale pulsante è stato premuto dall'operatore:

- 1 pulsante "OK"
- 2 pulsante "Annulla"
- 4 pulsante "Riprova"
- 6 pulsante "Sì"
- 7 pulsante "No"

Nel caso del comando "Quit" il parametro **contenitore** conterrà il valore:

- 1 se Albatros è stato chiuso correttamente
- 0 in caso contrario

"@Param"

Permette di conoscere il numero progressivo di registrazione dei file di parametrica Partec.xpar e Partool.xpar. L'informazione richiesta viene memorizzata nel parametro **contenitore**. I valori ammessi per il parametro **identificatore** sono:

- "partec" richiede il progressivo di registrazione di partec.xpar
- "partool" richiede il progressivo di registrazione di partool.xpar

"@Ini"

Legge una combinazione di chiave=valore dal file tpa.ini. Il parametro **identificatore** è il nome della chiave da leggere nel file tpa.ini nella sezione [Tpa]. Per leggere da una sezione specifica, al nome della chiave si deve aggiungere il nome della sezione tra parentesi quadre. ("[Sezione]Chiave").

"@ShellExecute"

Chiede al sistema operativo di aprire un file utilizzando il programma associato all'estensione del file. È anche possibile lanciare un eseguibile. Il parametro **identificatore** è il nome del file da aprire o il nome del programma da lanciare. Il nome del file può essere dichiarato con un percorso completo, altrimenti viene cercato nella cartella corrente di Albatros. Il nome del file viene anche cercato tra quelli definiti tramite "@FileName". Il parametro **contenitore** conterrà il valore 0 se non si sono verificati errori nell'apertura del file, altrimenti conterrà il codice di errore.

"@StartProg"

Esegue il programma definito nel parametro **identificatore**. Non è possibile passare argomenti al programma da lanciare. Il nome del programma deve contenere l'intero percorso, altrimenti, viene cercato nella cartella corrente di Albatros. Il nome del programma viene anche cercato tra quelli definiti tramite "@FileName". Il parametro **contenitore** conterrà il valore 0, se è riuscito il lancio del programma, altrimenti conterrà il codice di errore. Se il programma era già stato lanciato, il codice d'errore è 1056.

"@TermProg"

Termina il programma definito nel parametro **identificatore** e lanciato tramite "@StartProg". Il nome del programma deve contenere l'intero percorso, altrimenti viene cercato nella cartella corrente di Albatros. Il nome del programma viene cercato anche tra quelli definiti tramite "@FileName". Il parametro **contenitore** conterrà il valore 0, se è riuscito il lancio del programma, altrimenti conterrà il codice di errore. Se il programma era già stato lanciato, il codice d'errore è 1056.

"@ProgRunning"

Verifica se il programma lanciato con "@StartProg" è ancora in esecuzione. Il nome programma è definito nel parametro **identificatore**. Il nome del programma deve contenere l'intero percorso, altrimenti viene cercato nella cartella corrente di Albatros. Il nome del programma viene anche cercato tra quelli definiti tramite "@FileName". Il parametro **contenitore** conterrà il valore 1, se il programma è ancora in esecuzione, altrimenti conterrà valore 0.

"@DialogFile"

Aprire la finestra di dialogo di File Apri o di File Salva per permettere la scelta di un nome di file. Per aprire la finestra di File Apri, impostare il parametro **identificatore** = "Open", per aprire la finestra di File Salva, impostare il parametro **identificatore** = "Save". Il nome del file scelto viene memorizzato nel parametro **contenitore**.

"@AxisCorrectors"

Sostituisce la tabella dei correttori di linearità di un asse con una nuova tabella, caricata da file, che deve comunque avere lo stesso numero di correttori e gli stessi assi per i correttori incrociati. Il parametro **identificatore** è il nome del file, che tipicamente ha estensione .csv e si trova nella cartella ...\\Mod.n\\Config (il nome del file viene 'normalizzato' come avviene ad esempio per "@FileExist"). Il parametro **contenitore** è definito come variabile integer e conterrà 1, se i nuovi correttori sono stati mandati, altrimenti 0.

"@Language"

Riceve il testo traducibile corrispondente ad un messaggio di gruppo o di libreria o di modulo, che è associato a una istruzione MESSAGE o a una istruzione ERROR. I valori ammessi per il parametro **identificatore** sono:

- DEFMSG:numero", dove numero è una sequenza di cifre. Albatros scrive in **contenitore** il testo del messaggio di modulo numero "numero".
- "DEFMSG:nome", dove "nome" è il nome, compreso gruppo e libreria, di una DEFMSG. Albatros scrive in **contenitore** il testo del messaggio indicato. Se manca il nome del gruppo o della libreria viene usato il nome del task che ha inviato la RECEIVE.
- "DEFMSG:*", Albatros scrive in **contenitore** il testo del messaggio di gruppo o di modulo precedentemente indicato tramite l'istruzione SEND.

Esempio

```

; in GPL
RECEIVE "@Param", "partec", 0, prog
RECEIVE "@Param", "partool", 0, prog

; in GPL
; legge il valore della chiave Radix nella sezione [Albatros] dal file
tpa.ini
RECEIVE "@INI", "[Albatros]Radix", 0, valore

; apre la finestra di File Apri e memorizza il nome del file nella
variabile NomeFile
RECEIVE "@DialogFile", "Open", 0, NomeFile

; lettura completa di un file
Function ReadProperties
PARAM file AS STRING
LOCAL version AS INTEGER

```

LOCAL size AS DOUBLE

```
SEND "@FileName" "theFile" 0 file
WAITRECEIVE "@FileInfo", "version:theFile", 0, version
WAITRECEIVE "@FileInfo", "size:theFile", 0, size
```

SEND

Sintassi

SEND [destinatario,] identificatore, flags [, informazione]

Argomenti

destinatario	costante di tipo stringa
identificatore	costante di tipo stringa
flags	costante di tipo integer
informazione	nome di dispositivo o costante o variabile (numerica o stringa)

Descrizione

Questa istruzione, assieme alla RECEIVE, viene utilizzata per scambiare informazioni tra i moduli dell'impianto ed il PC supervisore. La SEND viene utilizzata per inviare informazioni, la RECEIVE per richiederle. Le informazioni possono essere inviate ad Albatros o ad un programma esterno (Server OLE Automation). Nel secondo caso l'informazione sarà comunque ricevuta da Albatros, che si occuperà di inviarla a sua volta al programma esterno.

Il parametro **destinatario** è una stringa che permette di specificare a chi è indirizzata l'informazione. Ci sono tre classi di destinatari:

- destinatari che iniziano con il carattere "@" (vedere l'elenco più avanti). Il destinatario è in realtà Albatros o meglio una delle sue funzionalità.
- destinatari che non iniziano con il carattere "@". Vengono considerati come Server OLE, alla ricezione della prima informazione indirizzata ad essi Albatros cercherà di mandarli in esecuzione e, quindi, di passargli l'informazione ricevuta dal modulo.
- destinatario non specificato (il parametro è infatti opzionale). In questo caso l'informazione viene conservata da Albatros in una tabella e rimane disponibile per chi ne fa richiesta (un altro modulo o un programma esterno).

Il parametro **identificatore** è il nome dell'informazione, non può essere omissso. Assume diversi significati in funzione del destinatario:

- se il destinatario è Albatros sarà un comando legato alla funzionalità cui si sta accedendo
- se il destinatario è un Server OLE sarà una proprietà dell'oggetto OLE istanziato.
- se il destinatario non è specificato sarà l'etichetta che identifica l'informazione all'interno della tabella conservata da Albatros.

Il parametro **flags** permette di specificare come l'informazione va trattata da Albatros. I valori ammessi e gli effetti che comportano sono i seguenti:

valore	comando	descrizione
\$0001H	Broadcast	Normale invio dell'informazione
\$0008H	CancelAfter	L'informazione verrà cancellata dopo essere stata letta
\$0020H	ReadOnly	L'informazione può esser cancellata solo dal mittente
\$1000H	UpdateFlags	Modifica lo stato dell'informazione (letta / da leggere), senza modificare i dati
\$8000H	Delete	Cancella l'informazione

Il parametro **informazione** è infine l'informazione che si sta inviando. Questa può essere omissa e, in tal caso, l'invio di una informazione vuota assume il significato di una notifica di un evento (può essere utilizzata per sincronizzare l'esecuzione del codice GPL su moduli differenti). Sono ammessi, come parametro informazione, tutti i dispositivi (esclusi gli assi), le variabili semplici del GPL e le stringhe.

Elenco dei **destinatari** gestiti da Albatros e dei relativi comandi:

"@List"

Permette la gestione dei comandi Simulato e Setpoint

Sono ammessi i seguenti comandi (parametro **identificatore**):

- Sim: notifica il cambiamento di stato del flag switch Simulato. In base allo stato del flag viene visualizzato premuto o rilasciato il pulsante che lo identifica nella barra degli Strumenti (1=selezionato, 0=deselezionato)

- **Setp**: notifica il cambiamento di stato del flag switch CmdSetp. In base allo stato del flag viene visualizzato premuto o rilasciato il pulsante che lo identifica nella barra degli Strumenti (1=selezionato, 0=deselezionato)
- **Esc**: notifica il cambiamento di stato del flag switch Escluso. In base allo stato del flag viene visualizzato premuto o rilasciato il pulsante che lo identifica (lo stesso del flag switch CmdSetp) nella barra degli Strumenti (1=selezionato, 0=deselezionato)
- **End**: termina l'esecuzione della lista. Questo comando abbassa i pulsanti di Start e Stop e disabilita le voci di menu Start e Stop
- **Hold**: abbassa il pulsante di Stop e abilita la voce di menu Stop. Alza il pulsante di Start e disabilita la voce di menu Start

"@Syn"

Comunicazione tra GPL e visualizzatore dei quadri sinottici. Permette l'apertura e chiusura di quadri sinottici comandata da GPL e l'invio di informazioni in una casella di un sinottico. Sono ammessi i seguenti comandi (parametro **identificatore**):

- "Open:*nomefile*" apertura del sinottico *nomefile.xsyn*
- "Close:*nomefile*" chiusura del sinottico *nomefile.xsyn*
- "Open" apertura di un sinottico. Il nome del file viene letto dalla variabile

informazione

- "Close" chiusura di un sinottico. Il nome del file viene letto dalla variabile

informazione

- "*nomecasella*" casella in cui va visualizzata l'informazione inviata

È possibile interagire con la finestra di movimentazione degli assi secondo le specifiche definite anche per il parametro **destinatario** "@Devices" sottoriportato.

"@File"

Scrittura su file. Permette di creare dei file di log personalizzati in cui memorizzare le operazioni eseguite da una macchina. Sono dei file di testo (ASCII). Il parametro **identificatore** è il nome del file su cui verrà effettuata la scrittura.

Il nome del file è una stringa costante. Se nel parametro identificatore non è indicato il path completo in cui archiviare il file, Albatros considera path quello definito in tpa.ini nella sezione [tpa] alla voce dirreport.

Se l'identificatore inizia e termina con un carattere di % la stringa al suo interno viene cercata in tpa.ini nella sezione [tpa] e usata come nome del file. All'interno del nome possono essere inseriti dei caratteri che verranno sostituiti durante l'esecuzione dell'istruzione:

- %n numero del modulo che esegue l'istruzione SEND
- %h ora corrente (formato 00-23)
- %d giorno corrente (formato 01-31)
- %m mese corrente (formato 01-12)
- %y anno corrente (formato a quattro cifre)

Vedere l'esempio.

Le operazioni di scrittura avvengono in modalità append (i dati vengono aggiunti in fondo al file). Su un file possono essere inviati dati numerici (che vengono convertiti automaticamente in ASCII) o stringhe. È possibile scrivere delle stringhe in formato data e ora utilizzando i caratteri di formato %d per la data e %t per l'ora. Per l'ora viene utilizzato il formato "HH:mm:ss" (ovvero: ore, minuti e secondi separati da ":") e per la data un formato dipendente dalle impostazioni nazionali. È possibile utilizzare un altro formato impostando in tpa.ini nella sezione [Albatros] la voce "LogNoLocale=1" (il default è LogNoLocale=0, ossia uso del formato corrente). È anche possibile impostare il formato da utilizzare per la data e l'ora, indipendentemente dal formato impostato in Windows, definendo sempre in tpa.ini nella sezione [Albatros] le voci "LogDateFormat=" e "LogTimeFormat=" e assegnando una stringa di caratteri secondo lo schema indicato di seguito. Se queste voci non sono presenti o sono vuote vengono usati i formati impostati in Windows.

Formato dell'ora

h	ora nel formato 12 ore senza zeri iniziali
hh	ora nel formato 12 ore con zeri iniziali
H	ora nel formato 24 ore senza zeri iniziali
HH	ora nel formato 24 ore con zeri iniziali
m	minuti senza zeri iniziali
mm	minuti con zeri iniziali
s	secondi senza zeri iniziali
ss	secondi con zeri iniziali
t	un solo carattere per indicare il time marker, ad esempio A o P
tt	più caratteri per indicare il time marker, ad esempio AM o PM

Nota: i formati "t" e "tt" usano il time marker indicato nel pannello di controllo dell'utente corrente. Non è necessariamente "AM" e "PM".

Esempio: se sono le 11:29 del pomeriggio e la stringa è così composta "hh':'mm':'ss tt", verrà scritto "11:29:40 PM".

Formato del giorno

d	giorno del mese senza zeri iniziali, rappresentato in cifra
dd	giorno del mese con zeri iniziali,, rappresentato in cifra
ddd	giorno della settimana, rappresentato in caratteri e abbreviato a tre lettere
dddd	giorno della settimana, rappresentato in caratteri con il nome completo
M	mese senza zeri iniziali, rappresentato in cifre
MM	mese con zeri iniziali, , rappresentato in cifre
MMM	mese, rappresentato in caratteri e abbreviato a tre lettere
MMMM	mese, rappresentato in caratteri con il nome completo
y	anno con due cifre senza zeri iniziali per anni inferiori a 10
yy	anno con due cifre con zeri iniziali per anni inferiori a 10
yyyy	anno rappresentato con quattro o cinque cifre a seconda del calendario in uso
yyyyy	anno rappresentato con quattro o cinque cifre a seconda del calendario in uso

Esempio: se è mercoledì 31 Agosto 1994 e la stringa è così composta "ddd',' MMM dd yy", verrà scritto "Mer, Agosto 31 94"

Se l'informazione viene omessa viene aggiunto al file un "ritorno a capo".

"@FileName"

Memorizza un'associazione tra una stringa costante e un nome di file componibile con variabili stringa. Da quando Albatros riceve la comunicazione dell'associazione sostituisce tutti i successivi nomi di file con il nome ricevuto tramite questa istruzione. Il parametro **identificatore** è il nome del file su cui verrà effettuata la scrittura. Il nome del file è una variabile stringa. Se nel parametro identificatore non è indicato il path completo in cui archiviare il file, Albatros considera path quello definito in tpa.ini nella sezione [tpa] alla voce dirreport. Il valore del parametro identificatore viene memorizzato in tpa.ini nella sezione [GPLFileName] alla voce Log, in modo da poterlo riutilizzare anche alle successive esecuzioni di Albatros. Per eliminare l'associazione bisogna impostare come parametro identificatore una stringa vuota. L'associazione così definita vale per tutti i moduli.

"@FileDelete"

Cancellazione di un file. Il parametro **identificatore** è il nome del file che verrà cancellato (path completo). Se nel parametro identificatore non è indicato il path completo in cui archiviare il file, Albatros considera path quello definito in tpa.ini nella sezione [tpa] alla voce dirreport. Il nome del file può essere definito secondo le regole descritte in corrispondenza del parametro **destinatario** [@File](#).

"@FileRead"

Posiziona il puntatore al file a inizio file. Il parametro **identificatore** è il nome del file (path completo). Se nel parametro identificatore non è indicato il path completo in cui archiviare il file, Albatros considera path quello definito in tpa.ini nella sezione [tpa] alla voce dirreport. Il nome del file può essere definito secondo le regole descritte in corrispondenza del parametro **destinatario** [@FileRead](#).

"@Axis"

Interagisce con la finestra di movimentazione manuale degli assi secondo le specifiche definite anche per il parametro destinatario "@Devices" sottoriportato. Se una finestra che controlla il movimento dell'asse indicato è già aperta, il comando agisce su questa finestra sia che sia aperta in un sinottico, sia che sia aperta in diagnostica. Se la finestra è chiusa, il comando cerca di aprirla in diagnostica o in uno dei sinottici già aperti e che contiene quell'asse.

"@Devices"

Richiesta di apertura o chiusura della finestra Diagnostica relativa al modulo che invia l'informazione. Esecuzione di comandi all'interno della finestra di movimentazione asse in diagnostica. Il parametro **identificatore** può assumere i seguenti valori:

- "Open" apertura della Diagnostica
- "Close" chiusura della Diagnostica

Il parametro **identificatore**, quando si vuole interagire con la finestra di movimentazione asse, può assumere i seguenti valori:

"MoveAX#nome_asse#Open"	apertura della finestra di movimentazione asse.
"MoveAX#nome_asse#Close"	chiusura della finestra di movimentazione asse.
"MoveAX#nome_asse#Plus"	pressione del pulsante di movimento asse in direzione positiva
"MoveAX#nome_asse#Minus"	pressione del pulsante di movimento asse in direzione negativa
"MoveAX#nome_asse#Stop"	pressione del pulsante di stop del movimento
"MoveAX#nome_asse#Jog"	imposta la modalità di movimento per spostamenti gestiti runtime dall'operatore
"MoveAX#nome_asse#Step"	imposta la modalità di movimento per spostamenti di passo predefinito
"MoveAX#nome_asse#Absolute"	imposta la modalità di movimento con spostamento a quota stabilita dell'asse

dove nome_asse rappresenta il nome dell'asse visualizzato nella finestra. Ad esempio, se si vuole aprire la finestra di movimentazione asse X il parametro **identificatore** sarà "[@MoveAX#X#Open](#)". Il nome dell'asse può essere in una delle seguenti forme:

1. Nome_Gruppo.Nome_Sottogruppo.Nome_Asse o Nome_Gruppo.Nome_Asse: viene fornito il percorso completo dell'asse.
2. Nome_Asse: per identificare l'asse corretto vengono fatte nell'ordine le verifiche:
 - se il task da cui arriva il comando è una funzione di sottogruppo, l'asse viene cercato in quel sottogruppo.
 - se il task da cui arriva il comando è una funzione del sottogruppo principale, l'asse viene cercato in tutto il gruppo. Se è presente più di un asse con quel nome la ricerca fallisce.
 - se le verifiche precedenti sono fallite, l'asse viene cercato in tutti i gruppi del modulo. Se è presente più di un asse con nome Nome_Asse la ricerca non ha esito positivo.

È possibile impedire all'utente di agire sui tasti di movimento asse di tutte le finestre di movimentazione asse del modulo in diagnostica, impostando il parametro **identificatore** nel seguente modo:

- "MoveAX##UIENABLE" se il parametro **informazione** viene impostato a 0, il movimento assi da Albatros viene disabilitato; se viene impostato a 1 e viene abilitato il movimento assi da Albatros.

La disabilitazione del movimento assi da Albatros è consigliata quando si muovono gli assi da pulsantiera di macchina.

"@Vars"

Richiede il salvataggio del contenuto di una variabile globale GPL nell'archivio dei parametri tecnologici o utensili. Il parametro **identificatore** è il nome della variabile globale (di macchina o di gruppo o di libreria) di cui viene richiesto l'aggiornamento.

"@Application"

Interazione con Albatros. Permette di chiudere Albatros o visualizzare dei "message box" sullo schermo per informare l'utente o chiedere il consenso a successive attività. I valori ammessi per il parametro **identificatore** sono:

"Quit" chiude Albatros
 "Lock" impedisce la chiusura di Albatros da **File->Esci** o da combinazione di tasti [ALT+F4] o da bottone di chiusura.
 "UnLock" ripristina la possibilità di chiudere Albatros
 "MsgBox:flags" apre un message box
 "

Il comportamento dei message box viene controllato dalla parte "flags" della stringa **identificatore**. Questa può essere una sequenza dei seguenti caratteri (possono essere indifferentemente maiuscoli o minuscoli):

"O"	pulsante "OK"
"C"	pulsante "Annulla"
"Y"	pulsante "Sì"
"N"	pulsante "No"
"R"	pulsante "Riprova"
"S"	icona del cartello Stop
"?"	icona informativa, composta da una lettera 'i' minuscola all'interno di un cerchio
"!"	icona con punto esclamativo
"*"	icona informativa
"1"	il primo pulsante è quello di default
"2"	il secondo pulsante è quello di default
"3"	il terzo pulsante è quello di default
"4"	il quarto pulsante è quello di default

Se non indicato, il pulsante di default è il primo.

Ad esempio, "MsgBox:YN2" definisce un message box con l'icona informativa, due pulsanti "Sì" e "No" di cui il secondo è quello di default. Il parametro **informazione** può essere una stringa, che contiene il testo da visualizzare, oppure un numero intero che viene interpretato come codice di un messaggio di modulo gestito da TpaLangs.exe, oppure un'etichetta di un messaggio di gruppo definito con l'istruzione [DEFMSG](#).

Per quanto riguarda il testo, se al suo interno è presente il carattere di ritorno a capo, "\u000A", il testo verrà diviso in due e sarà visualizzata la prima parte come testo del message box, mentre la seconda parte verrà visualizzata come spiegazione, o dettaglio del testo.

La lingua nella quale vengono visualizzati i pulsanti è quella di Windows.

"@Help"

Apertura di un file di help. Permette di comandare la visualizzazione un file di help specificando l'argomento da visualizzare. I valori ammessi per il parametro **identificatore** sono:

- "Open:nomefile" apertura di un file di help
- "Close:nomefile" chiusura di un file di help

La parte "nomefile" della stringa specifica il nome del file di help da aprire.

Il parametro **informazione** può essere una stringa o un numero e assume rispettivamente il significato di chiave o di numero di contesto (serve per identificare la pagina o argomento dell'help che si vuole visualizzare).

"@Report"

Aggiunge segnalazioni al file di report di Albatros (MONTH(n. mese).TER). Il parametro **identificatore** è:

- "Add"

Il parametro **informazione** può essere:

- una variabile stringa o una costante stringa: in questo caso viene memorizzato il testo contenuto nella stringa
- una variabile integer o un valore numerico integer: in questo caso viene memorizzato il testo definito nell'istruzione [DEFMSG](#).

"@Ini"

Scrivere una combinazione di chiave=valore nel file tpa.ini. Il parametro **identificatore** è il nome della chiave da aggiungere nel file tpa.ini nella sezione [Tpa]. Per scrivere in una sezione specifica, al nome della chiave si deve aggiungere il nome della sezione tra parentesi quadre.("[Sezione]Chiave")

Il parametro **informazione** può essere una variabile stringa o numerica, una costante stringa o numerica.

"@ShellExecute"

Chiede al sistema operativo di aprire un file utilizzando il programma associato all'estensione del file. È anche possibile lanciare un eseguibile. Il parametro **identificatore** è il nome del file da aprire o il nome del programma da lanciare. Il nome del file può essere dichiarato con un percorso completo, altrimenti viene cercato nella cartella corrente di Albatros. Il nome del file viene anche cercato tra quelli definiti tramite "@FileName".

"@StartProg"

Esegue il programma definito nel parametro **identificatore**. Non è possibile passare argomenti al programma da lanciare. Il nome del programma deve contenere l'intero percorso, altrimenti viene cercato nella cartella corrente di Albatros. Il nome del programma viene cercato anche tra quelli definiti tramite "@FileName".

"@TermProg"

Termina il programma definito nel parametro **identificatore** e lanciato tramite "@StartProg". Il nome del programma deve contenere l'intero percorso, altrimenti viene cercato nella cartella corrente di Albatros. Il nome del programma viene cercato anche tra quelli definiti tramite "@FileName".

"@DialogFile"

Permette di impostare alcuni parametri relativi alla finestra di dialogo di File Apri o di File Salva. I valori ammessi per il parametro **identificatore** sono:

"Extension"	se l'utente non inserisce un'estensione viene utilizzata quella definita nel parametro informazione (variabile o costante stringa)
"Filter"	imposta il filtro sui tipi di file da utilizzare. Il parametro informazione può essere una variabile stringa o una costante stringa e in questo caso viene utilizzato come filtro il testo contenuto nella stringa, una variabile integer o un valore numerico integer e in questo caso viene utilizzato come filtro il testo definito nell'istruzione DEFMSG .
"Flags"	imposta i flag di inizializzazione. Per l'elenco dei valori da impostare nel campo informazione (variabile o costante integer) fare riferimento alla documentazione ufficiale Microsoft relativa al membro Flags della struttura OPENFILENAME.
"InitalDir"	imposta la cartella iniziale, definita nel campo informazione (variabile o costante stringa)
"Title"	imposta il titolo della finestra. Il parametro informazione può essere una variabile stringa o una costante stringa e in questo caso viene utilizzato come titolo il testo contenuto nella stringa, una variabile integer o un valore numerico integer e in questo caso viene utilizzato come titolo il testo definito nell'istruzione DEFMSG .

"@Language"

Imposta il numero di messaggio di gruppo o di modulo o di libreria che verrà utilizzato nella successiva RECEIVE con stesso identificatore. Il valore ammesso per il parametro **identificatore** è "DEFMSG:*". Il parametro **informazione** può essere una variabile di tipo integer o costante integer e in questo caso definisce il numero del messaggio di gruppo da visualizzare. Può essere di tipo stringa o costante stringa e in questo caso definisce il nome della DEFMSG.

Esempio

```
; Esempio di istruzione send file con nome costruito in esecuzione.
; Supponiamo che la data in cui viene eseguita l'istruzione
; sia 31-01-2000
```



```

; in GPL
SEND "@File", "%Log%", 0, "Inizio esecuzione"
; aggiunge un "ritorno a capo"
; nel file tpa.ini alla sezione [TPA] si aggiunge
SEND "@File", "%Log%", 0
Log=c:\Albatros\report\%y\Rep%m%d.txt

; Il nome del file risultante è:
c:\Albatros\report\2000\Rep0131.txt

; Esempio di istruzione send Vars
; si definisce una variabile Var_SendVars
; as double nel file delle variabili globali
; in parametrica Tecnologica si inserisce Var_SendVars
; nel campo Nome Matrice
; in GPL
SETVAL 100.0,Var_SendVars
; invia il valore 100.0 al parametro della parametrica Tecnologica
; associato alla variabile Var_SendVars
SEND "@Vars", "Var_SendVars", 0

; Esempio di istruzione send INI
; si scrive in tpa.ini la chiave Radix nella sezione [Albatros]
; per impostare una base numerica di visualizzazione decimale
SEND "@INI", "[Albatros]Radix", 0,1

; Esempio di impostazione di un'associazione tra stringa costante GPL
; e nome di un file.

; dichiarazione di una variabile stringa
nomefile as string
; composizione del nome del file
setstring "C:\albatros\report\LogFile.txt",nomefile
; associazione
SEND "@FileName", "LOG",0,nomefile
; tutte le operazioni di scrittura vengono da questo momento
; eseguite sul file definito dalla variabile nomefile
SEND "@File", "LOG",0,"Scrivo nel file LOG"

```

SENDIPC

Sintassi

```

SENDIPC nomeIPC, attesa [, nomevar1 [, nomevarN, ...]]
SENDIPC nomeIPC, attesa , matrice[riga]
SENDIPC nomeIPC, attesa , vettore
SENDIPC nomeIPC, attesa , matrice

```

Argomenti

nomeIPC	costante stringa. Nome della IPC
attesa	costante predefinita. Modo di attesa di lettura del comando I valori ammessi sono: WAIT attende la lettura del comando NOWAIT non attende la lettura del comando
nomevar1[...nomeVarN]	costante o variabile. Nomi variabili 1÷N
matrice[riga]	costante o variabile integer. Numero di riga della matrice
vettore	Nome di vettore
matrice	Nome di matrice

Descrizione

Invia un comando IPC alla memoria condivisa "**nomeIPC**".
La prima volta che viene eseguita un'istruzione di SENDIPC, viene allocata la memoria condivisa, la cui dimensione viene calcolata in base alla dimensione dei dati che vengono inviati. La dimensione massima della memoria condivisa è di 64 Kb. Possono essere definite al massimo 48 memorie condivise, identificate da 48 nomi univoci.

Alla memoria condivisa è associato un semaforo, che permette di sincronizzare l'esecuzione dei task che vi accedono. Il task che scrive i dati attiva il semaforo al termine della scrittura, il task che legge i dati lo disattiva al termine della lettura.

Se come parametro **attesa** è stato indicato WAIT, il task che ha scritto i dati attenderà che questi vengano letti (semaforo disattivato), prima di proseguire l'esecuzione.

Una SENDIPC senza dati si riduce ad un sincronismo tra task. In questo caso non viene allocata la memoria condivisa.

IPC intermodulo

Due moduli remoti possono scambiare dati tramite IPC. Questi IPC vengono chiamati IPC intermodulo. Per definire un IPC intermodulo bisogna scrivere il **nomeIPC** secondo il seguente formalismo: numero del modulo sorgente, "->", numero del modulo destinatario, ":", a seguire gli altri caratteri del nome dell'IPC.

Ad esempio "0->1:Parametri Base".

Vedere anche [WAITIPC](#) e [TESTIPC](#).

WAITIPC

Sintassi

WAITIPC	nomeIPC [, nomevar1 [, nomevarN, ...]]
WAITIPC	nomeIPC, matrice[riga]
WAITIPC	nomeIPC, vettore
WAITIPC	nomeIPC, matrice

Argomenti

nomeIPC	costante stringa.Nome della IPC
nomevar1[...nomeVarN]	costante o variabile. Nomi variabili 1÷N
matrice[riga]	costante o variabile integer. Numero di riga della matrice
vettore	Nome di vettore
matrice	Nome di matrice

Descrizione

Riceve un comando IPC dalla memoria condivisa "**nomeIPC**".

La prima volta che viene eseguita un'istruzione di WAITIPC viene allocata la memoria condivisa, la cui dimensione viene calcolata in base alla dimensione dei dati che vengono inviati. La dimensione massima della memoria condivisa è di 64 Kb. Possono essere definite al massimo 48 memorie condivise, identificate da 48 nomi univoci.

Alla memoria condivisa è associato un semaforo che permette di sincronizzare l'esecuzione dei task che vi accedono. Il task che legge i dati attende che il semaforo venga attivato dal task che scrive i dati, legge i dati e disattiva il semaforo.

Una WAITIPC senza dati si riduce ad un sincronismo tra task. In questo caso non viene allocata la memoria condivisa.

Vedere anche [SENDIPC](#) e [TESTIPC](#).

WAITRECEIVE

Sintassi

WAITRECEIVE	[sorgente,] identificatore, flags [, contenitore]
--------------------	---

Argomenti

sorgente	costante di tipo stringa
identificatore	costante di tipo stringa
flags	costante di tipo integer
contenitore	nome di dispositivo o variabile (numerica o stringa)

Descrizione

Attende che l'informazione richiesta (specificata da **identificatore**) sia arrivata prima di continuare l'esecuzione del programma GPL. Per l'utilizzo fare riferimento alla documentazione dell'istruzione [RECEIVE](#).

10.3.10 Matematica

ABS

Sintassi

ABS	operando, risultato
------------	----------------------------

Argomenti

operando costante o variabile o nome di dispositivo
risultato variabile o nome di dispositivo

Descrizione

Ricava il valore assoluto di **operando** e lo pone in **risultato**. Per la conversione dei dati, in base al tipo di dato dichiarato, riferirsi al capitolo [Conversione dati](#).

Esempio

```
SetVal -10,op ; assegna -10 alla variabile op
Abs op,var ; Il valore posto nella variabile var sarà 10
```

ADD**Sintassi**

ADD **operando1, operando2, risultato**

Argomenti

operando1 costante o variabile o nome di dispositivo
operando2 costante o variabile o nome di dispositivo
risultato variabile o nome di dispositivo

Descrizione

Esegue una operazione di somma tra **operando1** e **operando2** e pone il risultato in **risultato**. Per la conversione dei dati, in base al tipo di dato dichiarato, riferirsi al capitolo [Conversione dati](#).

Esempio

```
SetVal 5,op1 ; assegna 5 alla variabile op1
Add op1,3,var
;Il valore posto nella variabile var sarà 8
```

AND**Sintassi**

AND **operando1, operando2, risultato**

Argomenti

operando1 costante o variabile o nome di dispositivo
operando2 costante o variabile o nome di dispositivo
risultato variabile o nome di dispositivo

Descrizione

Esegue una operazione di AND binario (*tra due bit, risulta 1 solo se entrambi valgono 1*) tra **operando1** e **operando2** e pone il risultato in **risultato**. Per la conversione dei dati, in base al tipo di dato dichiarato, riferirsi al capitolo [Conversione dati](#).

Esempio

```
;Il valore posto nella variabile var sarà 1
;(Notazione binaria: 5 = 0101, 3 = 0011, 1 = 0001)
```

```
And 5,3,var
```

ARCCOS**Sintassi**

ARCCOS **operando, risultato**

Argomenti

operando costante o variabile o nome di dispositivo
risultato variabile o nome di dispositivo

Descrizione

Esegue una operazione di arcocoseno su **operando** e pone il valore in **risultato**, in gradi. Il valore del risultato è dato nell'intervallo $0^\circ \div 180^\circ$. Per la conversione dei dati, in base al tipo di dato dichiarato, riferirsi al capitolo [Conversione dati](#).


```
SetVal 10,op ; assegna 10 alla variabile op
Logdec op,var
;Il valore posto nella variabile var sarà 1
```

MOD

Sintassi

MOD **operando1, operando2, risultato**

Argomenti

operando1 costante o variabile integer o nome di dispositivo
operando2 costante o variabile integer o nome di dispositivo
risultato variabile integer o nome di dispositivo

Descrizione

Esegue una operazione di modulo tra **operando1** e **operando2** e pone il risultato in **risultato**. Il modulo è il resto risultante dalla divisione tra il primo e il secondo operando. La istruzione può generare errore di sistema quando **operando2** è uguale a 0. Per la conversione dei dati, in base al tipo di dato dichiarato, riferirsi al capitolo [Conversione dati](#).

Esempio

```
SetVal 20,op1 ; assegna 20 alla variabile op1
SetVal 3,op2 ; assegna 3 alla variabile op2
Mod op1,op2,var
;Il valore posto nella variabile var sarà 2
```

MUL

Sintassi

MUL **operando1, operando2, risultato**

Argomenti

operando1 costante o variabile o nome di dispositivo
operando2 costante o variabile o nome di dispositivo
risultato variabile o nome di dispositivo

Descrizione

Esegue una operazione di moltiplicazione tra **operando1** e **operando2** e pone il risultato in **risultato**. Per la conversione dei dati, in base al tipo di dato dichiarato, riferirsi al capitolo [Conversione dati](#).

Esempio

```
SetVal 5,op1 ; assegna 5 alla variabile op1
SetVal 2,op2 ; assegna 2 alla variabile op2
Mul op1,op2,var
;Il valore posto nella variabile var sarà 10
```

NOT

Sintassi

NOT **operando**

Argomenti

operando variabile o nome di dispositivo

Descrizione

Esegue una operazione di NOT binario (*vengono invertiti i singoli bit*) del valore espresso da **operando**. Il risultato viene memorizzato in **operando**.

Esempio

```
SetVal 5,var ; assegna il valore 5 a "var"
Not var
```

```

; Il risultato sarà var = -6
; Notazione binaria: 5 = 0000 0101,
; Notazione binaria:10 = 0000 1010
; Notazione esadecimale 5 = 0000 0000 0000 0005
; Notazione esadecimale 10 = 0000 0000 0000 000A
; eseguendo un NOT su valore 5 si ottiene 0xFFFF FFFF FFFF FFFA = -6

```

OR

Sintassi

OR **operando1, operando2, risultato**

Argomenti

operando1	costante o variabile o nome di dispositivo
operando2	costante o variabile o nome di dispositivo
risultato	variabile o nome di dispositivo

Descrizione

Esegue una operazione di OR binario (tra due bit, risulta 1 se almeno uno dei due vale 1) tra **operando1** e **operando2** e pone il risultato in **risultato**. Per la conversione dei dati, in base al tipo di dato dichiarato, riferirsi al capitolo [Conversione dati](#).

Esempio

```

;Il valore posto nella variabile var sarà 7
;(Notazione binaria: 5 = 0101, 3 = 0011, 7 = 0111 )

```

Or 5,3,var

RANDOM

Sintassi

RANDOM **min, max, risultato**

Argomenti

min	costante o variabile
max	costante o variabile
risultato	variabile o nome di dispositivo

Descrizione

Restituisce in **risultato** un numero pseudocasuale compreso tra **min** e **max** (estremi inclusi). Eseguendo l'istruzione ripetutamente si ottiene una sequenza di numeri pseudocasuali. Per la conversione dei dati, in base al tipo di dato dichiarato, riferirsi al capitolo [Conversione dati](#).

Esempio

```

SetVal 2,op1 ; assegna 2 alla variabile op1
SetVal 100,op2 ; assegna 100 alla variabile op2
Random op1,op2,var ;Il valore posto nella variabile var
;sarà un numero casuale
;compreso tra 2 e 100

```

RESETBIT

Sintassi

RESETBIT **maschera, nbit**

Argomenti

maschera	costante o variabile intera o nomecounter o nomeporto. Rappresenta il valore da modificare (max 32 bit)
nbit	costante o variabile intera o nomecounter. Numero del bit da modificare

Descrizione

Imposta a 0 un singolo bit, specificato da **nbit**, della **maschera** di bit passata. L'argomento **maschera** deve poter corrispondere ad un valore intero con un massimo di 32 bit. Il numero di bit, **nbit**, va da 1 a 32.

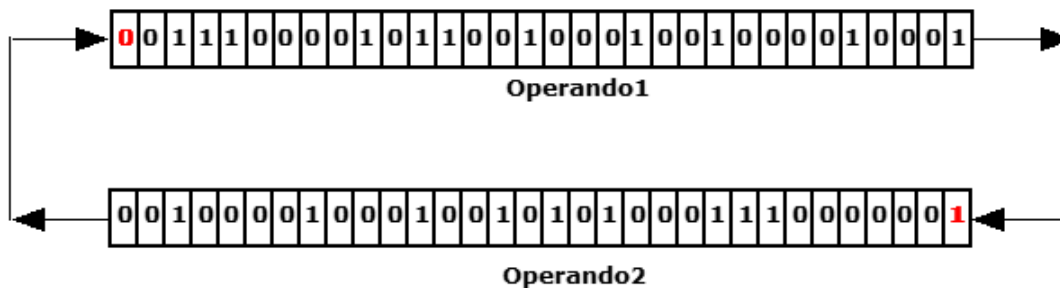
0 1 1 1 0 0 0 0 1 0 1 1 0 0 1 0 0 0 1 0 0 1 0 0 0 0 1 0 0 0 1 1

Operando1

0 0 0 1 0 0 0 0 1 0 0 0 1 0 0 1 0 1 0 1 0 0 0 1 1 1 0 0 0 0 0 0

Operando2

Dopo la rotazione



Scorrimento a destra di un char (shift a destra senza riporto)



0 1 1 1 0 0 0 0 1 0 1 1 0 0 1 0 0 0 1 0 0 1 0 0 0 0 1 0 0 0 1 0

Operando1 dopo
shift a destra

Scorrimento a destra di un integer (shift a destra senza riporto)

operando2 costante o variabile o nome di dispositivo
risultato variabile o nome di dispositivo

Descrizione

Esegue una operazione di sottrazione tra **operando1** e **operando2** e pone il risultato in **risultato**. Per la conversione dei dati, in base al tipo di dato dichiarato, riferirsi al capitolo [Conversione dati](#).

Esempio

```
SetVal 10,op1 ; assegna 10 alla variabile op1
SetVal 4,op2 ; assegna 4 alla variabile op2
Sub op1,op2,var
;Il valore posto nella variabile var sarà 6
```

TAN**Sintassi**

TAN **operando, risultato**

Argomenti

operando costante o variabile o nome di dispositivo
risultato variabile o nome di dispositivo

Descrizione

Esegue una operazione di tangente su **operando** e pone il valore in **risultato**. L'argomento **operando** è espresso in gradi. Per la conversione dei dati, in base al tipo di dato dichiarato, riferirsi al capitolo [Conversione dati](#).

Esempio

```
SetVal 45,op ;assegna 45 alla variabile op
Tan op,var
;Il valore posto nella variabile var sarà 1
```

TRUNC**Sintassi**

TRUNC **operando, risultato**

Argomenti

operando costante o variabile o nome di dispositivo
risultato variabile o nome di dispositivo

Descrizione

Tronca ad intero il valore espresso da **operando** e pone il valore in **risultato**. (La parte decimale viene persa). Per la conversione dei dati, in base al tipo di dato dichiarato, riferirsi al capitolo [Conversione dati](#).

Esempio

```
SetVal 5.7,op ;assegna 5.7 alla variabile op
Trunc op,var
;Il valore posto nella variabile var sarà 5
```

XOR**Sintassi**

XOR **operando1, operando2, risultato**

Argomenti

operando1 costante o variabile o nome di dispositivo
operando2 costante o variabile o nome di dispositivo
risultato variabile o nome di dispositivo

Descrizione

Esegue una operazione di XOR binario (tra due bit, risulta 1 se uno solo dei due vale 1) tra **operando1** e **operando2** e pone il risultato in **risultato**. Per la conversione dei dati, in base al tipo di dato dichiarato, riferirsi al capitolo [Conversione dati](#).

Esempio

```
Xor 5,3,var
;Il valore posto nella variabile var sarà 6
;(Notazione binaria: 5 = 0101, 3 = 0011, 6 = 0110)
```

10.3.11 Multitasking

ENDMAIL

Sintassi

ENDMAIL **mail**

Argomenti

mail costante o variabile di tipo integer. Numero del mailbox (1÷256)

Descrizione

Segnala la fine dell'esecuzione di un comando associato ad un messaggio prelevato dalla casella postale **mail**.

Il task che aveva inviato il messaggio (con l'istruzione [SENDMAIL](#)) ed era rimasto in attesa dell'esecuzione del comando (argomento di attesa WAITACK) può così proseguire la propria esecuzione. Questa istruzione **ha effetto solo se eseguita dal task** che ha precedentemente ricevuto il messaggio (con l'istruzione WAITMAIL o TESTMAIL).

Vedere anche le istruzioni [SENDMAIL](#), [WAITMAIL](#) e [TESTMAIL](#)

Esempio

[Server di movimentazione assi](#)

ENDREALTIMETASK

Sintassi

ENDREALTIMETASK **nomefunzione**

Argomenti

nomefunzione nome funzione

Descrizione

Termina l'esecuzione di un [task real-time](#). Vedere anche [STARTREALTIMETASK](#).

ENDTASK

Sintassi

ENDTASK [**nometask**]

Argomenti

nometask nome task

Descrizione

Termina l'esecuzione di un task e di tutti i task che questa ha attivato (task figli).

Questa istruzione sospende inoltre il movimento degli assi, annulla le RECEIVE pendenti e chiude le eventuali connessioni a porte seriali.

Se la variabile **nometask** è omessa, fa terminare l'esecuzione del task corrente.

GETPRIORITYLEVEL

Sintassi

GETPRIORITYLEVEL **livello [,nomefunzione]**

Argomenti

livello variabile. Livello di priorità di esecuzione
nomefunzione nome funzione

nomevar1[...nomevarN] - **WAITACK** attende l'esecuzione del comando costante o variabile integer. Nomi variabili 1÷20
matrice[riga] costante o variabile integer. Numero di riga della matrice

Descrizione

Invia un messaggio (o comando) alla casella **mail**. I messaggi possono essere utilizzati per sincronizzare e scambiare informazioni tra due o più task.

Se la casella **mail** non esiste, ovvero non è stata ancora eseguita un'istruzione [WAITMAIL](#) o [TESTMAIL](#), l'istruzione viene ignorata.

Se il task ricevente non è in attesa di un messaggio (istruzione [WAITMAIL](#)) oppure è impegnato, i dati (**nomevar** (1÷20) o la riga di matrice indicata da **matrice[riga]**) passati dall'istruzione vengono memorizzati in una coda. In questo caso:

1. se l'argomento di attesa è **NOWAIT** l'esecuzione continua con l'istruzione successiva;
2. se l'argomento di attesa è **WAIT** l'esecuzione attende che il messaggio venga letto dal task ricevente;
3. se l'argomento di attesa è **WAITACK** l'esecuzione attende che il messaggio venga letto e che venga confermata l'esecuzione del comando dal task ricevente (attraverso l'istruzione [ENDMAIL](#) o una nuova [WAITMAIL](#)).

È molto importante che il numero delle variabili passate ed i loro tipi coincidano con quelli utilizzati per creare la casella postale con l'istruzione [WAITMAIL](#). Il controllo non consente l'utilizzo di tipi diversi e non effettua conversioni di tipo automatici (cast) come avviene di solito.

Una SENDMAIL senza i parametri opzionali (dati) si riduce ad un semplice meccanismo di sincronizzazione tra task.

Esempio

[Server di movimentazione assi](#)

SETPRIORITYLEVEL

Sintassi

SETPRIORITYLEVEL **livello [, nomefunzione]**

Argomenti

livello costante o variabile. Livello di priorità di esecuzione
nomefunzione nome funzione

Descrizione

Assegna al task definito in **nomefunzione** il valore di priorità contenuto nella variabile **livello**. Detto valore è un numero compreso tra 0 e 255, dove 0 rappresenta il livello più alto di priorità e 255 il più basso. Se non è definito il nome del task nella variabile **nomefunzione** viene modificato il valore di priorità del task corrente, ovvero il livello di esecuzione della funzione in cui l'istruzione viene eseguita. Vedere anche [GETPRIORITYLEVEL](#).

STARTREALTIMETASK

Sintassi

STARTREALTIMETASK **nomefunzione**

Argomenti

nomefunzione nome funzione

Descrizione

Attiva l'esecuzione di un [task real-time](#). Tale task viene eseguito con la stessa frequenza del real-time di controllo degli assi. A differenza dei normali task GPL viene eseguito interamente ogni real-time, dalla prima istruzione della function fino alla prima istruzione FRET. Vedere anche [ENDREALTIMETASK](#).

Nota:

Le variabili locali dichiarate nel task real-time vengono inizializzate solo in avvio del task e in seguito mantengono il valore dell'ultima esecuzione.

Esempio[Esecuzione Sequenziale / Parallela](#)**10.3.12 Gestione di Flusso****CALL****Sintassi****CALL** **nomesottoprogramma****Argomenti****nomesottoprogramma** nome sottoprogramma, etichetta**Descrizione**

Esegue il sottoprogramma definito dall'etichetta **nomesottoprogramma**. Ogni sottoprogramma, per poter tornare all'istruzione successiva alla CALL, deve terminare, nel punto di uscita, con l'istruzione [RET](#).

Nota

Questa istruzione, assieme alla RET, è una tipica fonte di errori di programmazione. Si consiglia di prestare attenzione al suo utilizzo, in particolare si consiglia di posizionare le sottoprocedure al termine del corpo di una funzione (dopo l'istruzione FRET) in modo da evitare l'esecuzione accidentale del codice della sottoprocedura come se fosse parte integrante del codice principale. Il risultato di questa situazione è nella migliore delle ipotesi un errore di sistema, in altri casi si ottengono comportamenti anomali della macchina la cui causa è di difficile individuazione.

DELONFLAG**Sintassi****DELONFLAG** **nomeflag****Argomenti****nomeflag** nome di dispositivo flag**Descrizione**

Disabilita la gestione dell'interruzione software sullo stato di un flag bit o di un flag switch che era stata precedentemente abilitata con [ONFLAG](#).

DELONINPUT**Sintassi****DELONINPUT** **nomeinput****Argomenti****nomeinput** nome ingresso**Descrizione**

Disabilita la gestione dell'interruzione software sullo stato di un ingresso precedentemente abilitata con l'istruzione [ONINPUT](#).

FCALL**Sintassi****[FCALL]** **nomefunzione** **nomefunzione [, parametri]**
[parametri]**Argomenti****nomefunzione** nome della funzione da richiamare
parametri eventuali parametri passati alla funzione**Descrizione**

Effettua una chiamata a funzione, ovvero viene eseguita la funzione **nomefunzione**. Alla funzione possono essere passati eventuali **parametri**. Questi devono corrispondere per numero e tipo a quelli dichiarati nella funzione chiamata. L'esecuzione della funzione chiamante (quella in cui viene eseguita la FCALL) riparte al termine dell'esecuzione della funzione chiamata (quella specificata dal parametro **nomefunzione**).

Notare la differenza con l'istruzione [STARTTASK](#) che manda in esecuzione un'altra funzione in parallelo a quella chiamante (utilizzata per avere più task in esecuzione contemporanea).

Esempio

[Esecuzione Sequenziale / Parallela](#)

FOR/NEXT

Sintassi

```
FOR           indice, inizio, fine [, step]
             istruzione
             istruzione
             ...
NEXT
```

Argomenti

indice	variabile o nomecounter
inizio	costante o variabile o nomecounter. Valore iniziale
fine	costante o variabile o nomecounter. Valore finale
step	costante o variabile o nomecounter. Passo di incremento o decremento

Descrizione

Ripete ciclicamente l'esecuzione delle istruzioni racchiuse tra l'istruzione FOR e l'istruzione NEXT. Durante il primo ciclo la variabile **indice** viene inizializzata al valore della variabile **inizio**. Al secondo ciclo la variabile **indice** avrà valore uguale a (**inizio+step**), e così via finché la variabile **indice** non diventa maggiore, (o minore nel caso di valore negativo della variabile **step**), della variabile **fine**. Se la variabile **step** è omessa, viene assunto un valore di default pari a +1. Le istruzioni racchiuse tra FOR e NEXT possono modificare il numero delle ripetizioni modificando **indice**. Quando le ripetizioni sono terminate viene eseguita l'istruzione successiva a NEXT.

Esempio

```
Function Loop
local  i As integer
local  vettore[10] as integer

For    i,1,10
  Setval i, vettore[i]      ; riempie gli elementi del vettore
                                ; con i numeri 1,2, .... 10
Next
Fret

Function loop2
local  j As integer
local  vettore[10] as integer

For    j,1,10,2
  Setval 27, vettore[j]    ; inserisce il valore 27 nei seguenti
                                ; elementi del vettore: 1,3,5,7,9
Next
Fret
```

FRET

Sintassi

```
FRET
```

Argomenti

nessun argomento

Descrizione

Ritorno da una funzione. Causa la terminazione dell'esecuzione della funzione ed il rilascio della memoria allocata per le variabili locali. Se la funzione era stata mandata in esecuzione con una FCALL l'esecuzione della funzione chiamante riprende dall'istruzione successiva.

Se in precedenza erano state eseguite delle WAITTASK con la funzione corrente (quella in cui viene eseguita la FRET) come argomento, i task in attesa vengono sbloccati.

GOTO

Sintassi

GOTO **etichetta**

Argomenti

etichetta etichetta

Descrizione

Effettua un salto incondizionato all'etichetta specificata dal parametro **etichetta**.

Un' etichetta viene definita da una parola chiave seguita immediatamente dal carattere ":".

L'etichetta deve trovarsi all'interno del corpo della funzione in cui viene eseguita l'istruzione GOTO.

Nota

Il corpo di una funzione è compreso tra l'istruzione FUNCTION che dichiara il nome della funzione e l'analoga istruzione che definisce la funzione successiva (oppure dalla fine del file). Da questo si deduce la possibilità di eseguire salti dal corpo principale della funzione ad eventuali sottoprocedure (vedere istruzioni [CALL](#) e [RET](#)). Si sconsiglia vivamente questo stile di programmazione essendo fonte di numerosi errori di difficile individuazione.

Esempio

```
; Function che fa lampeggiare un flag
; (ad es. una spia di allarme su un quadro sinottico)
```

Function Loop

```
loop:
setflag    allarme    ; attiva il flag
delay      1
resetflag  allarme    ; disattiva il flag
delay      1
goto       loop
Fret
```

IF/IFVALUE/IF-THEN-ELSE

Sintassi

IF **nomevar, operatore di confronto, valore, GOTO etichetta**
IF **nomevar, operatore di confronto, valore, CALL nomesottoprogramma**
IF **nomevar, operatore di confronto, valore, nomefunzione**

IF **nomevar, operatore di confronto, valore THEN**

istruzione

istruzione

...

ENDIF

IF **nomevar, operatore di confronto, valore THEN**

istruzione

istruzione

...

ELSE

istruzione

istruzione

...

ENDIF

Argomenti

nomevar costante o variabile o nomedevice

operatore di confronto i simboli che possono essere utilizzati per eseguire il confronto sono:

confronto < (minore) = (uguale)

operando2	costante o variabile o nome device
valoreverifica	costante. Valore utilizzato per la verifica del risultato dell'operazione. Può assumere i seguenti valori: TRUE 1, FALSE 0
etichetta	nome dell'etichetta a cui saltare
nomesottoprogramma	nome del sottoprogramma
nomefunzione	nome della funzione

Descrizione

Vengono effettuati due confronti, il primo tra **operando1** e **operando2**, il secondo tra il risultato del primo confronto e **valoreverifica**.

Il primo confronto consiste in un AND binario tra **operando1** e **operando2**. I due operandi vengono considerati come maschere di bit. Se il risultato dell'AND binario ha almeno un bit diverso da 0, il risultato del primo confronto è TRUE. Questo viene quindi confrontato con **valoreverifica**. Se i due valori coincidono viene effettuato il salto ad etichetta o la chiamata a funzione o sottoprogramma.

Per maggiori dettagli vedere il costrutto [IF-THEN-ELSE](#).

IFBIT**Sintassi**

IFBIT		maschera, nbit, stato, GOTO etichetta
IFBIT		maschera, nbit, stato, CALL nomesottoprogramma
IFBIT		maschera, nbit, stato, nomefunzione
IFBIT		maschera, nbit, stato THEN
	istruzione	
	istruzione	
	...	
ENDIF		
IFBIT		maschera, nbit, stato THEN
	istruzione	
	istruzione	
	...	
ELSE		
	istruzione	
	istruzione	
	...	
ENDIF		

Argomenti

maschera	costante o variabile intera o nomecounter o nomeporto. Valore da verificare
nbit	costante o variabile intera o nomecounter. Numero del bit (1÷32)
stato	costante predefinita. Stato da verificare su maschera. I valori ammessi sono: ON bit scelto a 1 OFF bit scelto a 0
etichetta	etichetta di salto (GOTO)
nomesottoprogramma	chiamata sottoprogramma (CALL)
ma	
nomefunzione	nome della funzione

Descrizione

Test su un singolo bit della **maschera** di bit passata. L'argomento **maschera** deve poter corrispondere ad un valore intero con un massimo di 32 bit. Il numero da assegnare alla variabile **nbit** per identificare il bit da verificare va da 1 a 32. Se la condizione indicata in **stato** è verificata, salta a **etichetta** o chiama **nomesottoprogramma** o **nomefunzione**.

Per maggiori dettagli vedere il costrutto [IF-THEN-ELSE](#).

IFBLACKBOX**Sintassi**

IFBLACKBOX	GOTO etichetta
IFBLACKBOX	CALL nomesottoprogramma
IFBLACKBOX	nomefunzione

Argomenti

etichetta	nome dell'etichetta a cui saltare
nomesottoprogramma	nome del sottoprogramma
nomefunzione	nome della funzione

Descrizione

Se la registrazione è attiva, salta a **etichetta** o chiama **nomesottoprogramma** o **nomefunzione**. Vedere anche [STARTBLACKBOX](#), [PAUSEBLACKBOX](#) e [ENDBLACKBOX](#).

IFCHANGEVEL**Sintassi**

IFCHANGEVEL **asse** [, **stato**], **GOTO** **etichetta**
IFCHANGEVEL **asse** [, **stato**], **CALL** **nomesottoprogramma**
IFCHANGEVEL **asse** [, **stato**], **nomefunzione**

Argomenti

asse nome del dispositivo tipo asse
stato tipo di variazione. I valori ammessi sono: **POSITIVE**, **NEGATIVE**
etichetta nome dell'etichetta a cui saltare
nomesottoprogramma nome del sottoprogramma
nomefunzione nome della funzione

Descrizione

Testa se si è verificata una variazione di velocità di un asse.

Se l'asse definito dalla variabile **asse** sta effettuando un cambio di velocità durante uno spostamento, viene effettuato un salto a **etichetta** o una chiamata a **nomesottoprogramma** o **nomefunzione**.

Il parametro **stato** specifica se la velocità è aumentata (**POSITIVE**) o diminuita (**NEGATIVE**).

IFCOUNTER**Sintassi**

IFCOUNTER **nomecounter**, **operatore di confronto**, **valore**, **GOTO** **etichetta**
IFCOUNTER **nomecounter**, **operatore di confronto**, **valore**, **CALL**
 nomesottoprogramma
IFCOUNTER **nomecounter**, **operatore di confronto**, **valore**, **nomefunzione**

IFCOUNTER **nomevar**, **operatore di confronto**, **valore** **THEN**
 istruzione
 istruzione
 ...
ENDIF

IFCOUNTER **nomevar**, **operatore di confronto**, **valore** **THEN**
 istruzione
 istruzione
 ...
ELSE **istruzione**
 istruzione
 ...
ENDIF

Argomenti

nomecounter nome del contatore

operatore di confronto i simboli che possono essere utilizzati per eseguire il confronto sono:
< (minore) = (uguale)
> (maggiore) =< (minore uguale)
>= (maggiore uguale) <> (diverso)

valore costante o variabile o **nomecounter**

etichetta nome dell'etichetta a cui saltare

nomesottoprogramma nome del sottoprogramma

mma

nomefunzione nome della funzione

Descrizione

L'istruzione esegue il test di un contatore.

Se il contenuto del contatore definito nella variabile **nomecounter** verifica la condizione specificata dall'**operatore di confronto** con il valore espresso dalla variabile **valore**, viene effettuato un salto all'etichetta specificata da **etichetta** o una chiamata al sottoprogramma definito da **nomesottoprogramma** o alla funzione definita da **nomefunzione**.

Per maggiori dettagli vedere il costrutto [IF-THEN-ELSE](#).

IFDEC**Sintassi**

IFDEC	asse, GOTO etichetta
IFDEC	asse, CALL nomesottoprogramma
IFDEC	asse, nomefunzione

Argomenti

asse	nome del dispositivo tipo asse
etichetta	nome dell'etichetta a cui saltare
nomesottoprogramma	nome del sottoprogramma
nomefunzione	nome della funzione

Descrizione

Testa se l'asse definito dalla variabile **asse** è nello stato di decelerazione.
Se la condizione è verificata, salta a **etichetta** o chiama **nomesottoprogramma** o **nomefunzione**.

IFDIR**Sintassi**

IFDIR	asse, direzione, GOTO etichetta
IFDIR	asse, direzione, CALL nomesottoprogramma
IFDIR	asse, direzione, nomefunzione

IFDIR	asse, direzione THEN
--------------	-----------------------------

istruzione
istruzione
...

ENDIF

IFDIR	asse, direzione THEN
--------------	-----------------------------

istruzione
istruzione
...

ELSE

istruzione
istruzione
...

ENDIF**Argomenti**

asse	nome di dispositivo tipo asse
direzione	direzione asse. I valori ammessi sono: POSITIVE direzione asse positiva NEGATIVE direzione asse negativa
etichetta	nome dell'etichetta a cui saltare
nomesottoprogramma	nome di sottoprogramma
nomefunzione	nome di funzione

Descrizione

Test sulla direzione corrente di un asse.
Se l'**asse** si muove nella direzione specificata dalla variabile **direzione**, viene effettuato un salto a **etichetta** o una chiamata a **nomesottoprogramma** o **nomefunzione**.
Per maggiori dettagli vedere il costrutto [IF-THEN-ELSE](#).

IFERRAN**Sintassi**

IFERRAN	asse, operatore di confronto, valore, GOTO etichetta
IFERRAN	asse, operatore di confronto, valore, CALL nomesottoprogramma
IFERRAN	asse, operatore di confronto, valore, nomefunzione

IFERRAN	asse, operatore di confronto, valore THEN
----------------	--

istruzione
istruzione
...

ENDIF

IFERRAN **asse, operatore di confronto, valore THEN**
 istruzione
 istruzione
 ...
ELSE **istruzione**
 istruzione
 ...
ENDIF

Argomenti

asse nome del dispositivo tipo asse
operatore di confronto i simboli che possono essere utilizzati per eseguire il confronto sono:
 < (minore) = (uguale)
 > (maggiore) =< (minore uguale)
 >= (maggiore uguale) <> (diverso)
valore costante o variabile o nomecounter
etichetta nome dell'etichetta a cui saltare
nomesottoprogramma nome del sottoprogramma
nomefunzione nome della funzione

Descrizione

Test sul valore dell'errore d'inseguimento (errore d'anello) dell'asse definito nella variabile **asse**.
 Se l'errore d'anello dell'**asse** verifica la condizione espressa dall'**operatore di confronto** con il valore espresso da **valore**, viene effettuato un salto a **etichetta** o una chiamata a **nomesottoprogramma** o a **nomefunzione**.
 Per maggiori dettagli vedere il costrutto [IF-THEN-ELSE](#).

IFERROR**Sintassi**

IFERROR **numero, IDposiz, GOTO etichetta**
IFERROR **numero, IDposiz, CALL etichetta**
IFERROR **numero, IDposiz, nomefunzione**
IFERROR **nomedevice, stato, IDposiz, GOTO etichetta**
IFERROR **nomedevice, stato, IDposiz, CALL etichetta**
IFERROR **nomedevice, stato, IDposiz, nomefunzione**

Argomenti

numero DEFMSG o costante o variabile integer
nomedevice nome del dispositivo
stato costante predefinita. Può assumere i seguenti valori: **ON, OFF**
IDposiz costante o variabile. Valore numerico utilizzato nei sinottici.
etichetta nome dell'etichetta a cui saltare
nomefunzione nome della funzione

Descrizione

Test sull'errore di ciclo attivo.
 Se l'errore di ciclo, identificato da **numero** e **IDposiz** o da **nomedevice, stato** e **IDposiz** è attivo, viene effettuato un salto ad **etichetta** o una chiamata alla funzione **nomefunzione**,
 Il parametro **numero** può identificare un errore di ciclo di modulo (quindi un valore numerico intero) o di gruppo (in questo caso si usa una DEFMSG).
 Il parametro **nomedevice** è il nome di un dispositivo e il parametro **stato** rappresenta lo stato ON/OFF in cui deve trovarsi il dispositivo al momento della generazione dell'errore.
 Il parametro **IDposiz** è un parametro opzionale che specifica il valore numerico utilizzato nei sinottici per smistare errori di ciclo in caselle differenti. Deve corrispondere al valore specificato nel costruttore sinottici per quella particolare casella di visualizzazione. Se non serve indicare una casella specifica va assegnata la costante predefinita NOPLACE. Il range di valori impostabili è compreso tra 0 (NOPLACE) e 1023.
 Se l'istruzione viene utilizzata senza che la gestione degli allarmi a stati sia abilitata viene generato un errore di sistema.
 Vedere anche l'istruzione [ERROR](#).

IFFLAG**Sintassi**

IFFLAG **nomeflag, stato, GOTO etichetta**
IFFLAG **nomeflag, stato, CALL nomesottoprogramma**
IFFLAG **nomeflag, stato, nomefunzione**

IFFLAG **nomeflag, stato THEN**

```

        istruzione
        istruzione
        ...
ENDIF

IFFLAG          nomeflag, stato THEN
        istruzione
        istruzione
        ...
ELSE
        istruzione
        istruzione
        ...
ENDIF

```

Argomenti

nomeflag nome di dispositivo flag
stato costante predefinita. Stato da verificare. I valori ammessi sono:
 ON attivo
 OFF disattivo

etichetta nome dell'etichetta a cui saltare
nomesottoprogramm nome del sottoprogramma
a
nomefunzione nome della funzione

Descrizione

Test dello stato logico di un flag.

Se il flag definito dalla variabile **nomeflag** si trova nello **stato** indicato, viene effettuato un salto a **etichetta** o una chiamata a **nomesottoprogramma** o a **nomefunzione**.

Per maggiori dettagli vedere il costrutto [IF-THEN-ELSE](#).

IFINPUT**Sintassi**

```

IFINPUT          nomeinput, stato, GOTO etichetta
IFINPUT          nomeinput, stato, CALL nomesottoprogramma
IFINPUT          nomeinput, stato, nomefunzione

IFINPUT          nomeinput, stato THEN
        istruzione
        istruzione
        ...
ENDIF

IFINPUT          nomeinput, stato THEN
        istruzione
        istruzione
        ...
ELSE
        istruzione
        istruzione
        ...
ENDIF

```

Argomenti

nomeinput nome ingresso
stato costante predefinita. Stato da verificare
 I valori ammessi sono:
 - **ON** attivo
 - **OFF** disattivo

etichetta nome dell'etichetta a cui saltare
nomesottoprogramma nome del sottoprogramma
nomefunzione nome della funzione

Descrizione

Test dello stato logico di un ingresso.

Se l'ingresso definito nella variabile **nomeinput** si trova nello **stato** indicato, viene effettuato un salto a **etichetta** o una chiamata a **nomesottoprogramma** o a **nomefunzione**.

Per maggiori dettagli vedere il costrutto [IF-THEN-ELSE](#).

IFOUTPUT

Sintassi

IFOUTPUT **nomeoutput, stato, GOTO etichetta**
IFOUTPUT **nomeoutput, stato, CALL nomesottoprogramma**
IFOUTPUT **nomeoutput, stato, nomefunzione**

IFOUTPUT **nomeoutput, stato THEN**
 istruzione
 istruzione

...
ENDIF

IFOUTPUT **nomeoutput, stato THEN**
 istruzione
 istruzione

...
ELSE

istruzione
 istruzione

...
ENDIF

Argomenti

nomeoutput nome uscita
stato costante predefinita. Rappresenta lo stato da verificare sull' uscita.
 I valori ammessi sono:
 ON attivo
 OFF disattivo
etichetta nome dell'etichetta a cui saltare
nomesottoprogramma nome del sottoprogramma
mma
nomefunzione nome della funzione

Descrizione

Test dello stato logico di una uscita.

Se l'uscita definita nella variabile **nomeoutput** si trova nello **stato** indicato, viene effettuato un salto a **etichetta** o una chiamata a **nomesottoprogramma** o a **nomefunzione**.

Per maggiori dettagli vedere il costrutto [IF-THEN-ELSE](#).

IFQUOTER

Sintassi

IFQUOTER **asse, operatore di confronto, valore, GOTO etichetta**
IFQUOTER **asse, operatore di confronto, valore, CALL nomesottoprogramma**
IFQUOTER **asse, operatore di confronto, valore, nomefunzione**

IFQUOTER **asse, operatore di confronto, valore THEN**
 istruzione
 istruzione

...
ENDIF

IFQUOTER **asse, operatore di confronto, valore THEN**
 istruzione
 istruzione

...
ELSE

istruzione
 istruzione

...
ENDIF

Argomenti

asse nome del dispositivo tipo asse
operatore di confronto i simboli che possono essere utilizzati per eseguire il confronto sono:
 < (minore) = (uguale)
 > (maggiore) =< (minore uguale)
 >= (maggiore uguale) <> (diverso)

valore	costante o variabile o nomecounter
etichetta	nome dell'etichetta a cui saltare
nomesottoprogramma	nome del sottoprogramma
nomefunzione	nome della funzione

Descrizione

Test su quota reale espressa dalla variabile **asse**.

Se il valore della variabile **asse** verifica la condizione espressa dall'**operatore di confronto** con il valore espresso da **valore**, viene effettuato un salto a **etichetta** o una chiamata a **nomesottoprogramma** o a **nomefunzione**.

Per maggiori dettagli vedere il costrutto [IF-THEN-ELSE](#).

IFQUOTET**Sintassi**

IFQUOTET **asse, operatore di confronto, valore, GOTO etichetta**
IFQUOTET **asse, operatore di confronto, valore, CALL nomesottoprogramma**
IFQUOTET **asse, operatore di confronto, valore, nomefunzione**

IFQUOTET **asse, operatore di confronto, valore THEN**

istruzione
istruzione

...

ENDIF

IFQUOTET **asse, operatore di confronto, valore THEN**

istruzione
istruzione

...

ELSE

istruzione
istruzione

...

ENDIF

Argomenti

asse	nome del dispositivo tipo asse
operatore di confronto	i simboli che possono essere utilizzati per eseguire il confronto sono: < (minore) = (uguale) > (maggiore) =< (minore uguale) >= (maggiore uguale) <> (diverso)
valore	costante o variabile o nomecounter
etichetta	nome dell'etichetta a cui saltare
nomesottoprogramma	nome del sottoprogramma
nomefunzione	nome della funzione

Descrizione

Test su quota teorica espressa dalla variabile **asse**.

Se il valore della variabile **asse** verifica la condizione espressa dall'**operatore di confronto** con il valore espresso da **valore**, viene effettuato un salto a **etichetta** o una chiamata a **nomesottoprogramma** o a **nomefunzione**.

Per maggiori dettagli vedere il costrutto [IF-THEN-ELSE](#).

IFRECEIVED**Sintassi**

IFRECEIVED **[sorgente,] identificatore, GOTO etichetta**
IFRECEIVED **[sorgente,] identificatore, CALL nomesottoprogramma**
IFRECEIVED **[sorgente,] identificatore, nomefunzione**

Argomenti

sorgente	costante di tipo stringa
identificatore	costante di tipo stringa
etichetta	nome dell'etichetta a cui saltare
nomesottoprogramma	nome del sottoprogramma
nomefunzione	nome della funzione

Descrizione

Verifica se un'istruzione [RECEIVE](#) è stata soddisfatta.

Se una particolare RECEIVE precedente è stata soddisfatta, viene effettuato un salto a **etichetta** o una chiamata alla **nomesottoprogramma** o a **nomefunzione**.
Vedere anche istruzioni [RECEIVE](#), [WAITRECEIVE](#), [SEND](#).

IFREG

Sintassi

IFREG **asse, GOTO etichetta**
IFREG **asse, CALL nomesottoprogramma**
IFREG **asse, nomefunzione**

Argomenti

asse nome del dispositivo tipo asse
etichetta nome dell'etichetta a cui saltare
nomesottoprogramma nome del sottoprogramma
nomefunzione nome della funzione

Descrizione

Testa se l'asse definito dalla variabile **asse** è nello stato di regime.
Se la condizione è verificata, salta a **etichetta** o chiama **nomesottoprogramma** o **nomefunzione**.

IFSAME

Sintassi

IFSAME **operando1, operando2, GOTO etichetta**
IFSAME **operando1, operando2, CALL nomesottoprogramma**
IFSAME **operando1, operando2, nomefunzione**

Argomenti

operando1 variabile o nomedevice
operando2 variabile o nomedevice
etichetta nome dell'etichetta a cui saltare
nomesottoprogramma nome del sottoprogramma
nomefunzione nome della funzione

Descrizione

Test tra due operandi.
Verifica che il valore definito in **operando1** e **operando2** facciano riferimento o allo stesso dispositivo o alla stessa area di memoria.
Se il test tra i due operandi è verificato, viene effettuato un salto a **etichetta** o chiama **nomesottoprogramma** o **nomefunzione**.

IFSTILL

Sintassi

IFSTILL **asse, GOTO etichetta**
IFSTILL **asse, CALL nomesottoprogramma**
IFSTILL **asse, nomefunzione**

Argomenti

asse nome del dispositivo tipo asse
etichetta nome dell'etichetta a cui saltare
nomesottoprogramma nome del sottoprogramma
nomefunzione nome della funzione

Descrizione

Test se l'asse definito dalla variabile **asse** è fermo ovvero in "quota".
Se la condizione è verificata, salta a **etichetta** o chiama **nomesottoprogramma** o **nomefunzione**.
Vedere anche [IFTARGET](#) e [IFWIN](#).

IFSTR

Sintassi

IFSTR **stringa1, operatore di confronto, stringa2, GOTO etichetta**
IFSTR **stringa1, operatore di confronto, stringa2, CALL**
 nomesottoprogramma
IFSTR **stringa1, operatore di confronto, stringa2, nomefunzione**

IFSTR **stringa1, operatore di confronto, stringa2 THEN**
 istruzione

```

        istruzione
        ...
ENDIF

IFSTR          stringa1, operatore di confronto, stringa2 THEN
        istruzione
        istruzione
        ...
ELSE
        istruzione
        istruzione
        ...
ENDIF

```

Argomenti

stringa1	variabile string. È la prima stringa ASCII
operatore di confronto	i simboli che possono essere utilizzati per eseguire il confronto tra stringhe sono: < (minore) = (uguale) > (maggiore) =< (minore uguale) >= (maggiore uguale) <> (diverso)
stringa2	variabile string. È la seconda stringa ASCII
etichetta	nome dell'etichetta a cui saltare
nomesottoprogramma	nome del sottoprogramma
a	
nomefunzione	nome della funzione

Descrizione

Test su stringhe ASCII.

Se la stringa definita in **stringa1** verifica la condizione espressa dall'**operatore di confronto** con la stringa contenuta in **stringa2**, viene effettuato un salto a **etichetta** o una chiamata a **nomesottoprogramma** o a **nomefunzione**.

Per maggiori dettagli vedere il costrutto [IF-THEN-ELSE](#).

IFTARGET**Sintassi**

```

IFTARGET          asse, GOTO etichetta
IFTARGET          asse, CALL nomesottoprogramma
IFTARGET          asse, nomefunzione

```

Argomenti

asse	nome del dispositivo tipo asse
etichetta	nome dell'etichetta a cui saltare
nomesottoprogramma	nome del sottoprogramma
nomefunzione	nome della funzione

Descrizione

Testa se l'asse definito dalla variabile **asse** ha raggiunto la quota finale programmata. L'asse, anche se ha raggiunto la quota target finale, non è necessariamente fermo, in genere dovrà recuperare l'errore d'anello.

Se la condizione è verificata, salta a **etichetta** o chiama **nomesottoprogramma** o **nomefunzione**.

Vedere anche [IFSTILL](#) e [IFWIN](#).

IFTASKHOLD**Sintassi**

```

IFTASKHOLD        nometask, GOTO etichetta
IFTASKHOLD        nometask, CALL etichetta
IFTASKHOLD        nometask, nomefunzione

```

Argomenti

nometask	nome task parallelo
etichetta	nome dell'etichetta a cui saltare
nomesottoprogramma	nome del sottoprogramma
nomefunzione	nome della funzione

Descrizione

Verifica se il task è sospeso (stato hold).

Se il task **nometask** è sospeso, viene effettuato un salto a **etichetta** o una chiamata a **nomesottoprogramma** o a **nomefunzione**.

IFTASKRUN

Sintassi

IFTASKRUN	nometask, GOTO etichetta
IFTASKRUN	nometask, CALL nomesottoprogramma
IFTASKRUN	nometask, nomefunzione

Argomenti

nometask	nome task parallelo
etichetta	nome dell'etichetta a cui saltare
nomesottoprogramma	nome del sottoprogramma
nomefunzione	nome della funzione

Descrizione

Verifica se il task è in esecuzione.

Se il task definito in **nometask** è in esecuzione, viene effettuato un salto a **etichetta** o una chiamata a **nomesottoprogramma** o a **nomefunzione**.

IFTIMER

Sintassi

IFTIMER	nometimer, operatore di confronto, valore, GOTO etichetta
IFTIMER	nometimer, operatore di confronto, valore, CALL
IFTIMER	nomesottoprogramma
IFTIMER	nometimer, operatore di confronto, valore, nomefunzione

IFTIMER	nometimer, operatore di confronto, valore THEN
---------	---

istruzione
istruzione
...

ENDIF

IFTIMER	nometimer, operatore di confronto, valore THEN
---------	---

istruzione
istruzione
...

ELSE

istruzione
istruzione
...

ENDIF

Argomenti

nometimer	nome di dispositivo timer
operatore di confronto	i simboli che possono essere utilizzati per eseguire il confronto tra stringhe sono: < (minore) = (uguale) > (maggiore) =< (minore uguale) >= (maggiore uguale) <> (diverso)
valore	costante o variabile o nometimer . È il valore su cui eseguire il confronto
etichetta	nome dell'etichetta a cui saltare
nomesottoprogramma	nome del sottoprogramma
a	
nomefunzione	nome della funzione

Descrizione

Test di un timer.

Se il contenuto del timer **nometimer** verifica la condizione espressa dall'**operatore di confronto** con il valore espresso da **valore**, viene effettuato un salto a **etichetta** o una chiamata a **nomesottoprogramma** o a **nomefunzione**.

Per maggiori dettagli vedere il costrutto [IF-THEN-ELSE](#).

IFVEL

Sintassi

IFVEL	asse, operatore di confronto, valore, GOTO etichetta
IFVEL	asse, operatore di confronto, valore, CALL nomesottoprogramma
IFVEL	asse, operatore di confronto, valore, nomefunzione


```

IFVEL           asse, operatore di confronto, valore THEN
                   istruzione
                   istruzione
                   ...
ENDIF

IFVEL           asse, operatore di confronto, valore THEN
                   istruzione
                   istruzione
                   ...
ELSE
                   istruzione
                   istruzione
                   ...
ENDIF

```

Argomenti

asse	nome del dispositivo tipo asse
operatore di confronto	i simboli che possono essere utilizzati per eseguire il confronto sono: < (minore) = (uguale) > (maggiore) =< (minore uguale) >= (maggiore uguale) <> (diverso)
etichetta	nome dell'etichetta a cui saltare
nomesottoprogramma	nome del sottoprogramma
ma	
nomefunzione	nome della funzione

Descrizione

Test sulla velocità corrente di un asse.

Se la velocità dell'asse verifica la condizione espressa dall'**operatore di confronto** con il valore espresso da **valore**, viene effettuato un salto a **etichetta** o una chiamata a **nomesottoprogramma** o a **nomefunzione**.

Per maggiori dettagli vedere il costrutto [IF-THEN-ELSE](#).

IFWIN

Sintassi

```

IFWIN           asse, GOTO etichetta
IFWIN           asse, CALL nomesottoprogramma
IFWIN           asse, nomefunzione

```

Argomenti

asse	nome del dispositivo tipo asse
etichetta	nome dell'etichetta a cui saltare
nomesottoprogramma	nome del sottoprogramma
nomefunzione	nome della funzione

Descrizione

Testa se l'asse definito dalla variabile **asse** è entrato all'interno della finestra di arrivo in quota (vedi [Convenzioni e terminologia](#)).

Se la condizione è verificata, salta a **etichetta** o chiama **nomesottoprogramma** o **nomefunzione**.

Vedere anche [IFTARGET](#) e [IFSTILL](#).

IFXOR

Sintassi

```

IFXOR           operando1, operando2, valoreverifica, GOTO etichetta
IFXOR           operando1, operando2, valoreverifica, CALL nomesottoprogramma
IFXOR           operando1, operando2, valoreverifica, nomefunzione

IFXOR           operando1, operando2, valoreverifica THEN
                   istruzione
                   istruzione
                   ...
ENDIF

IFXOR           operando1, operando2, valoreverifica THEN

```

```

        istruzione
        istruzione
        ...
ELSE
        istruzione
        istruzione
        ...
ENDIF

```

Argomenti

operando1	costante o variabile o nome device
operando2	costante o variabile o nome device
valoreverifica	costante. È il valore utilizzato per la verifica del risultato dell'operazione. Può assumere i seguenti valori: TRUE 1, FALSE 0
etichetta	nome dell'etichetta a cui saltare
nome sottoprogramma	nome del sottoprogramma
nome funzione	nome della funzione

Descrizione

Vengono effettuati due confronti, il primo tra **operando1** e **operando2**, il secondo tra il risultato del primo confronto e **valoreverifica**.

Il primo confronto consiste in un XOR binario tra **operando1** e **operando2**. I due operandi vengono considerati come maschere di bit. Se il risultato dell'XOR binario ha almeno un bit diverso da 0, il risultato del primo confronto è TRUE. Questo viene quindi confrontato con **valoreverifica**. Se i due valori coincidono viene effettuato il salto ad etichetta o la chiamata a funzione o sottoprogramma.

Per maggiori dettagli vedere il costrutto [IF-THEN-ELSE](#).

ONERRSYS**Sintassi**

```
ONERRSYS          nomefunzione
```

Argomenti

nomefunzione	nome funzione
---------------------	---------------

Descrizione

Abilita la gestione degli errori di sistema. Il comportamento normale del controllo, quando si verifica un errore di sistema, è di terminare tutti i task. La gestione degli errori di sistema permette di evitare la terminazione dei task per i quali è stata abilitata.

Quando si verifica un errore di sistema viene mandata in esecuzione la funzione **nomefunzione**. Compito di questa funzione è di analizzare l'errore di sistema e intraprendere le opportune azioni per mettere la macchina in sicurezza.

La funzione **nomefunzione** è soggetta a due vincoli.

Primo deve accettare i seguenti parametri:

- il numero dell'errore di sistema, come Integer
- il task che in cui si è verificato l'errore, come Function
- il dispositivo che ha generato errore, come device

Secondo non può contenere un certo numero di istruzioni GPL. Vedere [l'elenco delle istruzioni non utilizzabili su interrupt](#).

In caso di Errori di Sistema multipli la funzione viene chiamata una volta per ogni errore generato, in maniera sequenziale. Se la funzione stessa genera un Errore di Sistema vengono terminati tutti i task.

Durante l'esecuzione della funzione, il task per cui è stata abilitata la gestione degli errori viene fermato e riprende solo al termine della prima funzione chiamata dall'istruzione ONERRSYS. In particolare, il task riprenderà l'esecuzione rieseguendo l'istruzione interrotta dall'errore di sistema.

Esempio

[Ciclo di Main con gestione Errori](#)

valore costante o variabile o nomecounter. Numero di ripetizioni

Descrizione

Ripete l'esecuzione delle istruzioni racchiuse tra l'istruzione REPEAT e l'istruzione ENDREP per un numero di volte indicato dalla variabile **valore**.
 Quando il programma incontra l'istruzione ENDREP il contatore del numero di ripetizioni viene decrementato e se non è minore o uguale a zero il blocco di istruzioni viene rieseguito partendo dall'istruzione sulla linea successiva al REPEAT. Le istruzioni vengono quindi eseguite almeno una volta (anche se il parametro valore è in partenza nullo o negativo).
 Quando le ripetizioni sono terminate viene eseguita l'istruzione successiva a ENDREP.
 Vedere anche l'istruzione [FOR/NEXT](#).

Esempio

```
; esempio di ciclo che movimenta un asse
; tra due quote per 10 volte
Function ciclo
  Repeat 10
    MovAbs asse,100
    waitinput switch,ON
    Movabs asse,-100
    waitinput switch, OFF
  EndRep
Fret
```

RET

Sintassi

RET

Argomenti

nessun argomento

Descrizione

Termina l'esecuzione del sottoprogramma con ritorno all'istruzione immediatamente successiva alla CALL di chiamata.
 Vedere anche l'istruzione [CALL](#)

Nota

Questa istruzione, assieme alla CALL, è una tipica fonte di errori di programmazione. Si consiglia di prestare attenzione al suo utilizzo, in particolare si consiglia di posizionare le sottoprocedure al termine del corpo di una funzione (dopo l'istruzione FRET) in modo da evitare l'esecuzione accidentale del codice della sottoprocedura come se fosse parte integrante del codice principale. Il risultato di questa situazione è nella migliore delle ipotesi un errore di sistema, in altri casi si ottengono comportamenti anomali della macchina la cui causa è di difficile individuazione.

SELECT

Sintassi

```
SELECT nomevar
  CASE valore
    istruzione
  CASE valore1 TO valore2
    istruzione
  CASE IS < => valore
    istruzione
  CASE ELSE
    istruzione
ENDSELECT
```

istruzione deve essere sostituita da uno dei seguenti valori:

```
GOTO etichetta
CALL nomesottoprogramma
[FCALL] nomefunzione [parametro1,...parametroN]
[EXPR] variabile =
[EXPR] dispositivo = espressione
```

Argomenti

nomevar	costante o variabile integer o nomecounter
valore, valore1, valore2	costanti integer
etichetta	nome dell'etichetta a cui saltare
nomesottoprogramma	nome del sottoprogramma
nomefunzione	nome della funzione
parametro1...parametroN	parametro passato alla funzione chiamata
variabile	nome di variabile
dispositivo	nome di dispositivo
espressione	insieme di operatori

Descrizione

Selezione multipla sulla base del **valore** della variabile **nomevar**. Viene eseguito il codice che si trova nel CASE della condizione verificata. Il ramo CASE - ELSE viene eseguito se nessun CASE precedente viene soddisfatto. Per ogni CASE (opzionale) può esserci una sola istruzione [GOTO](#), [CALL](#) o [FCALL](#) o [EXPR](#).

Ci deve essere almeno un CASE tra SELECT ed ENDSELECT. Quest'ultima segnala il termine dell'istruzione SELECT.

Dopo ogni CALL o FCALL o EXPR l'esecuzione della funzione continua all'istruzione successiva a ENDSELECT.

Esempio

[Server di movimentazione assi](#)

TESTIPC**Sintassi**

TESTIPC	nomeIPC, [, nomevar1 [, nomevarN, ...]], GOTO etichetta
TESTIPC	nomeIPC, [, nomevar1 [, nomevarN, ...]], CALL nomesottoprogramma
TESTIPC	nomeIPC, [, nomevar1 [, nomevarN, ...]], nomefunzione
TESTIPC	nomeIPC, matrice[riga], GOTO etichetta
TESTIPC	nomeIPC, matrice[riga], CALL nomesottoprogramma
TESTIPC	nomeIPC, matrice[riga], nomefunzione
TESTIPC	nomeIPC, vettore, GOTO etichetta
TESTIPC	nomeIPC, vettore, CALL nomesottoprogramma
TESTIPC	nomeIPC, vettore, nomefunzione

Argomenti

nomeIPC	costante stringa. Nome della IPC
nomevar1[...nomeVarN]	costante o variabile. Nomi variabili 1÷N
matrice[riga]	costante o variabile integer. Numero di riga della matrice
vettore	nome del vettore
matrice	nome della matrice
etichetta	nome dell'etichetta a cui saltare
nomesottoprogramma	nome del sottoprogramma
nomefunzione	nome della funzione

Descrizione

Test e ricezione di un comando IPC.

La prima volta che viene eseguita un'istruzione di TESTIPC viene allocata la memoria condivisa, la cui dimensione viene calcolata in base alla dimensione dei dati che vengono inviati. La dimensione massima della memoria condivisa è di 64 Kb.

Alla memoria condivisa è associato un semaforo che permette di sincronizzare l'esecuzione dei task che vi accedono. Il task che vi accede verifica la presenza di un semaforo attivo, legge i dati dalla memoria condivisa e disattiva il semaforo. Di seguito viene eseguita l'istruzione di salto ad etichetta, o la funzione o il sottoprogramma descritti come ultimo parametro dell'istruzione TESTIPC.

Vedere anche [SENDIPC](#) e [WAITIPC](#).

TESTMAIL**Sintassi**

TESTMAIL	mail, [nomevar1 [,..nomevarN]], GOTO etichetta
TESTMAIL	mail, [nomevar1 [,..nomevarN]], CALL etichetta
TESTMAIL	mail, [nomevar1 [,..nomevarN]], nomefunzione
TESTMAIL	mail, matrice[riga], GOTO etichetta
TESTMAIL	mail, matrice[riga], CALL nomesottoprogramma
TESTMAIL	mail, matrice[riga], nomefunzione

Argomenti

mail	costante o variabile integer (1÷256). Numero del mailbox
[nomevar1,..nomevarN]	variabile integer. Nomi variabili 1÷20
matrice[riga]	costante o variabile integer. Numero di riga della matrice
etichetta	nome dell'etichetta a cui saltare
nomesottoprogramma	nome del sottoprogramma
nomefunzione	nome della funzione

Descrizione

Test e ricezione di un messaggio.

La prima TESTMAIL sul mailbox **mail** crea il mailbox.

Se il messaggio è presente nella casella postale **mail**, i dati inviati insieme al messaggio vengono memorizzati nelle variabili **nomevar** (1÷20), solo se queste sono indicate, o nella riga della matrice indicata da **matrice[riga]**; viene inoltre effettuato un salto a **etichetta** o una chiamata a **nomesottoprogramma** o a **nomefunzione**.

Durante l'esecuzione viene controllata la congruenza tra i dati passati e quelli attesi dall'istruzione.

Vedere anche le istruzioni [SENDMAIL](#), [WAITMAIL](#) e [ENDMAIL](#).

10.3.13 Varie**CLEARERRORS****Sintassi**

CLEARERRORS [IDposiz]

Argomenti

IDposiz costante o variabile. È un valore numerico utilizzato nei sinottici

Descrizione

Fa eseguire al PC supervisore la cancellazione di tutti gli errori di ciclo, relativi al modulo che esegue l'istruzione, inviati precedentemente con l'istruzione ERROR. Il parametro **IDposiz** è un parametro opzionale che specifica il valore numerico utilizzato nei sinottici per smistare errori di ciclo in caselle differenti. Deve corrispondere al valore specificato nel costruttore sinottici per quella particolare casella di visualizzazione. Albatros utilizza questo identificativo per gestire gli errori di ciclo in code separate. Per ogni IDposiz viene creata una coda. Il range di valori impostabili è compreso tra 0 (NOPLACE) e 1023. Se il parametro **IDposiz** non è specificato si esegue la cancellazione di tutti gli errori di ciclo sia nella coda di default che nelle eventuali altre code.

Vedere anche le istruzioni [ERROR](#) e [DELEERROR](#).

CLEARMESSAGES**Sintassi**

CLEARMESSAGES [IDposiz]

Argomenti

IDposiz costante o variabile. È un valore numerico utilizzato nei sinottici

Descrizione

Fa eseguire al PC supervisore la cancellazione di tutti i messaggi, relativi al modulo che esegue l'istruzione, inviati precedentemente con l'istruzione MESSAGE. Il parametro **IDposiz** è un parametro opzionale che specifica il valore numerico utilizzato nei sinottici per smistare messaggi in caselle differenti. Deve corrispondere al valore specificato nel costruttore sinottici per quella particolare casella di visualizzazione. Albatros utilizza questo identificativo per gestire i messaggi in code separate. Per ogni IDposiz viene creata una coda. Il range di valori impostabili è compreso tra 0 (NOPLACE) e 1023. Se il parametro **IDposiz** non è specificato si esegue la cancellazione di tutti i messaggi sia nella coda di default che nelle eventuali altre code.

Vedere anche le istruzioni [MESSAGE](#) e [DELMESSAGE](#).

DEFMSG**Sintassi**

DEFMSG etichetta [, prefissolingua1], "stringamessaggio", ... , [, prefissolinguaN, "stringamessaggio"]

Argomenti

etichetta nome mnemonico del messaggio da visualizzare
prefissolingua costante predefinita. Lingua in cui è scritto il messaggio

Descrizione

Genera un errore di ciclo. L'errore è identificato dal parametro **numero** o dal nome di un dispositivo. Il parametro **numero** può identificare un errore di ciclo di modulo (quindi un valore numerico intero) o di gruppo (in questo caso si usa una DEFMSG). Se viene specificato un nome di dispositivo, al posto del numero, viene passato al PC il tipo e l'indirizzo logico del dispositivo. L'errore di ciclo viene inviato al PC supervisore e visualizzato sulla barra degli errori di Albatros.

Il parametro **IDposiz** viene utilizzato nei quadri sinottici per smistare errori di ciclo in caselle differenti. Deve corrispondere al valore specificato nel costruttore sinottici per quella particolare casella di visualizzazione. Albatros utilizza questo identificativo per gestire gli errori di ciclo in code separate. Per ogni IDposiz viene creata una coda. Se non è specificato IDposiz, o se si utilizza la costante predefinita NOPLACE, l'errore di ciclo va nella coda di default (IDposiz=0). Il range di valori impostabili è compreso tra 0 (NOPLACE) e 1023. L'impostazione del parametro **log a STORE** comporta la registrazione dell'errore di ciclo nel file di report errori del mese corrente. La generazione di un elevato numero di errori (o la loro cancellazione) può compromettere il livello di prestazioni dei moduli remoti. Il PC supervisore deve infatti gestire tutti gli errori inviati (e la loro eventuale cancellazione) e questo può rallentare l'invio di dati importanti al controllo, in particolare i programmi di lavorazione.

I parametri opzionali **arg1, ..., arg3** permettono di definire dei messaggi di errore parametrici. Nella stringa di definizione del messaggio di errore andranno inseriti dei marcatori che verranno sostituiti, quando viene generato l'errore, con il valore o il nome del dispositivo o della variabile passata come parametro. I marcatori da inserire nella stringa sono:

- \$1, ... \$2 sostituiti con il **nome** del dispositivo o della variabile (\$1 corrisponde ad arg1 etc.)
- \$(1), ..., \$(3) sostituiti con il **valore** del dispositivo o della variabile

I tipi di dati ammessi per i parametri arg1, ..., arg3 sono:

- CHAR
- INTEGER
- FLOAT
- DOUBLE (che viene però convertito automaticamente in FLOAT)
- numero di messaggio (o etichetta di DEFMSG)
- dispositivo
- variabile globale o locale
- parametro di funzione. È possibile utilizzare come parametro di funzione anche l'etichetta definita con una istruzione [DEFMSG](#)

Non possono essere utilizzati come parametri le stringhe, le matrici ed i vettori (i singoli elementi di vettore o matrice sono ammessi). Per le variabili locali è possibile decodificare solamente il valore e non il nome.

Al fine della cancellazione di un messaggio con l'istruzione `DELEERROR` i parametri arg1, ...arg3 vengono ignorati.

Sono definite due modalità di gestione degli errori di ciclo stabilite dal costruttore della macchina:

Allarmi gestiti come segnalazioni: vengono inviati tutti gli errori di ciclo. Albatros mantiene una coda degli ultimi 100 errori della coda specificata e degli ultimi 100 errori della coda di default.

Allarmi gestiti come stati: l'errore viene considerato attivo o non attivo. Se attivo ogni ulteriore invio dello stesso errore di ciclo (tramite istruzione `ERROR`) viene ignorato.

Vedere anche le istruzioni [DELEERROR](#), [CLEARERRORS](#).

Esempio 1

```
DEFMSG ERR_TOOL "Utensile non presente"
DEFMSG ERR_TOOL_P "Caricare l'utensile $(1) in posizione $(2)"
```

```
; tag per i sinottici
CONST TOOLCHANGE = 5
```

```
; errore visualizzato sulla barra degli errori o
; nelle caselle di sinottico non marcate.
ERROR ERR_TOOL
```

```
; errore salvato nei report e visualizzato nelle caselle
; di sinottico marcate con il codice 5
ERROR ERR_TOOL, TOOLCHANGE, STORE
```

```
; errore salvato nei report ma non associato
; alle caselle di sinottico
ERROR ERR_TOOL, NOPLACE, STORE
```

```
; errore con parametri
ERROR ERR_TOOL_P, NOPLACE, NOSTORE, MxUtensili[3].Cod, 5
```

Esempio 2

```

; definita in un gruppo
DEFMSG      MSG_ERR_CARICO      "Caricamento utensile non eseguito"

```

```

Function MostraMessaggio
MsgTool MSG_ERR_CARICO MxUtensili[3].Cod
fret

```

```

Function MostraErrore
ErrTool STORE MSG_ERR_CARICO MxUtensili[3].Cod
Fret

```

```

; definita in una libreria
DEFMSG      MSG_ERR_TOOL      "Errore utensile: $1 $(2)"

```

```

Function MsgTool Public
PARAM parameter1 as integer
PARAM parameter2 as integer

MESSAGE MSG_ERR_TOOL NOPLACE parameter1 parameter2
fret

```

```

Function ErrTool Public
PARAM log as integer
PARAM argument1 as integer
PARAM argument2 as integer

ERROR MSG_ERR_TOOL NOPLACE log argument1 argument2
fret

```

IFDEF/ELSEDEF/ENDDF**Sintassi**

IFDEF	istruzione	costante
	...	
ENDDF		
IFDEF	istruzione	costante, operatore di confronto, valore
	...	
ENDDF		
IFDEF	istruzione	EXIST, nomegruppo
	...	
ENDDF		
IFDEF	istruzione	LINKED, nomedevice
	...	
ENDDF		
IFDEF	istruzione	UNLINKED, nomedevice
	...	
ENDDF		

```

IFDEF           costante, operatore di confronto, valore
                 istruzione
                 ...
ELSEDEF        istruzione
                 ...
ENDDF

```

Argomenti

costante	costante di tipo integer, char, double, stringa
operatore di confronto	i simboli che possono essere utilizzati per eseguire il confronto sono: < (minore) = (uguale) > (maggiore) =< (minore uguale) >= (maggiore uguale) <> (diverso)
valore	costante o nome di dispositivo
nomegruppo	costante stringa o nome di gruppo
nomedevice	costante stringa o nome di dispositivo

Descrizione

La compilazione condizionale consente di controllare quali parti di un file di function GPL devono essere compilate e quindi eseguite. Il compilatore verifica che la condizione richiesta come argomento dell'istruzione IFDEF sia verificata. In questo caso il codice compreso tra l'istruzione IFDEF e l'istruzione ENDDF o ELSEDEF viene compilato. Se è presente l'istruzione ELSEDEF e la condizione non è verificata verrà compilato il codice compreso tra l'istruzione ELSEDEF e l'istruzione ENDDF.

La condizione di compilazione può essere espressa in diversi modi:

- dopo l'istruzione IFDEF viene specificato il nome di una [costante](#). In questo caso la condizione è verificata se esiste una costante globale o del gruppo corrente con il nome specificato.
- dopo l'istruzione IFDEF viene specificata una relazione tra due operatori ed un operando. Il primo operando deve essere una costante. In questo caso la condizione è verificata se la relazione è vera (ad es. MAX_TOOLS = 100).
- dopo l'istruzione IFDEF viene specificata la parola chiave EXIST o NOTEXIST seguita dal nome di un gruppo della macchina o da una stringa contenente il nome di un gruppo della macchina o il nome di una libreria. In questo caso la condizione è verificata se in Configurazione di Macchina è presente o non è presente un gruppo con lo stesso nome.
- dopo l'istruzione IFDEF viene specificata la parola chiave LINKED o UNLINKED seguita dal nome di un dispositivo. In questo caso la condizione è verificata se il dispositivo è collegato (LINKED) o non è collegato (UNLINKED) in virtuale-fisico. Il nome del dispositivo può essere espresso nella forma: Nome_Gruppo.Nome_Sottogruppo.Nome_Dispositivo oppure Nome_Gruppo.Nome_Dispositivo oppure Nome_Sottogruppo.NomeDispositivo oppure Nome_Dispositivo. Se il dispositivo non esiste in configurazione viene considerato come non collegato.

È possibile innestare più istruzioni IFDEF, tenendo presente che ad ogni istruzione IFDEF deve corrispondere una istruzione ENDDF.

Esempio 1

```

; l'esecuzione del codice GPL cambia a seconda
; della presenza in macchina del gruppo FRESA
Const GruppoFresa = "Fresa"
IFDEF Exist GruppoFresa
  istruzione
  istruzione
ELSEDEF
  istruzione
  istruzione
ENDDF

```

Esempio 2

```

; l'esecuzione del codice GPL cambia
; a seconda del modulo
IFDEF _ID_MODULE = 1 ; compila le istruzioni per il modulo 1
  istruzione
  istruzione
ELSEDEF ; compila le istruzioni per gli altri moduli
  istruzione
  istruzione
ENDDF

```

```
; compila il codice per la versione 3.2.0 di Albatros
```

```
IFDEF _VER_MAJOR = 3
  IFDEF _VER_MINOR = 2
    IFDEF _VER_REVISION = 0
      istruzione
      istruzione
    ENDDF
  ENDDF
ENDDF
```

```
; compila il codice per la versione
; di service pack 10 Albatros
```

```
IFDEF _VER_SP = "Service Pack 10"
  istruzione
ENDDF
```

```
; compila il codice solo se il sistema
; è configurato per un modulo remoto
```

```
IFDEF _REMOTE_MODULE = 1 ; 1 = modulo remoto, altrimenti 0 =
modulo locale
  istruzione
ENDDF
```

```
; compila il codice per la versione
; 2.4 service pack 10 Albatros
```

```
IFDEF _VER_FULL = $0002040AH
  istruzione
ENDDF
```

Esempio 3

```
; l'esecuzione del codice GPL cambia
; a seconda che il dispositivo sia collegato in virtuale fisico
```

```
IFDEF LINKED out1 ; se out1 è collegato il codice viene
eseguito
  istruzione
  istruzione
  istruzione
ENDIF
```

MESSAGE

Sintassi

```
MESSAGE                   numero [, IDposiz [, arg1, ..., arg3]]
```

Argomenti

numero	DEFMSG o costante o variabile
IDposiz	costante o variabile. Valore numerico utilizzato nei sinottici
arg1, ..., arg3	costante o dispositivo o variabile.

Descrizione

Genera un messaggio per l'operatore. Il messaggio è identificato dal parametro **numero**. Il parametro **numero** può identificare un messaggio di modulo (quindi un valore numerico intero) o di gruppo (in questo caso si usa una DEFMSG). Opzionalmente può essere passato un argomento, rappresentato da **IDposiz**. Indica su quale finestra di sinottico visualizzare il messaggio. Deve corrispondere al valore specificato nel costruttore sinottici per quella particolare casella di visualizzazione. Albatros utilizza questo identificativo per gestire i messaggi in code separate. Per ogni IDposiz viene creata una coda. Se non è specificato IDposiz, il messaggio va nella coda di default (IDposiz=0). Il range di valori impostabili è compreso tra 0 (NOPLACE) e 1023.

I parametri opzionali **arg1, ..., arg3** permettono di definire dei messaggi parametrici. Nella stringa di definizione del messaggio andranno inseriti dei marcatori che verranno sostituiti, quando viene generato il messaggio, con il valore o il nome del dispositivo o della variabile passata come parametro. I marcatori da inserire nella stringa sono:

- \$1, ... \$2 sostituiti con il **nome** del dispositivo o della variabile (\$1 corrisponde ad arg1 etc.)
 - \$(1), ..., \$(3) sostituiti con il **valore** del dispositivo o della variabile
- I tipi di dati ammessi per i parametri arg1, ..., arg3 sono:
- CHAR
 - INTEGER
 - FLOAT
 - DOUBLE (che viene però convertito automaticamente in FLOAT)
 - numero di messaggio (o etichetta di DEFMSG)
 - dispositivo
 - variabile globale o locale
 - parametro di funzione. È possibile utilizzare come parametro di funzione anche l' etichetta definita con una istruzione [DEFMSG](#)

Sono definite due modalità di gestione dei messaggi stabilite dal costruttore della macchina:

Messaggi gestiti come segnalazioni: vengono inviati tutti i messaggi. Albatros mantiene una coda degli ultimi 100 messaggi della coda specificata e degli ultimi 100 messaggi della coda di default. Quando la coda dei messaggi è piena viene sovrascritto il messaggio più vecchio. Se il messaggio precedente della coda è identico a quello che si sta inviando, il messaggio non viene inviato (task uguale, numero uguale, argomento uguale).

Messaggi gestiti come stati: il messaggio viene considerato attivo o non attivo. Se attivo ogni ulteriore invio dello stesso messaggio (tramite istruzione MESSAGE) viene ignorata.

Non possono essere utilizzati come parametri le stringhe, le matrici ed i vettori (i singoli elementi di vettore o matrice sono ammessi). Per le variabili locali è possibile decodificare solamente il valore e non il nome. Al fine della cancellazione di un messaggio con l'istruzione DELMESSAGE i parametri arg1, ...arg3 vengono ignorati.

Vedere anche le istruzioni [DELMESSAGE](#) e [CLEARMESSAGES](#).

Esempio 1

```
DEFMSG      MSG_TOOL      "Effettuare cambio utensile"
DEFMSG      MSG_TOOL_P    "Caricato utensile n° $(1)"

;tag per i sinottici
CONST      TOOLCHANGE = 7

;messaggio visualizzato sulla barra degli errori o
;nelle caselle di sinottico non marcate.
MESSAGE    MSG_TOOL

;messaggio visualizzato sulla barra degli errori e
;nelle caselle di sinottico marcate con il codice 7
MESSAGE    MSG_TOOL, TOOLCHANGE

; messaggio con parametri
MESSAGE    MSG_TOOL_P, NOPLACE, MxUtensili[3].Cod
```

Esempio 2

```
; definita in un gruppo

DEFMSG      MSG_CARICO    "Caricamento eseguito"

Function MostraMessaggio
  MsgTool MSG_CARICO MxUtensili[3].Cod
fret

; definita in una libreria
DEFMSG      MSG_TOOL      "utensile: $(1) $2"

Function MsgTool public
PARAM parameter1 as integer
PARAM parameter2 as integer

MESSAGE MSG_TOOL NOPLACE parameter1 parameter2
fret
```

SYSFAULT

Sintassi

SYSFAULT

Argomenti

nessun argomento

Descrizione

Disattiva il segnale di SYSOK.

Questo segnale viene disattivato per indicare che la macchina non è più in sicurezza (es.: non sono più in esecuzione i task GPL che gestiscono le emergenze).

Vedere anche l'istruzione [SYSOK](#).

SYSOK

Sintassi

SYSOK [nomeoutput1 [, ... nomeoutput8]]

Argomenti

nomeoutput1 [...nomeoutput8] nome dispositivo di tipo output digitale

Descrizione

Indica al controllo numerico quali sono le uscite collegate ai circuiti di sicurezza della macchina (può essere un'uscita collegata ad un relè di sicurezza che controlla l'alimentazione di potenza della macchina). Le uscite vengono attivate quando il controllo numerico ha completato l'inizializzazione della macchina ed ha attivato tutti i task di gestione delle emergenze. La macchina può essere, quindi, considerata in condizione di sicurezza. Possono essere definite fino a un totale di 8 uscite digitali. Su ogni remoto può essere abilitata una sola uscita. L'elenco delle uscite dichiarate nel primo utilizzo della istruzione SYSOK **non** può essere modificato nelle eventuali successive chiamate al sysok, se non dopo aver inizializzato il controllo. Se si esegue l'istruzione senza parametri viene ripristinato il segnale di SYSOK.

Vedere anche l'istruzione [SYSFAULT](#).

Nota

L'istruzione SYSYOK può essere abilitata solo:

- su tutti i moduli remoti GreenBus v3.0 con uscite digitali.
- sui moduli remoti GreenBus v4.0 di tipo TRS-IO.
- sui moduli remoti TRS-CAT, solo sulla base e non sulle espansioni, che hanno una versione firmware 1.2 o superiore (rev 1.00 del modulo remoto).

TYPEOF

Sintassi

TYPEOF **nome, risultato**

Argomenti

nome nome di dispositivo, costante, nomefunzione, variabile, vettore, matrice o riga di matrice

risultato variabile integer. Tipo del primo argomento

Descrizione

Ritorna nella variabile **risultato** il tipo dell'argomento **nome**.

WATCHDOG

Sintassi

WATCHDOG **stato**

Argomenti

stato costante predefinita. Può assumere i seguenti valori: **ON, OFF**

Descrizione

L'istruzione abilita l'uso del watchdog collegato all'hardware TMSWD. Permette di individuare situazioni di errore che si verificano durante l'esecuzione del codice GPL.

La prima volta che viene eseguita con il parametro **stato** a ON, l'istruzione abilita l'uso del watchdog. Tutte le volte successive si deve assegnare ON al parametro **stato** per effettuare l'aggiornamento del contatore della scheda. Se l'aggiornamento non viene eseguito, il watchdog scatta e il modulo TMSWD disattiva l'uscita di sicurezza della macchina.

Per terminare l'uso del watchdog si deve assegnare OFF al parametro **stato**.

Questa istruzione è utilizzabile solo con schede TMSbus+, TMSCan+ e TMSCombo+ con versione FPGA 2.0 o superiore e con montato il modulo hardware TMSWD.

Esempio

Function TestWatchDog autorun

```
watchdog ON ; abilita la gestione del watchdog
```

Loop:

```
watchdog ON ; esegue l'aggiornamento del contatore della scheda
```

```
goto loop
```

```
fret
```

10.3.14 MECHATROLINK-II

MECCOMMAND

Sintassi

MECCOMMAND **asse,comando,parametri,risposta,errore**

Argomenti

asse	nome di dispositivo di tipo asse digitale.
comando	costante integer.
parametri	vettore di integer.
risposta	vettore di integer.
errore	variabile integer. Codice di errore

Descrizione

Invia all'azionamento dell'**asse** indicato un **comando** e ne attende la risposta. I dati necessari all'esecuzione del comando vengono inseriti nel vettore **parametri**, mentre i dati ritornati dall'esecuzione dell'istruzione vengono memorizzati nel vettore **risposta**. I vettori **parametri** e **risposta** devono avere la stessa dimensione e un numero di elementi massimo di 14. Il valore considerato è il byte più basso dei singoli interi. Il parametro **errore** contiene i codici degli eventuali errori generati durante l'operazione. I codici di errore devono essere gestiti da Gpl come errori di ciclo.

I codici di **errore** restituiti sono:

Codice	Messaggio
Errore	
-40	Comando non permesso nelle attuali condizioni di funzionamento
-41	Errore di timeout durante l'esecuzione di un comando MECHATROLINK-II
-44	Errore di timeout durante l'esecuzione di un sottocomando MECHATROLINK-II
-45	Errore di collegamento dell'azionamento

Per i valori da assegnare ai parametri **comando**, **parametri**, **risposta** ed **errore** si faccia riferimento alla documentazione ufficiale Yaskawa MECHATROLINK-II. In essa i valori da assegnare per il comando sono descritti a partire dall'indice 2 fino all'indice 15. I valori da impostare per i sottocomandi sono descritti a partire dall'indice 18 fino all'indice 32.

I valori di **comando** possono variare come di seguito rispetto alla documentazione ufficiale.

I comandi possono essere distinti in:

- comando. Hanno codice compreso tra 0x00 e 0xFF. Per ragioni di sicurezza vengono eseguiti solo con servo asse disabilitato.
- sottocomando. I comandi intesi come sottocomandi devono sommare al valore documentato il codice 0x100. Ad esempio il comando NOP ha codice documentato 0x00, inteso come sottocomando è 0x100.
- procedura. I comandi intesi come procedure hanno codice con valore a partire da 0x200. Attualmente sono previste queste procedure:
 - \$201H procedura di abilitazione parametri offline (da usare ad asse disabilitato)

Questa istruzione è utilizzabile solo con schede AlbMech, DualMech e DualMech Mono. Per ulteriori informazioni riguardanti l'utilizzo di questa istruzione consultare la TPA.

Nota

Questa istruzione agisce sul comportamento di assi digitali e quindi va utilizzata in un contesto controllato

MECGETPARAM

Sintassi

MECGETPARAM **asse,parametro,dimensione,dato,errore**

Argomenti

asse	nome di dispositivo di tipo asse digitale.
parametro	costante o variabile integer.
dimensione	costante o variabile integer.
dato	variabile integer.
errore	variabile integer. Codice di errore

Descrizione

Legge un parametro dell'azionamento dell'**asse** indicato e lo memorizza nella variabile **dato**. Il parametro **errore** contiene i codici degli eventuali errori generati durante l'operazione. I codici di errore devono essere gestiti da Gpl come errori di ciclo.

I codici di **errore** restituiti sono:

Codice	Messaggio
Errore	
-40	Comando non permesso nelle attuali condizioni di funzionamento
-41	Errore di timeout durante l'esecuzione di un comando MECHATROLINK-II
-44	Errore di timeout durante l'esecuzione di un sottocomando MECHATROLINK-II
-45	Errore di collegamento dell'azionamento

Per i valori da assegnare alla variabili **parametro** e **dimensione** si faccia riferimento alla documentazione ufficiale Yaskawa MECHATROLINK-II.

Questa istruzione è utilizzabile solo con schede AlbMech, DualMech e DualMech Mono. Per ulteriori informazioni riguardanti l'utilizzo di questa istruzione consultare la TPA.

MECGETSTATUS

Sintassi

MECGETSTATUS **asse,stato,inout,errore**

Argomenti

asse	nome di dispositivo di tipo asse digitale.
stato	costante o variabile integer.
inout	costante o variabile integer.
errore	variabile integer. Codice di errore

Descrizione

Legge e memorizza nella variabile **stato** il valore di STATUS e di ALARM e nella variabile **inout** il valore di IO_MON relativi all'**asse** specificato. Per i valori di STATUS, ALARM, IO_MON si faccia riferimento alla documentazione ufficiale Yaskawa MECHATROLINK-II.

Il parametro **errore** contiene i codici degli eventuali errori generati durante l'operazione. I codici di errore devono essere gestiti da Gpl come errori di ciclo.

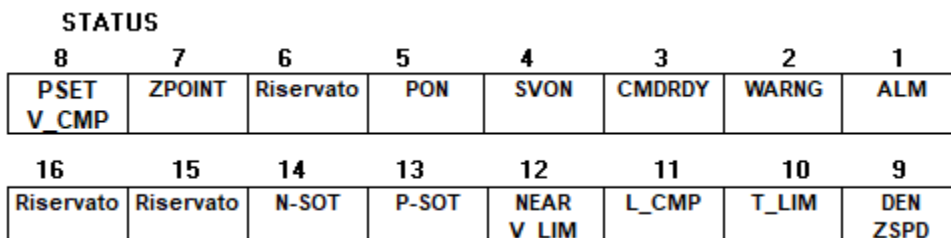
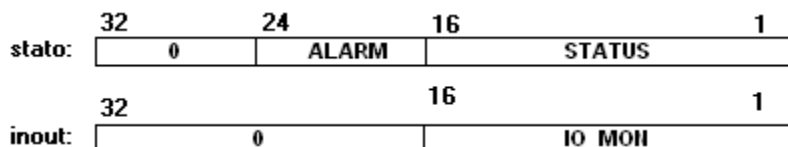
I codici di **errore** restituiti sono:

Codice	Messaggio
Errore	
-40	Comando non permesso nelle attuali condizioni di funzionamento
-41	Errore di timeout durante l'esecuzione di un comando MECHATROLINK-II
-44	Errore di timeout durante l'esecuzione di un sottocomando MECHATROLINK-II
-45	Errore di collegamento dell'azionamento

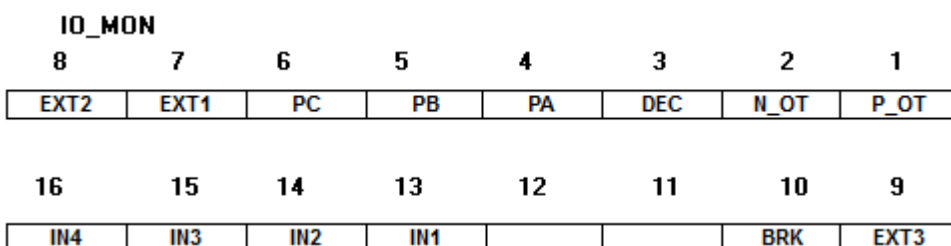
Sono definite una serie di categorie di errori. Le categorie rappresentano il valore del nibble più alto di ALARM.

Per uscire da allarmi che rientrano in una delle seguenti categorie 0x30,0x70,0xD0,0xF0 deve essere inviato un comando di CLEAR (0x06). Allarmi che rientrano in una delle seguenti categorie 0x00,0x10,0x40,0xB0 non possono essere eliminati con alcun comando. È necessario risolvere il problema che genera l'allarme, spegnere il servozionamento e riaccenderlo.

La struttura delle variabili **stato** e **inout** è una maschera di bit organizzata come nella rappresentazione qui sotto:

**Significato dei bit di STATUS**

Bit	Comando	Pin fisici collegabili in virtuale fisico
1	ALM (Alarm)	Ingresso digitale
2	WARNG (Warning)	Ingresso digitale
3	CMDRDY (Command Ready)	
4	SVON (Servo ON)	Uscita digitale
5	PON (Main Power ON)	Ingresso digitale
6	Riservato	
7	ZPOINT (Zero Point)	
8	PSET (Position Complete) V_CMP (Velocity Agreement)	
9	DEN (Command Distribution Completed Flag) ZSPD (Zero Velocity)	
10	T_LIM (Torque Limit)	
11	L_CMP (Latch Completed)	
12	NEAR (Position Proximity) V_LIM (Velocity Limit)	
13	P-SOT (Forward-direction Software Limit)	
14	N-SOT (Reverse-direction Software Limit)	
15	Riservato	
16	Riservato	

**Significato dei bit di IO_MON**

Bit	Comando	Pin fisici collegabili in virtuale fisico
-----	---------	---

1	P_OT (Forward Over Travel)	
2	N_OT (Reverse Over Travel)	
3	DEC (Deceleration Limit Switch)	
4	PA (Phase A)	
5	PB (Phase B)	
6	PC (Phase C)	Ingresso digitale
7	EXT1 (First external latch input)	Ingresso digitale
8	EXT2 (Second external latch input)	Ingresso digitale
9	EXT3 (Third external latch input)	
10	BRK (Brake output)	
11		
12		
13	IN1 (General-purpose input 1)	
14	IN2 (General-purpose input 2)	
15	IN3 (General-purpose input 3)	
16	IN4 (General-purpose input 4)	

Questa istruzione è utilizzabile solo con schede AlbMech, DualMech e DualMech Mono. Per ulteriori informazioni riguardanti l'utilizzo di questa istruzione consultare la TPA.

MECSETPARAM

Sintassi

MECSETPARAM **asse,parametro,dimensione,dato,errore**

Argomenti

asse	nome di dispositivo di tipo asse digitale.
parametro	costante o variabile integer.
dimensione	costante o variabile integer.
dato	variabile integer.
errore	variabile integer. Codice di errore

Descrizione

Scrivere un **dato** nel **parametro** dell'**asse** indicato.

Per i valori da assegnare alle variabili **parametro** e **dimensione** si faccia riferimento alla documentazione ufficiale Yaskawa MECHATROLINK-II. Il parametro **errore** contiene i codici degli eventuali errori generati durante l'operazione. I codici di errore devono essere gestiti da Gpl come errori di ciclo.

I codici di **errore** restituiti sono:

Codice	Messaggio
Errore	
-40	Comando non permesso nelle attuali condizioni di funzionamento
-41	Errore di timeout durante l'esecuzione di un comando MECHATROLINK-II
-44	Errore di timeout durante l'esecuzione di un sottocomando MECHATROLINK-II
-45	Errore di collegamento dell'azionamento

Questa istruzione è utilizzabile solo con schede AlbMech, DualMech e DualMech Mono. Per ulteriori informazioni riguardanti l'utilizzo di questa istruzione consultare la TPA.

Nota

Questa istruzione agisce sul comportamento di assi digitali e quindi va utilizzata in un contesto controllato.

Per scrivere un dato nella memoria non-volatile si deve utilizzare l'istruzione [MECCOMMAND](#)

10.3.15 Bus di campo standard

AXCONTROL

Sintassi

AXCONTROL **asse,dato**

Argomenti

asse nome di dispositivo tipo asse
dato variabile o costante integer. Imposta la ControlWord

Descrizione

Imposta il **dato** della ControlWord, in modo conforme all'operatività attiva, secondo "CiA 402 CANopen device profile".

Tabella definizione dei valori per EtherCAT

CONTROLWORD						
8	7	5	4	3	2	1
fr	oms		eo	qs	ev	so
16	12		11	10	9	
ms			r	oms	h	

Bit	Significato	Nome in virtuale fisico
1	so=Switch ON	CW1
2	ev=Enable voltage	EV
3	qs=Quick stop	STOP
4	eo=Enable operation	SVON
5	oms=Operation mode specific	CW5
6	oms=Operation mode specific	CW6
7	oms=Operation mode specific	CW7
8	fr=Fault reset	RESALM
9	h=Halt	CW9
10	oms=Operation mode specific	CW10
11	r=Reserved	CW11
12	ms=Manufacturer specific	CW12
13	ms=Manufacturer specific	CW13
14	ms=Manufacturer specific	CW14
15	ms=Manufacturer specific	CW15
16	ms=Manufacturer specific	CW16

Tabella definizione dei valori per S-CAN

CONTROLWORD						
8	7	5	4	3	2	1
fr	oms		eo	qs	ev	so

Bit	Significato	Nome in virtuale fisico
1	Ten_cmd=comando abilitazione coppia. 1:asse in coppia 0:asse libero	SVON
2	Ien_cmd=comando abilitazione movimentazioni. 1:movimentazioni abilitate 0:asse in fermo coppia	ENMOVE
3	Stp_cmd=comando stop. 1:comando di stop attivo 0:comando di stop non attivo	STOP
4	Alm_rst=stato allarmi 1:comando reset allarmi	RESALM
5	Ltc_rst:reset bit 5 di StatusWord	CW5
6	oms=specifico del modo selezionato	CW6
7	oms=specifico del modo selezionato	CW7
8	oms=specifico del modo selezionato	CW8

AXSTATUS

Sintassi

AXSTATUS

asse, valore

Argomenti

asse

nome di dispositivo tipo asse

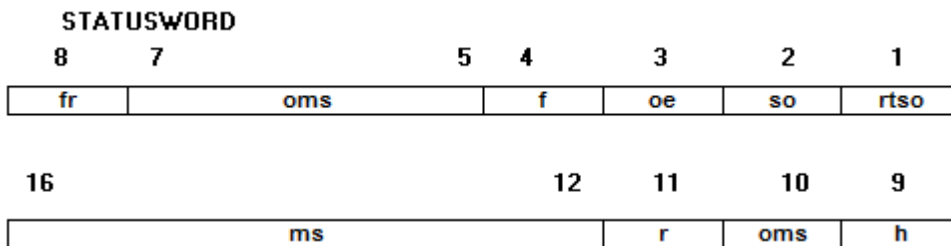
valore

variabile integer.

Descrizione

Restituisce il valore contenuto nella StatusWord in modo conforme a "CiA 402 CANopen device profile".

Tabella definizione dei valori per EtherCAT



Bit	Significato	Nome in virtuale fisico
1	rtso=Ready to switch on	RTSO
2	so=Switched on	SW2
3	oe=Operation enabled	OE
4	f=Fault	ALM
5	ve=Voltage enable	VE
6	qs=Quick stop	QS
7	sod=Switch on disabled	SOD
8	w=Warning	WARNG

9	ms=Manufacturer specific	SW9
10	rm=Remote	SW10
11	tr=Target reached or reserved	SW11
12	ila=Internal limit active	SW12
13	oms=Operation mode specific	SW13
14	oms=Operation mode specific	SW14
15	ms=Manufacturer specific	SW15
16	ms=Manufacturer specific	SW16

Tabella definizione dei valori per S-CAN

STATUSWORD							
8	7	6	5	4	3	2	1
w	sod	qs	ve	f	oe	so	rtso

Bit	Significato	Nome in virtuale fisico
1	Ten_st=stato abilitazione coppia 1:asse in coppia 0:asse libero	SW1
2	Ien_st=stato abilitazione movimentazioni 1:movimentazioni abilitate 0:asse in fermo coppia	SW2
3	Stp_st=stato stop 1:rampa di stop in corso 0:stop non attivato o rampa conclusa	SW3
4	Alm_st=stato allarmi 1:macchina in allarme 0:nessun allarme rilevato	ALM
5	Ltc_st=stato latch posizione 1:latch di posizione eseguito, registro pronto per la lettura 0:nessun latch di posizione rilevato	SW5
6	oms=specifico del modo selezionato	SW6
7	oms=specifico del modo selezionato	SW7
8	oms=specifico del modo selezionato	SW8

CNBYDEVICE

Sintassi

CNBYDEVICE**dispositivo,scheda,cn**

Argomenti

dispositivo
scheda
cnnome di dispositivo.
variabile integer. Numero di scheda restituito
variabile integer. Numero di nodo restituito

Descrizione

Restituisce numero di scheda e numero di nodo del dispositivo definito nel parametro **dispositivo**. Questa istruzione serve per poter utilizzare le istruzioni che non hanno dei riferimenti diretti a dispositivi, come, ad esempio, le istruzioni [READDICTIONARY](#) e [WRITEDICTIONARY](#).

L'istruzione può essere utilizzata per dispositivi configurati su bus CAN, S-CAN, GreenBUS o EtherCAT.

Se il numero nodo restituito è un valore negativo, significa che il nodo è disabilitato.

READDICTIONARY

Sintassi

READDICTIONARY **scheda,cn,indice,sottoindice,dimdato,dato,errore**

Argomenti

scheda costante o variabile integer. Numero scheda
cn costante o variabile integer. Numero nodo
indice costante o variabile integer. Indice oggetto nel dizionario
sottoindice costante o variabile integer. Sottoindice oggetto nel dizionario
dimdato variabile integer. Grandezza dato letto.
dato variabile char, integer,float,double,vettore di char,string. Variabile che riceve il dato
errore variabile integer. Codice di errore restituito dal nodo

Descrizione

Legge il contenuto di un oggetto del dizionario di oggetti contenuti nel nodo. L'istruzione permette di leggere tramite il protocollo SDO tutti gli oggetti definiti secondo "CiA 402 CANopen device profile" oltre a tutti gli oggetti resi disponibili dal costruttore del nodo. Per il significato dei parametri **indice**, **sottoindice** e **dimdato** fare riferimento a "CiA 402 CANopen device profile" o alla specifica del costruttore del nodo. Per i dispositivi S-CAN il parametro sottoindice va sempre impostato a zero.

WRITEDICTIONARY

Sintassi

WRITEDICTIONARY **scheda,cn,indice,sottoindice,dimdato,dato,errore**

Argomenti

scheda costante o variabile integer. Numero scheda
cn costante o variabile integer. Numero nodo
indice costante o variabile integer. Indice oggetto nel dizionario
sottoindice costante o variabile integer. Sottoindice oggetto nel dizionario
dimdato costante o variabile integer. Dimensione del dato da scrivere
dato variabile char, integer,float,double,vettore di char,string. Variabile che contiene il dato
errore variabile integer. Codice di errore restituito dal nodo

Descrizione

Scriva il contenuto di un oggetto del dizionario di oggetti contenuti nel nodo. L'istruzione permette di scrivere tramite il protocollo SDO tutti gli oggetti secondo "CiA 402 CANopen device profile" oltre a tutti gli oggetti resi disponibili dal costruttore del nodo. Per il significato dei parametri **indice**, **sottoindice** e **dimdato** da scrivere fare riferimento a "CiA 402 CANopen device profile" o alla specifica del costruttore del nodo. Per i dispositivi S-CAN il parametro sottoindice va sempre impostato a zero.

10.3.16 EtherCAT

ACTIVATEMODE

Sintassi

ACTIVATEMODE **asse,dato,errore**

Argomenti

asse nome di dispositivo tipo asse
dato costante o variabile integer. Modalità operativa
errore variabile integer. Codice di errore restituito dal servo

Descrizione

Imposta la modalità operativa definita nella variabile **dato** in modo conforme a "CiA 402 CANopen device profile". La modalità operativa dell'asse in partenza corrisponde al valore di **dato=9**, ossia "modalità di velocità sincrona ciclica". L'istruzione restituisce valore **errore=0** se il comando è riuscito, altrimenti restituisce un codice di errore.

Di seguito la tabella dei valori da assegnare a **dato** per la scelta della modalità operativa.

Valore	Definizione
+6	modalità homing

+9	modalità di velocità sincrona ciclica
----	---------------------------------------

ECATGETREGISTER

Sintassi

ECATGETREGISTER **nodo,indirizzo,dim,dato,errore**

Argomenti

nodo	costante o variabile integer. Posizione occupata dallo slave nella catena EtherCAT (da 1 in poi)
indirizzo	costante o variabile integer. Indirizzo del registro dell'ESC (EtherCAT Slave Controller) da leggere (da 0 in poi)
dim	costante o variabile integer. Numero di byte da leggere (1, 2 o 4)
dato	variabile integer. Contenitore del dato letto
errore	variabile integer. Codice di errore.

Descrizione

Restituisce **[dato]** il contenuto di un registro dell'ESC (EtherCAT Slave Controller) del nodo EtherCAT indicato. Il parametro **errore** conterrà il codice numerico dell'errore, 0 se non si sono verificati errori.

ECATSETREGISTER

Sintassi

ECATSETREGISTER **nodo,indirizzo,dim,dato,errore**

Argomenti

nodo	costante o variabile integer. Posizione occupata dallo slave nella catena EtherCAT (da 1 in poi)
indirizzo	costante o variabile integer. Indirizzo del registro dell'ESC (EtherCAT Slave Controller) da scrivere (da 0 in poi)
dim	costante o variabile integer. Numero di byte da scrivere (1, 2 o 4)
dato	costante o variabile integer. Dato che deve essere scritto
errore	variabile integer. Codice di errore

Descrizione

L'istruzione scrive il contenuto **[dato]** di un registro dell'ESC (EtherCAT Slave Controller) del nodo EtherCAT indicato. Il parametro **errore** conterrà il codice numerico dell'errore, 0 se non si sono verificati errori.

GETPDO

Sintassi

GETPDO **scheda,nodo,nPDO,nObj,dato,[errore]**

Argomenti

scheda	costante o variabile integer. Numero di scheda
nodo	costante o variabile integer. Posizione occupata dallo slave nella catena EtherCAT (da 1 in poi)
nPDO	costante o variabile integer. Identificativo del PDO (es. \$1600h) o posizione dello stesso nella lista di PDO definiti in configurazione hardware per il nodo in esame (da 1 a 16)
nObj	costante o variabile integer. Identificativo dell'oggetto (es. \$6040h) o posizione dello stesso all'interno della lista di oggetti configurati nel PDO (da 1 a 32)
dato	variabile integer. Riceve il valore.
errore	variabile integer. Codice di errore.

Descrizione

Restituisce in **[dato]** il contenuto di un oggetto scambiato tramite i PDO configurati per il nodo EtherCAT. Se gli argomenti passati sono errati e non è stato impostato il parametro **errore**, viene generato un errore di sistema. Se è stato impostato un parametro **errore** questo conterrà il codice numerico del corrispondente errore di sistema.

SETEOE

Sintassi

SETEOE **stato,[errore]**

Per maggiori informazioni sul significato di questi parametri riferirsi direttamente alla documentazione del singolo dispositivo.

Valore	Stato del protocollo
1	Operational
128	Pre-Operational

10.3.18 Simulazione

DISABLE

Sintassi

DISABLE **asse1,[...asse6]**

Argomenti

asse1...[...asse6] nomi di dispositivi tipo asse

Descrizione

Disabilita gli assi specificati. La disabilitazione permette di effettuare delle simulazioni della ciclica di macchina senza movimentare realmente gli assi. Un asse disabilitato non legge le informazioni che provengono dall'encoder ma simula un errore d'anello proporzionale alla velocità corrente. La disabilitazione dell'asse non disattiva tuttavia il riferimento di velocità, si avrà quindi una tensione diversa da zero sul connettore dell'asse anche durante i movimenti simulati. Pertanto, è necessario disalimentare gli azionamenti o scollegarli dalla scheda assi durante i movimenti simulati ovvero ad assi disabilitati. Vedere anche [ENABLE](#).

Nota

Assi di tipo passo-passo possono essere utilizzati in questa istruzione solo se gestiti da un remoto TRS-AX.

DISABLEFORCEDINPUT

Sintassi

DISABLEFORCEDINPUT

Argomenti

nessun argomento

Descrizione

Disabilita la possibilità di utilizzare le funzioni per la forzatura degli ingressi. Se in precedenza erano stati forzati degli ingressi l'esecuzione di questa istruzione li riporta nello stato reale. Vedere anche [ENABLEFORCEDINPUT](#), [DISABLEFORCEDINPUT](#), [SETFORCEDINPUT](#), [RESETFORCEDINPUT](#), [SETFORCEDPORT](#), [SETFORCEDANALOG](#).

ENABLE

Sintassi

ENABLE **asse1,[...asse6]**

Argomenti

asse1...[...asse6] nomi di dispositivi tipo asse

Descrizione

Abilita gli assi specificati. Gli assi in fase di inizializzazione sono sempre abilitati. Questa istruzione va chiamata solo se gli assi sono stati precedentemente disabilitati da una istruzione [DISABLE](#).

Nota

Assi di tipo passo-passo possono essere utilizzati in questa istruzione solo se gestiti da un remoto TRS-AX.

ENABLEFORCEDINPUT

Sintassi

ENABLEFORCEDINPUT

Argomenti

nessun argomento

Descrizione

Abilita la forzatura degli ingressi. Prima di utilizzare le istruzioni per abilitare o disabilitare i dispositivi di ingresso in modo forzato, è necessario eseguire questa istruzione. In caso contrario le istruzioni di forzatura degli ingressi non producono alcun effetto.

Vedere anche [DISABLEFORCEDINPUT](#), [SETFORCEDINPUT](#), [RESETFORCEDINPUT](#), [SETFORCEDPORT](#), [SETFORCEDANALOG](#).

RESETFORCEDINPUT

Sintassi

RESETFORCEDINPUT **nomeinput**

Argomenti

nomeinput nome ingresso digitale

Descrizione

Forza ad OFF l'ingresso specificato da **nomeinput**.

Per poter utilizzare questa istruzione è necessario aver già abilitato la forzatura degli ingressi, con l'istruzione [ENABLEFORCEDINPUT](#). Vedere anche [DISABLEFORCEDINPUT](#), [SETFORCEDINPUT](#), [SETFORCEDPORT](#), [SETFORCEDANALOG](#).

SETFORCEDANALOG

Sintassi

SETFORCEDANALOG **ingressoanalogico, valore**

Argomenti

**ingressoanalogico
variabile** nome di dispositivo ingresso analogico
costante o variabile integer o float o double

Descrizione

Forza il **valore** nell'ingresso analogico specificato da **ingressoanalogico**.

Per poter utilizzare questa istruzione è necessario aver già abilitato la forzatura degli ingressi, con l'istruzione [ENABLEFORCEDINPUT](#). Vedere anche [DISABLEFORCEDINPUT](#), [SETFORCEDINPUT](#), [RESETFORCEDINPUT](#), [SETFORCEDPORT](#).

SETFORCEDINPUT

Sintassi

SETFORCEDINPUT **nomeinput**

Argomenti

nomeinput nome ingresso digitale

Descrizione

Forza ad ON l'ingresso specificato da **nomeinput**.

Per poter utilizzare questa istruzione è necessario aver già abilitato la forzatura degli ingressi, con l'istruzione [ENABLEFORCEDINPUT](#).

Vedere anche [DISABLEFORCEDINPUT](#), [RESETFORCEDINPUT](#), [SETFORCEDPORT](#), [SETFORCEDANALOG](#).

SETFORCEDPORT

Sintassi

SETFORCEDPORT **nomeporto, valore**

Argomenti

**nomeporto
variabile** nome di dispositivo porto di ingressi
costante o variabile integer o char

Descrizione

Forza il **valore** nel porto di ingressi espresso da **nomeporto**. Il porto di ingressi viene visto come una maschera di bit. Se un bit vale 1, il corrispondente ingresso viene forzato ad "ON".

Per poter utilizzare questa istruzione è necessario aver già abilitato la forzatura degli ingressi, con l'istruzione [ENABLEFORCEDINPUT](#).

Vedere anche [DISABLEFORCEDINPUT](#), [SETFORCEDINPUT](#), [RESETFORCEDINPUT](#), [SETFORCEDANALOG](#).

10.3.19 BlackBox

Lo scopo della funzionalità "BlackBox" è quello di registrare in un database tutta l'attività di una macchina, ovvero un modulo locale o remoto. Per "attività di una macchina" si intende la variazione nel tempo di un sottoinsieme di

tutti i dispositivi logici utilizzabili in GPL. In tal modo è possibile analizzare a posteriori il comportamento della macchina, correlando gli stati dei dispositivi memorizzati. Il database ha una tabella contenente un'informazione temporale e lo stato di tutti i dispositivi in quell'istante, uno per ogni colonna. Nel linguaggio GPL sono state introdotte nuove istruzioni per avviare, terminare e interrogare l'attività di registrazione e sono descritte di seguito.

Ogni file di blackbox è un database SQLite e contiene informazioni relative a un solo modulo. Il nome del file include il numero del modulo e la data e l'ora di quando è stata avviata l'attività di registrazione.

L'aggiunta di record nel database è effettuata in modo transazionale. Ogni transazione contiene al più i record generati in 1 secondo. In caso di blackout è garantita la coerenza del file e può essere persa l'ultima transazione. La durata limite della transazione può essere modificata con una voce in Tpa.ini (per informazioni contattare la TPA).

È stato inserito il limite di 12 ore alla durata della registrazione. Questo significa che ogni database conterrà sempre solo le ultime 12 ore di registrazione. Durante la registrazione vengono rimossi dal database i record più vecchi. La durata limite della storia registrata nel database può essere modificata con una voce in Tpa.ini (per informazioni contattare la TPA).

Questa funzionalità è disponibile per dispositivi fisici su bus GreenBus, EtherCAT, CAN, S-CAN, MECHATROLINK-II, che sono collegati tramite TMSbus, TMSbus+, TMScan, TMScan+, DualMech, DualMech Mono, AlbMech.

ENDBLACKBOX

Sintassi

ENDBLACKBOX

Descrizione

Termina la funzionalità di registrazione su file di tutta l'attività di un modulo locale o remoto. Vedere anche [STARTBLACKBOX](#) e [PAUSEBLACKBOX](#).

PAUSEBLACKBOX

Sintassi

PAUSEBLACKBOX

Descrizione

Sospende la funzionalità di registrazione su file di tutta l'attività di un modulo locale o remoto. Per riprendere la registrazione bisogna eseguire l'istruzione [STARTBLACKBOX](#) senza argomenti. Vedere anche [ENDBLACKBOX](#).

STARTBLACKBOX

Sintassi

STARTBLACKBOX **[valore][,errore]**

Argomenti

valore costante o variabile integer. Periodo di registrazione
errore variabile integer. Codice di errore

Descrizione

Attiva la funzionalità di registrazione su file di tutta l'attività di un modulo locale o remoto. Per attività di un modulo si intende la variazione nel tempo dello stato dei dispositivi logici, ad esclusione dei dispositivi di tipo flag switch.

Il periodo di registrazione (**valore**) è espresso in millisecondi. Non può essere inferiore a 10 e deve essere un multiplo del periodo di real-time. Se non lo fosse verrebbe generato l'errore di sistema n. 4399 (Parametro fuori range).

Se l'istruzione avvia una registrazione e il **valore** è omesso, viene considerato come valore di default 20.

Se l'istruzione riprende una registrazione precedentemente interrotta, non viene considerato alcun **valore** impostato.

Se non è stato possibile far partire la registrazione, **errore** contiene un valore diverso da 0, altrimenti contiene 0:

Codice errore	Descrizione
0	Nessun errore
1	Ci sono differenze tra la configurazione dei dispositivi nel controllo numerico e la configurazione dei dispositivi in Albatros
2	Il numero dei dispositivi da registrare eccede il numero massimo previsto per il sistema
3	In configurazione non sono presenti dispositivi

4	Il software di comunicazione nel modulo remoto non supporta la funzionalità di blackbox (solo moduli remoti)
5	Il controllo numerico impedisce l'avvio della registrazione
6	Errore nel caricamento della libreria di gestione del database
7	Il numero di colonne per la tabella eccede il numero massimo di colonne gestibili dal database
8	Non è stato possibile aprire il database su disco
9	Non è stato possibile creare nel database la tabella delle registrazioni
10	Errore nell'indirizzo IP per la comunicazione col modulo remoto (solo moduli remoti)
11	Non è stato possibile creare il socket di comunicazione per ricevere i dati (solo moduli remoti)
12	Non è stato possibile associare un indirizzo locale al socket di comunicazione (solo moduli remoti)
13	Non è stato possibile collegarsi al socket remoto (solo moduli remoti)
14	Non è stato possibile accedere all'area di memoria condivisa con il controllo numerico
15	La configurazione hardware impedisce l'uso della funzionalità "BlackBox"
16	La funzionalità è stata disabilitata in tpa.ini

Vedere anche [PAUSEBLACKBOX](#) e [ENDBLACKBOX](#).

10.3.20 ISO

ISOGO

Sintassi

ISOGO

etichetta, asse1 quota1, asse2, quota2, asse3, quota3, asse4, quota4, asse5, quota5, [valore]

Argomenti

etichetta

costante o variabile integer. Etichetta che identifica un blocco di spostamento. N nello standard ISO.

asse1

nome di dispositivo tipo asse. (X nello standard ISO)

quota1

costante o variabile. Quota spazio operativo dell'asse1

asse2

nome di dispositivo tipo asse. (Y nello standard ISO)

quota2

costante o variabile. Quota spazio operativo dell'asse2

asse3

nome di dispositivo tipo asse. (Z nello standard ISO)

quota3

costante o variabile. Quota spazio operativo dell'asse3

asse4

nome di dispositivo tipo asse. (C nello standard ISO)

quota4

costante o variabile. Quota spazio dei giunti dell'asse4

asse5

nome di dispositivo tipo asse. (B nello standard ISO)

quota5

costante o variabile. Quota spazio dei giunti dell'asse5

valore

costante o variabile double. Rappresenta la percentuale di feed rate. (F in Standard ISO)

Descrizione

Imposta il movimento in Rapido. I tratti di movimento in rapido sono gestiti in sincronizzazione. I punti definiti dall'utente sono i punti estremanti del singolo tratto di spostamento, che viene percorso, in modo che gli assi siano tra loro sincronizzati. Ciò significa che gli assi fisici si muovono indipendentemente l'uno dall'altro mantenendo però la contemporaneità della partenza e dell'arrivo, similmente a quello che avviene nelle istruzioni [MULTIABS](#) e [MULTIINC](#). La punta dell'utensile non percorre una linea nello spazio operativo e la sua traiettoria non risulta controllata. Il parametro **etichetta** è utilizzato in associazione alla istruzione [SETLABELINTERP](#) per identificare in modo univoco il blocco di spostamento. Le prime tre **quote** identificano la posizione della punta nello spazio operativo mentre le successive due definiscono il valore degli assi rotanti nello spazio dei giunti. Il **valore** di FeedRate definisce la percentuale di riduzione rispetto alla velocità massima possibile (In ISO: F0 massima velocità, F100 FeedRate nullo e quindi assi fermi).

L'istruzione genera errore di sistema (4105- Istruzione non eseguibile sull'asse NomeAsse) se utilizzata su assi passo-passo.

Non è possibile utilizzare l'istruzione [WAITCOLL](#), in quanto dal momento della collisione in avanti si perderebbe il legame di interpolazione con gli altri assi che concorrono al movimento, generando un profilo diverso da quello atteso. Se utilizzata viene generato l'errore di sistema 4101 - Gestione incongruente dell'asse NomeAsse.

ISOG1

Sintassi

ISOG1

etichetta, asse1, quota1, asse2, quota2, asse3, quota3, asse4, quota4, asse5, quota5, [valore]

Argomenti

Imposta l'interpretazione delle quote come quote relative. Il parametro **asse** individua il canale di interpolazione 5 Assi che da questa istruzione in poi interpreterà le quote degli assi come quote relative. In questo caso non vi è differenza se si sta eseguendo una [ISOG1](#) o una istruzione [ISOG0](#).

ISOG93

Sintassi

ISOG93 **asse**

Argomenti

asse nome di dispositivo tipo asse

Descrizione

Imposta l'interpretazione delle velocità come inverso del tempo di esecuzione. Il parametro **asse** individua il canale di interpolazione 5 Assi che da questa istruzione in poi interpreta il valore passato tramite il parametro F della istruzione [ISOG1](#) come inverso del tempo di esecuzione espresso in minuti. Grazie a questo il controllo è in grado di determinare la velocità che deve essere tenuta dalla punta dell'utensile nei blocchi di interpolazione.

ISOG94

Sintassi

ISOG94 **asse**

Argomenti

asse nome di dispositivo tipo asse

Descrizione

Imposta l'interpretazione delle velocità come unità di misura al minuto. Il parametro **asse** individua il canale

ISOG216

Sintassi

ISOG216 **nomeMatriceRotanti, nomeMatricePortaUtensili, nomeMatriceUtensili, mascheraAbilitazione,asse1, asse2,asse3,asse4,asse5**

Argomenti

nomeMatriceRotanti	nome della matrice. Contiene i dati relativi agli assi rotanti
nomeMatricePortaUtensili	nome della matrice. Contiene i dati relativi al porta utensili
nomeMatriceUtensili	nome della matrice. Contiene i dati relativi agli utensili
mascheraAbilitazione	variabile o costante integer. Maschera di abilitazione degli assi C e B
asse1	nome di dispositivo tipo asse. (X nello standard ISO)
asse2	nome di dispositivo tipo asse. (Y nello standard ISO)
asse3	nome di dispositivo tipo asse. (Z nello standard ISO)
asse4	nome di dispositivo tipo asse. (C nello standard ISO)
asse5	nome di dispositivo tipo asse. (B nello standard ISO)

Descrizione

Identifica le tre matrici per la parametrizzazione della macchina e i cinque dispositivi di tipo asse che costituiscono la stessa. Tale istruzione **deve** essere eseguita prima di ogni altra istruzione ISO. Il parametro **mascheraAbilitazione** definisce quali assi rotanti (C e/o B) abilitare.

Per i valori da impostare riferirsi alla tabella seguente:

mascheraAbilitazione	Descrizione
31	Abilitazione assi C e B
23	Abilitazione del solo asse B
15	Abilitazione del solo asse C
7	Disabilitazione assi C e B

Nota

L'unità di misura in cui sono espressi, in configurazione, i valori degli assi rotanti, deve essere gradi.

Il legame tra gli assi fisici e gli assi ISO virtuali impostato tramite questa istruzione viene sciolto nel momento in cui viene terminato il task, in cui è definita l'istruzione oppure tramite l'istruzione [ISOM2](#). Gli assi possono quindi essere utilizzati per movimenti di tipo classico.

Offset Y	Offset lungo Y tra il fulcro e il punto di comando della testa
Offset Z	Offset lungo Z tra il fulcro e il punto di comando della testa
Disassamento X	Scostamento in X tra l'asse di rotazione e quello di brandeggio (quando la posizione dell'asse C =0)
Disassamento Y	Scostamento in Y tra l'asse di rotazione e quello di brandeggio (quando la posizione dell'asse C =0)
Disassamento Z	Distanza Naso-fulcro
Angolo δ	Angolo attorno a Z per il posizionamento corretto della testa rispetto allo zero macchina
Angolo γ	Angolo tra il piano di rotazione e quello di brandeggio

Campi Matrice	Matrice PortaUtensili
OffsetPU X	Offset in X tra il punto di aggancio del portautensile al motore ed il punto di aggancio dell'utensile al portautensile (quando la posizione dell'asse C =0 e motore verticale)
OffsetPU Y	Offset in Y tra il punto di aggancio del portautensile al motore ed il punto di aggancio dell'utensile al portautensile (quando la posizione dell'asse C =0 e motore verticale)
OffsetPU Z	Offset in Z tra il punto di aggancio del portautensile al motore ed il punto di aggancio dell'utensile al portautensile (quando la posizione dell'asse C =0 e motore verticale)
Angolo α	Angolo di sfasamento tra l'asse del motore e quello dell'utensile (rispetto a Z)
Angolo β	Angolo di sfasamento tra l'asse del motore e quello dell'utensile (rispetto a Y)

Campi Matrice	Matrice Utensili
Lunghezza utensile	Lunghezza dell'utensile

ISOSETPARAM

Sintassi

ISOSETPARAM

IdentificativoParametro, valore

Argomenti

**IdentificativoParametro
valore**

costante o variabile integer. È il numero che individua un parametro
costante o variabile float. È il valore da impostare.

Descrizione

Imposta alcuni parametri caratterizzanti la fluidità del movimento interpolato ISO. La tabella seguente spiega il significato di ogni **IdentificativoParametro**, i valori entro cui deve essere compresa la variabile **valore** e i valori di default.

IdentificativoParametro	Range	Default	Significato
0	0.0-100.0	50.0	Percentuale di rallentamento sugli assi lineari in caso di punto angoloso. (0=nessun rallentamento, 100=massimo del rallentamento permesso dall'interpolatore)
1	0.0-100.0	50.0	Percentuale di rallentamento sugli assi rotanti in caso di punto angoloso. (0=nessun rallentamento, 100=massimo del rallentamento permesso dall'interpolatore)
2	0.5-1.0	0.9	Fattore di riduzione velocità su ascissa curvilinea nel caso di punto angoloso. (1=nessuna riduzione, 0.5=massima riduzione permessa)
3	0.0-100.0	60.0	Percentuale di rallentamento in caso di discontinuità ravvicinate. (0=nessun rallentamento, 100=massimo del rallentamento permesso dall'interpolatore)

4	0.0-100.0	10.0	Percentuale di smooth della traiettoria
5	(Maggiore di 0.0)-1.0	0.2	Dimensione minima del tratto da percorrere con soli assi lineari. Il valore è espresso in millimetri.
6	(Maggiore di 0.0)-1.0	0.1	Dimensione minima del tratto da percorrere con soli assi rotanti. Il valore è espresso in gradi
7	0.0-100.0	100.0	Limite inferiore del filtro di smooth
8	1.0-100.0	1.0	Fattore moltiplicativo applicato alle accelerazioni e alle decelerazioni definite in configurazione. Incrementa l'accelerazione e la decelerazione massime dei soli assi lineari. Valori esterni all'intervallo generano l'errore di sistema 4399 - Parametro fuori range.
9	1.0-100.0	1.0	Fattore moltiplicativo applicato alle accelerazioni e decelerazioni definite in configurazione. Incrementa l'accelerazione e la decelerazione massime dei soli assi rotanti. Valori esterni all'intervallo generano l'errore di sistema 4399 - Parametro fuori range.
10	0.0-1.0	0.0	Flag per abilitare (0.0) o disabilitare (1.0) la riduzione di velocità tra blocchi consecutivi nel caso di punto angoloso. La disabilitazione può essere ottenuta anche utilizzando l'istruzione con i seguenti parametri: ISOSETPARAM 0 0.0 ISOSETPARAM 1 0.0 ISOSETPARAM 2 1.0

KINEMATICEXPR

Sintassi

KINEMATICEXPR **asse = espressione**

Argomenti

asse nome di dispositivo tipo asse fisico o virtuale
espressione insieme di operatori

Descrizione

Permette la definizione di singole espressioni di cinematica diretta ed inversa. Prima di eseguire questa istruzione deve essere chiamata l'istruzione [ISOG217](#) che descrive gli assi fisici e gli assi virtuali che compongono la macchina. Per ogni asse definito nella [ISOG217](#) deve essere chiamata l'istruzione KINEMATICEXPR. L'espressione di cinematica di un asse nello spazio dei giunti (cinematica inversa) può essere funzione di variabili, di costanti e delle quote degli assi nello spazio operativo. L'espressione di cinematica di un asse nello spazio operativo (cinematica diretta) può essere funzione di variabili, di costanti e delle quote degli assi nello spazio dei giunti.

La sintassi dell'**espressione** è la stessa descritta nell'istruzione [EXPR](#) con la differenza che non possono essere utilizzate variabili locali. Inoltre non si possono utilizzare assi dello stesso tipo dell'asse dichiarato in **asse** e non dichiarati nella istruzione [ISOG217](#). Per esempio se si sta definendo la cinematica di un asse virtuale, già dichiarato nella istruzione [ISOG217](#), nell'espressione possono essere utilizzati solo i cinque assi fisici dichiarati nella [ISOG217](#).

Esempio

```
ut as double ; numero utensile
offsety as double ; offset Y naso fulcro
offsetz as double ; offset Z naso fulcro
```

Function ISO5Ax

```

setval 100,ut
setval 120.0,offsety
setval 60.0,offsetz
; CINEMATICA ESPLICITA
ISO217 Rx Ry Rz Rc Rb X Y Z C B

; DEFINIZIONE DELLE ESPRESSIONI DI CINEMATICA
; CINEMATICA ESPLICITA INVERSA ASSE fisico Rx
KinematicExpr Rx = X - 135 + ut * sin ( B ) * cos ( C )

; CINEMATICA ESPLICITA INVERSA ASSE fisico Ry
KinematicExpr Ry = Y + offsety + ut * sin ( B ) * sin ( C )

; CINEMATICA ESPLICITA INVERSA ASSE Rz
KinematicExpr Rz = Z + offsetz + ut * cos ( B )

; CINEMATICA ESPLICITA INVERSA ASSE Rc
KinematicExpr Rc = C

; CINEMATICA ESPLICITA INVERSA ASSE Rb
KinematicExpr Rb = B

; CINEMATICA ESPLICITA DIRETTA ASSE X
KinematicExpr X = Rx + 135 - ut * sin ( Rb ) * cos ( C )

; CINEMATICA ESPLICITA DIRETTA ASSE Y
KinematicExpr Y = Ry - offsety - ut * sin ( Rb ) * sin ( C )

; CINEMATICA ESPLICITA DIRETTA ASSE Z
KinematicExpr Z = Rz - offsetz - ut * cos ( Rb )

; CINEMATICA ESPLICITA DIRETTA ASSE C
KinematicExpr C = Rc

; CINEMATICA ESPLICITA DIRETTA ASSE B
KinematicExpr B = Rb

; MOVIMENTO
ISO200 1001,X 998.0,Y 600.0,Z 0.0,C 90.0,B 45.0,0.0
ISO201 1001,X 998.0,Y 600.0,Z 0.0,C 90.0,B 45.0,1000.0
ISO202 1001,X 998.0,Y 600.0,Z 0.0,C 90.0,B 45.0,1000.0
ISO203 1003,X 996.0,Y 600.0,Z 0.0,C 90.0,B 45.0,1000.0
WAITSTILL X
ISO204 X

```

FRet

10.3.21 Istruzioni non più disponibili

INPBCD	legge una serie di nibble digitali in formato BCD
OUTBCD	modifica una serie di nibble digitali in formato BCD
SETFORCEDBCD	forza una serie di nibble in formato BCD
CANOPENDRIVER	apre un canale di comunicazione CANopen
CANCLOSEDRIVER	chiude un canale di comunicazione CANopen
CANRESETBOARD	esegue il reset di una scheda CANopen
CANSETOBJECT	scrive un oggetto CANopen
CANGETOBJECT	legge un oggetto CANopen
SLMCOMMAND	esegue un comando SLM
SLMEEPROMDISABLE	esegue un comando di disabilitazione alla scrittura EEPROM
SLMEEPROMENABLE	esegue un comando di abilitazione alla scrittura EEPROM
SLMGETEEPROM	legge una locazione di memoria EEPROM
SLMGETPARAM	legge un parametro SLM
SLMGETREGISTER	legge un registro SLM

SLMGETSTATUS	legge una grandezza dell'azionamento
SLMSETEEPROM	scrive una locazione di memoria EEPROM
SLMSETPARAM	imposta un parametro SLM
SLMSETREGISTER	imposta un registro SLM
HOMING	esegue la "ricerca dello zero"
SYNCRROOPEN	apre un canale di movimentazione sincronizzata
SYNCRROCLOSE	chiude in canale di movimentazione sincronizzata
SYNCRROMOVE	assegna un punto di movimento sincronizzato
SYNCRROSETACC	imposta l'accelerazione per i movimenti sincronizzati
SYNCRROSETDEC	imposta la decelerazione per i movimenti sincronizzati
SYNCRROSETVEL	imposta la velocità per i movimenti sincronizzati
SYNCRROSETFEED	scala la velocità degli assi in un movimento sincronizzato
SYNCRROSTARTMOVE	inizia l'elaborazione di un movimento sincronizzato
GETVf	legge il valore del convertitore tensione/frequenza

10.3.22 Istruzioni non utilizzabili su evento

Le seguenti istruzioni non possono essere utilizzate nelle funzioni che vengono richiamate dalle istruzioni [ONFLAG](#), [ONINPUT](#) e [ONERRSYS](#). Il loro utilizzo è vietato anche nei [task real-time](#).

Istruzioni che a loro volta effettuano una chiamata a funzione su interrupt:

- ONFLAG
- ONINPUT
- ONERRSYS

Istruzioni che comportano attesa:

- WAITINPUT
- WAITFLAG
- WAITACC
- WAITCOLL
- WAITDEC
- WAITREG
- WAITTARGET
- WAITWIN
- WAITSTILL
- WAITTASK
- WAITRECEIVE
- WAITPERSISTINPUT
- MULTIWAITFLAG
- MULTIWAITINPUT

Istruzioni di comunicazione:

- SEND
- RECEIVE
- CLEARRECEIVE
- COMOPEN
- COMCLOSE
- COMREAD
- COMREADSTRING
- COMWRITE
- COMWRITESTRING
- COMGETERROR
- COMCLEARRXBUFFER
- COMGETRXCOUNT

Le seguenti istruzioni legate al movimento degli assi:

- MOVINC
- MOVABS
- LINEARINC
- LINEARABS
- CIRCLE
- CIRCINC
- CIRCABS
- HELICINC
- HELICABS
- COORDIN

- MULTIABS
- MULTIINC
- SETRIFLOC
- SETTOLERANCE
- RESRIFLOC
- SETPFLY
- SETPZERO
- SETINDEXINTERP
- STARTINTERP
- FASTREAD
- ENABLE
- DISABLE
- ENDMOV

Istruzioni ISO:

- ISOG0
- ISOG1
- ISOG9
- ISOG90
- ISOG91
- ISOG93
- ISOG94
- ISOG216
- ISOG217
- ISOM2
- ISOM6
- ISOSETPARAM
- KINEMATICEXPR

Istruzioni Blackbox:

- ENDBLACKBOX
- PAUSEBLACKBOX
- STARTBLACKBOX

Le seguenti istruzioni legate alla gestione EtherCAT:

- READDICTIONARY
- WRITEDICTIONARY

Istruzioni legate al multitasking:

- SENDMAIL
- WAITMAIL
- ENDMAIL
- SENDIPC
- WAITIPC
- TESTMAIL
- TESTIPC

Istruzioni che comportano elevato tempo di elaborazione:

- SORT
- FIND
- FINDB
- MOVEMAT

10.4 Esempi

10.4.1 Azzeramento su Interrupt

```

;-----
; Esempio di routine di azzeramento al volo
;
; La function esegue le seguenti operazioni:
;
; 1) Imposta l'asse disabilitando i limiti software
;    e azzerando la quota
; 2) verifica che il sensore non sia già in stato ON.
;    Se è in stato ON muove l'asse e attende che torni
;

```

```

;   in stato OFF. Se ciò non avviene entro 30 secondi
;   genera un messaggio di errore.
;   3) Imposta la velocità di ricerca sensore
;   4) Lancia un movimento dell'asse e abilita l'azzeramento
;   al volo per il detto asse. Allo scattare dell'interrupt
;   la quota dell'asse viene azzerata e parte in automatico
;   un movimento ad una quota di disimpegno.
;   5) Attende l'arrivo dell'asse alla quota di disimpegno
;   6) Ripristina i limiti asse
;
;
;   © TPA.
-----
Function Azzeramento_Veloce

    ResLimPos   asse           ; Inizializzazione Asse
    ResLimNeg   asse
    SetQuote    asse,0

    IfInput     FastInput,OFF,Goto Continue    ; Test Sensore occupato
    SetVel      asse,5                        ; Imposta velocità di
                                           ; disimpegno
    MovAbs      asse,30                       ; muove l'asse
    WaitInput   FastInput,OFF,30,Call Error    ; Controllo disimpegno
                                           ; micro,
                                           ; errore dopo TimeOut=30

    EndMov      asse                        ; Stop asse
    WaitStill   asse                        ; Attesa asse fermo

Continue:
    SetVel      asse,10                       ; Velocità di ricerca sensore di azzeramento
    MovAbs      asse,-1000                    ; Movimento in Negativo di ricerca sensore
    SetPFly     asse,ON,10,0                 ; Aggancio Interrupt
                                           ; e impostazione quota e velocità di
                                           ; disimpegno
    WaitStill   asse                          ; Attesa asse Fermo

    SetLimPos   asse                          ; Ripristino limiti asse
    SetLimNeg   asse

    Fret

; sottoprocedura di invio messaggio di errore
Error:
    Error       ERR_SETP                      ;Segnalazione di errore impossibile procedere
    Ret

```

10.4.2 Server di movimentazione assi

```

-----
; Esempio di un server di movimentazione assi:
;
; Il server movimentata gli assi della macchina
; per conto di altri task.
;
; I task client inviano i comandi sotto forma di
; messaggi (mail) ad una casella di posta.
;
; Il server preleva i comandi dalla casella e li esegue.
;
; Le richieste vengono accodate nella casella, per cui
; se una richiesta arriva mentre il server è già

```

```

; impegnato, non viene persa, ma verrà evasa appena possibile.
;
; Il server è l'unico task a movimentare gli assi. Si evitano così
; i conflitti.
;
; Il server è implementato dalla funzione Master_assi.
;
; Un esempio di client è implementato dalla funzione Check_flag.
; Questa funzione verifica periodicamente lo stato di
; un flag e quando lo trova in stato ON invia al server
; il comando di esecuzione azzeramento assi.
; Il flag verrà presumibilmente impostato ad ON manualmente
; dall'operatore, utilizzando ad esempio un quadro sinottico.
;
;-----
;-----
; -- COSTANTI GLOBALI DI MACCHINA --
;-----
Const MBOX = 101 ; identifica la casella postale
; comandi
Const SETP = 10 ; azzeramento assi
Const CHG = 11 ; cambio utensile
Const FORO = 12 ; esecuzione di una foratura

;-----
; --- GRUPPO ASSI ---
;-----

; definizione dei messaggi di errore
Defmsg ERR_CMD "Comando gruppo assi sconosciuto"

; --- Server ---
Function Master_assi autorun

    local cmd as integer ; comando
    local quota_X as double ; quota X foro
    local quota_Y as double ; quota Y foro

loop:
    waitmail MBOX,cmd,quota_X,quota_Y ; attesa comando

    ; Una volta ricevuto il comando lo identifichiamo
    ; ed eseguiamo l'azione opportuna
    Select cmd

    case SETP
        fcall azzeramento_assi ; azzeramento degli assi
    case CHG
        fcall Cambio_utensile ; Esegue un cambio utensile
    case FORO
        fcall Foratura quota_X,quota_Y ; foratura alle quote
        ; specificate
    case else
        call errore
    endselect

    endmail MBOX ; notifica esecuzione comando
    goto loop ; torna in attesa di un nuovo
; comando

```

```

fret

; sottoprocedura di invio messaggio di errore
errore:
    error ERR_CMD
    ret

;-----
; --- GENERICO GRUPPO ---
;-----

; --- Client ---
Function Check_flag

Loop:

    ifflag Setp_assi,OFF, goto loop          ; test stato flag

    ; OK il flag è ad ON, invio il comando
    sendmail MBOX,WAITTACK,SETP,0.0,0.0

    resetflag Setp_assi                    ; resetto il flag

    goto loop                               ; torno in attesa

fret

; NOTARE CHE:
; - dopo il comando "SETP", devo specificare
;   i due parametri "quota_X" e "quota_Y" anche se ciò
;   non ha senso per l'operazione di azzeramento.
;   Il server tuttavia non può sapere a priori quale comando
;   gli verrà inviato, dobbiamo quindi specificare due valori
;   dello stesso tipo di quelli che il server si aspetta di ricevere
;   in questo caso due DOUBLE. I valori passati sono "0.0" e "0.0".
; - il parametro "WAITACK" fa in modo che il client rimanga in attesa
;   dell'avvenuta esecuzione del comando da parte del server.
;   Il client proseguirà la propria esecuzione solo quando il server
;   avrà eseguito una ENDMAIL o avrà iniziato a processare un nuovo
;   comando (WAITMAIL).

```

10.4.3 Ciclo di Main con gestione errori

```

;-----
; Ipotetica funzione principale
; inizializza la macchina ed esegue un ciclo di controllo
;-----
Function Main
    OnErrSys GestErrSys          ; abilita gestione errori

    StartTask Emergenze          ; inizializza
    StartTask Elaboratore
    AbilitaAssi

Loop:
    IfFlag Flag,OFF, RipristinoEmerg
    .....
    Goto loop
Fret

```



```

;-----
; funzione di gestione degli errori
;-----
Function GestErrSys
  Param nerrore as integer
  Param task as function
  Param tipodevice as device

  EndTask      Elaboratore          ; Termina task
  If           nerrore, >, 5, goto noerraxis ; Elaboratore
  ; I primi 5 errori
  ; sono relativi
  ; agli assi

  ResetFlag   Flag
  DisabilitaAssi

noerraxis:
Fret

```

10.4.4 Operazioni sulle stringhe

```

;-----
; Esempio di manipolazione delle stringhe
;-----
Function esempio
  Local stringa1 as string
  Local stringa2 as string
  Local stringa3 as string
  Local lunghezza as integer
  Local posizione as integer

  SetString "Stringa di",stringa1 ; stringa1 contiene
  ; ora "Stringa di"

  SetString " prova",stringa2

  AddString stringa1,stringa2,stringa3 ; stringa3 contiene
  ; "Stringa di prova"

  Search stringa3,'t',posizione ; posizione vale 2
  Search stringa3,'Z',posizione ; posizione vale -1

  Left stringa3,7,stringa1 ; stringa1
  ; contiene "Stringa"

  Right stringa3,2,stringa2 ; stringa2
  ; contiene "va"

  Mid stringa3,9,2,stringa3 ; stringa3
  ; contiene "di"

  ControlChar 65,stringa1 ; stringa1
  ; contiene "A"

  Len stringa3,lunghezza ; lunghezza vale 2

  Str lunghezza,stringa3 ; stringa3
  ; contiene "2"

  Val posizione,stringa1 ; stringa1
  ; contiene "-1"

```

```
AddString "Il risultato è ",stringa1,stringa2
; stringa2 contiene "Il risultato è -1"
```

Fret

10.4.5 Esecuzione Sequenziale / Parallela

```
-----
; Esempio di routine che controlla l'azzeramento
; di una macchina a 3 assi evitando eventuali
; interferenze meccaniche.
;
; Gli azzeramenti dei singoli assi sono implementati
; da delle funzioni il cui testo è omoesso.
; Vedere l'esempio "Routine di azzeramento".
;
; Viene effettuato prima il solo azzeramento
; dell'asse Z (che ipotizziamo non possa essere
; effettuato contemporaneamente agli altri),
; terminato questo vengono eseguiti
; contemporaneamente gli azzeramenti degli assi
; X e Y.
-----
; messaggio per l'operatore (tradotto in lingua)
DefMsg      MSG_SETP   ITA  "Azzeramento assi in corso ..."
            ENG  "Homing in progress ..."

Function AzzeramentoAssi

Message     MSG_SETP           ; informa l'operatore

Fcall      AzzeramentoAsseZ   ; azzeramento dell'asse Z

; OK l'azzeramento asse Z è terminato

StartTask  AzzeramentoAsseX   ; lancia azzeramento X e Y
StartTask  AzzeramentoAsseY

waitTask   AzzeramentoAsseX   ; attende il completamento
waitTask   AzzeramentoAsseY

DelMessage MSG_SETP           ; cancella il messaggio
; per l'operatore
```

Fret

10.4.6 Routine di Azzeramento

```
-----
; Esempio di routine di Azzeramento di un asse
;
; La function esegue le seguenti operazioni:
; 1) disabilita i limiti software dell'asse
; 2) imposta la velocità di ricerca switch
; 3) muove l'asse ad una quota incrementale che
;    garantisca il raggiungimento dello switch
; 4) attende che l'asse faccia scattare lo switch
; 5) ferma l'asse e attende la fine del movimento
; 6) imposta la velocità (bassa) di disimpegno switch
; 7) fa indietreggiare l'asse di quanto basta
;    per disimpegnare lo switch
```

```

; 8) attende il disimpegno dello switch
; 9) assegna la nuova quota all'asse
; 10) ripristina la velocità di default ed i limiti software
;
; © TPA.
;-----

```

Function Azzeramento

```

ResLimPos  asse      ; disabilita i limiti software
ResLimNeg  asse

SetVel     asse,10   ; imposta la velocità

MovInc     asse,10000 ; muove l'asse

WaitInput  Switch,ON ; attesa switch

EndMov     asse      ; ferma l'asse
waitStill  asse      ; attesa asse fermo

SetVel     asse, 0.1 ; imposta velocità disimpegno

MovInc     asse,-100 ; muove l'asse

WaitInput  Switch,OFF ; attesa disimpegno switch

SetQuote   asse,0    ; assegna la nuova quota

SetVel     asse      ; ripristino velocità
SetLimpos  asse      ; ripristino limiti software
SetLimneg  asse

```

Fret

10.4.7 Movimenti ISO

```

;-----
; Esempio di movimento ISO
;
; Viene generato un profilo mediante le istruzioni ISOG0 e ISOG1
;
; © TPA.
;-----*

```

```

; Dichiarazione delle matrici ISO
; Matrice assi rotanti
MxRot[5] as double:off_X double:off_Y double:off_Z double:dis_X
double:dis_Y double:dis_Z double:delta double:gamma
; Matrice portautensile
MxPorta[1] as double:off_X double:off_Y double:off_Z double:alpha
double:beta
; Matrice utensili
MxUtensili[10] as double:ut double

```

Function InterpolazioneISO

```

; impostazione dei valori standard di parametrizzazione macchina
setval 90.0 MxRot[5].gamma
setval 260.3 MxUtensili[10].ut
setval MxUtensili[10].ut ut

```

```

; impostazione dei parametri dell'algoritmo
IsoiParam 0 50
IsoiParam 1 50
IsoiParam 2 0.9
IsoiParam 3 60
IsoiParam 4 30

; impostazione della macchina: dichiara le tre matrici
; utilizzate per la parametrizzazione della macchina e
; gli assi fisici utilizzati nei movimenti ISO
isoG216 MxRot MxPorta MxUtensili 31 X Y Z C B ; CINEMATICA IMPLICITA

; impostazione del gruppo di parametri che descrivono
; la cinematica della macchina
isoM6 X 5 1 10 ;CINEMATICA IMPLICITA

; impostazione dei valori iniziali
setquote x 500
setquote y 300
setquote z 0
setquote c 0
setquote b 0
setvel x
setvel y
setvel z
setvel c
setvel b
setveli x y z c b

; esecuzione del profilo
isoG0 1001,X 998.0,Y 600.0,Z 0.0,C 90.0,B 45.0,50.0
isoG1 1001,X 998.0,Y 600.0,Z 0.0,C 90.0,B 45.0,10000.0
isoG1 1003,X 996.0,Y 600.0,Z 0.0,C 90.0,B 45.0,10000.0
isoG1 1002,X 600.0,Y 600.0,Z 0.0,C 90.0,B 45.0,10000.0
isoG1 1004,X 599.131759111665,Y 599.924038765061,Z 0,C 100,B
45.0,10000.0
isoG1 1006,X 598.289899283372,Y 599.69846310393,Z 0,C 110,B 45.0,10000.0
isoG1 1005,X 597.5,Y 599.330127018922,Z 0,C 120,B 45.0,10000.0
isoG1 1003,X 596.786061951567,Y 598.830222215595,Z 0,C 130,B
45.0,10000.0
isoG1 1002,X 596.169777784405,Y 598.213938048433,Z 0,C 140,B
45.0,10000.0
isoG1 1012,X 595.669872981078,Y 597.5,Z 0,C 150,B 45.0,10000.0
isoG1 1011,X 595.301536896071,Y 596.710100716628,Z 0,C 160,B
45.0,10000.0
isoG1 1031,X 595.075961234939,Y 595.868240888335,Z 0,C 170,B
45.0,10000.0
isoG1 1102,X 595.0,Y 0.0,Z 0.0,C 180.0,B 45.0,10000.0
waitstill X Y Z C B
fret

```



Tecnologie e Prodotti per l'Automazione S.r.l.

Via Carducci 221
I - 20099 Sesto S.Giovanni
(MI)
Tel. +39 02.36527550
Fax. +39 02.2481008
www.tpaspa.com