

Project ASSE_Z_ASM

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		1/31
	field2		

Table of contents

Header Page	1
Table of contents	2
1. ASSE_Z_ASM	3
1.1. Project description	3
1.2. Hardware architecture.....	3
1.2.1. Hardware view	3
1.2.2. Hardware description	4
1.3. Link architecture.....	4
1.4. DPL programs.....	5
1.4.1. CT-RTU_Node_1	6
1.4.2. APP1	7

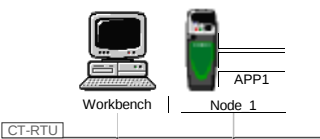
	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		2/31

1. ASSE_Z_ASM

1.1. Project description

1.2. Hardware architecture

1.2.1. Hardware view



1.2.2. Hardware description

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		3/31
	field2		

Networks: **CT-RTU:**

Type: CT-RTU

Long Name: CT-RTU

Nodes: **CT-RTU_Node_1:**

Address: 1

Short Name: Node_1

Title: Node_1

Network: CT-RTU

Version: V1.0

Author: My Name

Company: My Company

Drive: Unidrive SP

Target: SPOB

Embed debug information: YES

Embed source file: YES

UD7X extended memory: NO

User programming enabled: NO

DPL programming not supported

Cyclic data not supported

SyptLite editing supported

Options: **APP1:**

Address: 0

Short Name: APP1

Title: An07 Posizionatore assoluto

Network:

Version: V2.1

Author: Control Techniques

Company: Control Techniques spa

Drive: SM-Applications

Target: SMAPPL

Embed debug information: YES

Embed source file: YES

UD7X extended memory: NO

User programming enabled: YES

DPL programming supported

Cyclic data supported

SyptLite editing not supported

Option : 0

1.3. Link architecture

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		4/31

1.4. DPL programs

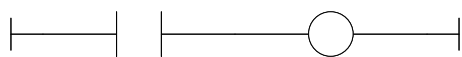
	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		5/31
	field2		

1.4.1. CT-RTU_Node_1

AUTHOR My Name
COMPANY My Company
TITLE Node_1
VERSION V1.0
TARGET SPOB
DRIVE Unidrive SP

DESCRIPTION:
No description

(* Double-click to enter a rung comment *)



	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		6/31

1.4.2. APP1

AUTHOR Control Techniques
COMPANY Control Techniques spa
TITLE An07 Posizionatore assoluto
VERSION V2.1
TARGET SMAPPL
DRIVE SM-Applications

DESCRIPTION:
No description

Notes {
Application Note SA07_a22

S = per Unidrive Sp + Sm Application module
A07 = codice applicazione
_abs = assoluto
22 = versione aggiornamento (0,1,2.....a,b,c....z)

Questo applicativo e' utilizzabile in tutte le applicazioni che richiedono dei movimenti assoluti rispetto alla posizione di zero macchina. I comandi possono essere inviati sia tramite ingressi / uscite digitali che tramite Bus di campo o via seriale.
Le funzionalita' proposte sono:

Ricerca di zero
Movimentazione manuale
Posizionamento
Fine corsa software (non presenti nella A21)

Menu 18
#18.11=3000 // vel. massima in RPM (posizionamenti)
#18.12=1000 // quota avanzamento in unita'
#18.13=1 // parte intera unita' di avanzamento per giro motore
#18.14=0 // parte decimale unita' di avanzamento per giro motore (4 cifre piu' significative-formato fisso)
#18.15=400 // accelerazione da 0 a vel max (#18.11) in millisecondi
#18.16=400 // decelerazione da 0 a vel max (#18.11) in millisecondi
#18.17=500 // vel. movimentaz. manuale
#18.18=0 //
#18.19=9000 // guadagno proporzionale di posizione
#18.20=1000 // guadagno feed forward

#18.26=-1000 // Velocita' di nricerca sensore di zero

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		7/31
	field2		

```

#18.27=10          // Velocita' di uscita / rientro dal / sul sensore di zero
#18.28=30    // finestra di posizionamento in count encoder

#18.30=1024  // impulsi encoder per giro
#18.31=0     // comandi dati da I/O
#18.31=1     // comandi dati via BUS di campo

#18.49=1     // non prevista esecuzione ricerca di zero

Menu 19
#19.09      // Errore di inseguimento
#19.10      // Posizione attuale in unita' (sola lettura)
#19.12      // Quota critica
#19.13      // quinta word da bus
#19.16      // Quota virtuale dopo azzeramento assi (non sei a zero ma a quota #19.16)
#19.49=1    //Abilita ricerca di zero su Index
#19.50=1    // inverte senso di posizionamento

#20.21      // valore di -Q8% (posizione assoluta)acquisita sul fronte di Acq_pos% da Homing
#20.39      // Parola di comando da Bus di Campo
#20.40      // Status Word

} //Notes

```

```

// =====
//      USER DEFINED FUNCTION BLOCK
// =====
- Link To Extra Help:
- Make Visible to Library User: Yes
- Function Block Description: No description
_Init_SM () {

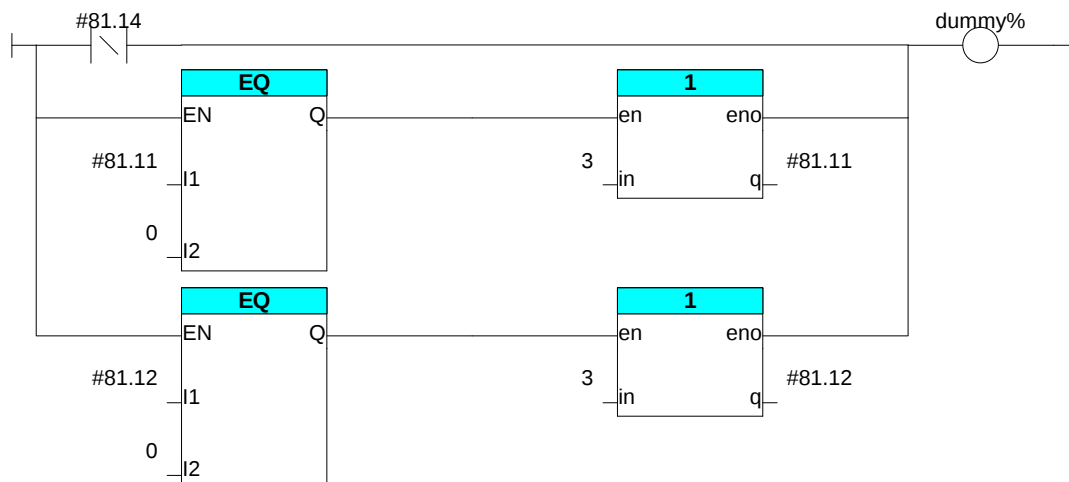
```

FBbody

Diagram 1:

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		8/31

(* *)



```
p0%=0
p1%=0
#1.10=1      // abilita bidirezionale
#1.14=3
#2.02=0
#2.04=0
#4.07=#4.05

#6.04=4      // Disabilita logiche di start - stop da input digitali

#90.44=0     // encoder di feedback da connettore drive
IF #15.01 =102 THEN #90.43=1
IF #16.01 =102 THEN #90.43=2
IF #17.01 =102 THEN #90.43=3

#91.1 = 3
#91.02=0

#8.10=0      // Terminale 31 = Drive Enable
#8.29=1      // logica positiva
#8.30=0      // resistenza di polarizzazione inserita
#8.39=1

// Impostazione I/O digitali
IF #18.31=0 THEN
    #8.31=1    // term 24 Output
    #8.32=1    // term 25 Output
    #8.33=0    // term 26 Input
    #8.21=19.31
```

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		9/31
	field2		

```

#8.22=19.32
#8.23=19.33
#8.24=19.34
#8.25=19.35
#8.26=19.36
ELSE
#8.31=1      // term 24 Output
#8.32=0      // term 25 Input
#8.33=0      // term 26 Input
#8.21=19.31
#8.22=19.32
#8.23=19.33
#8.24=19.34
#8.25=19.35
#8.26=19.36
ENDIF

#81.21=1
#81.20=1    // autosave menu 20

#6.30=1

#85.01=137 // schedula la task even su transizione 0-1 su Dig In0
#91.05=#1.06

} // _Init_SM ()

- Link To Extra Help:
- Make Visible to Library User: Yes
- Function Block Description: No description
_Init_UD70 () {
// initialisation code

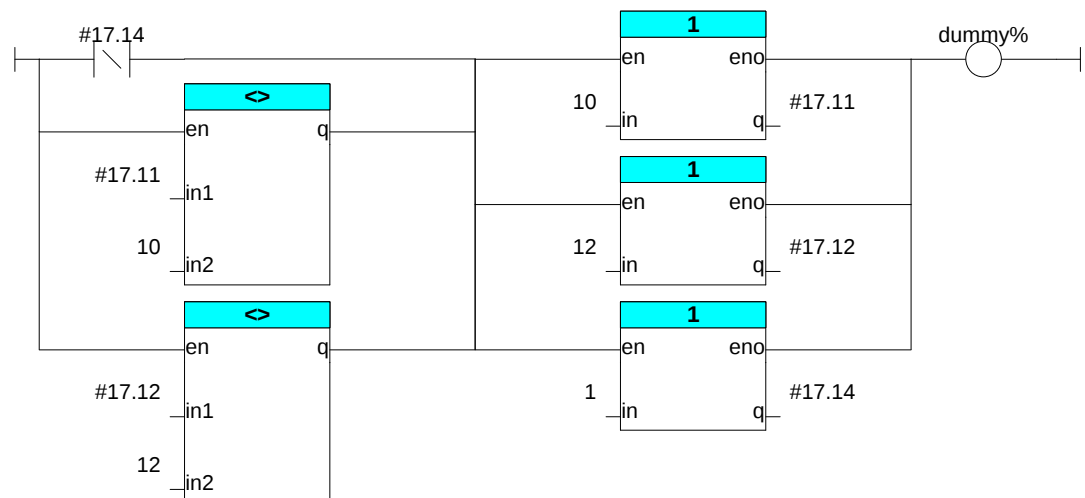
FBbody

Diagram 2:

```

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		10/31

(* *)



```
#1.10=1
#1.14=3
#2.02=0
#2.04=1
#6.30=1

#91.02=0

#8.27=1
#17.20=1    // autosave menu 20
```

```
#8.10=19.01
#8.13=19.01
#8.16=19.01
#8.19=19.01
#8.21=19.01
#8.23=19.01
```

```
// Reset Internal Variables and Initialise system constants
```

```
Q2%=0
Q3%=0
_Q1%=0
_Q8%=0
Q9%=_Q8%
```

```
// Setup the system parameters and Make sure that the position controller output is OFF
```

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		11/31

```

_Q20%.6 = 0
#6.30=1
#91.1 = 3

```

```

} // _Init_UD70 ()

```

- Link To Extra Help:
- Make Visible to Library User: Yes
- Function Block Description: No description

```

_load_default () {
// initialisation code

```

```

// Menu 18

```

```

IF #18.11=0 THEN

```

```

    #18.11=3000 // vel. massima in RPM (posizionamenti)

```

```

ENDIF

```

```

IF #18.12=0 THEN

```

```

    #18.12=100 // quota avanzamento in unita'

```

```

ENDIF

```

```

IF (#18.13=0 AND #18.14=0) THEN

```

```

    #18.13=1 // parte intera unita' di avanzamento per giro motore

```

```

    #18.14=0 // parte decimale unita' di avanzamento per giro motore (4 cifre piu'

```

```

significative-formato fisso)

```

```

ENDIF

```

```

IF #18.15=0 THEN

```

```

    #18.15=400 // accelerazione da 0 a vel max (#18.11) in millisecondi

```

```

ENDIF

```

```

IF #18.16=0 THEN

```

```

    #18.16=400 // decelerazione da 0 a vel max (#18.11) in millisecondi

```

```

ENDIF

```

```

IF #18.17=0 THEN

```

```

    #18.17=500 // vel. movimentaz. manuale

```

```

ENDIF

```

```

// #18.18=0 //

```

```

IF #18.19=0 THEN

```

```

    #18.19=9000 // guadagno proporzionale di posizione

```

```

ENDIF

```

```

IF #18.20=0 THEN

```

```

    #18.20=1000 // guadagno feed forward

```

```

ENDIF

```

```

IF #18.26=0 THEN

```

```

    #18.26=-500 // Vel. ricerca sensore di zero

```

```

ENDIF

```

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		12/31

```

IF #18.27=0 THEN
    #18.27=20          // Vel. ricerca index
ENDIF
IF #18.28=0 THEN
    #18.28=30          // finestra di posizionamento in count encoder
ENDIF
IF #18.30=0 THEN
    #18.30=4096         // impulsi encoder per giro
ENDIF

```

FBbody

```

} // _load_default ()

```

- Link To Extra Help:
- Make Visible to Library User: Yes
- Function Block Description: No description

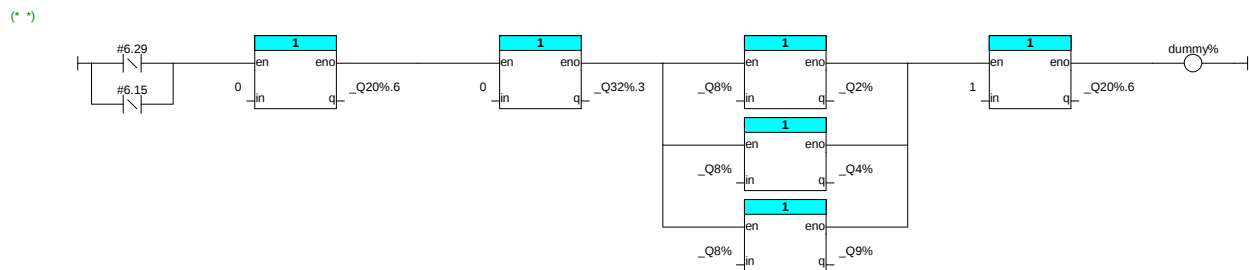
```

_Gest_disab () {
// initialisation code

```

FBbody

Diagram 3:



```

} // _Gest_disab ()

```

- Link To Extra Help:
- Make Visible to Library User: Yes
- Function Block Description: No description

```

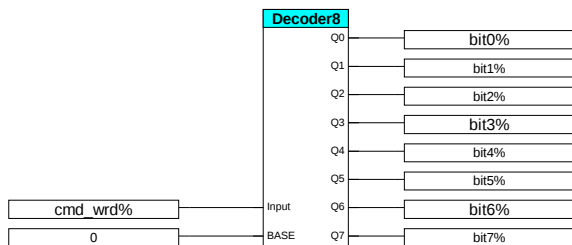
(bit0%, bit1%, bit2%, bit3%, bit4%, bit5%, bit6%, bit7%) = _cmd_bus (cmd_wrd%) {

```

FBbody

Diagram 4:

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		13/31



```
} //(bit0%, bit1%, bit2%, bit3%, bit4%, bit5%, bit6%, bit7%) = _cmd_bus (cmd_wrd%)
```

- Link To Extra Help:

- Make Visible to Library User: Yes

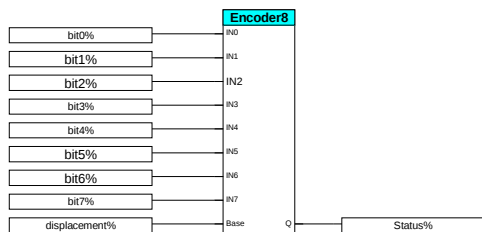
- Function Block Description: No description

```
(Status%) = _Stat_wrd (bit0%, bit1%, bit2%, bit3%, bit4%, bit5%, bit6%, bit7%, displacement%) {  
// initialisation code
```

FBbody

```
// main body code
```

Diagram 5:



```
} //(Status%) = _Stat_wrd (bit0%, bit1%, bit2%, bit3%, bit4%, bit5%, bit6%, bit7%, displacement%)
```

- Link To Extra Help:

- Make Visible to Library User: Yes

- Function Block Description: No description

```
(Home_ok%, home_in_progr%, h_stat%, Acq_ok%) = _Home_0 (Start%, V_ricerca%, V_uscita%,  
switch%, h_pos%, offs_home%, Acc%, Dec%, Acq_Abs_Pos%) {
```

```
Acq_ok%=0
```

FBbody

```
temp%=#18.30*4
```

```
V_search%=V_ricerca%*temp%/60
```

```
V_leave%=V_uscita%*temp%/60
```

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		14/31

```
Unit_per_giro=(FLOAT(#18.13)*10000.0+#18.14)/10000.0
Count_per_unit=temp%/Unit_per_giro
home_pos%=h_pos%*Count_per_unit
offs_count%=offs_home%*Count_per_unit
acc%=INT(FLOAT(#1.06)*temp%*1000.0/60.0)/Acc%
dec%=INT(FLOAT(#1.06)*temp%*1000.0/60.0)/Dec%
```

Diagram 23:

(* Double-click to enter a rung comment *)

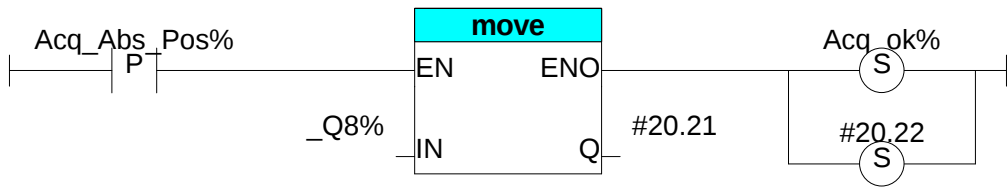
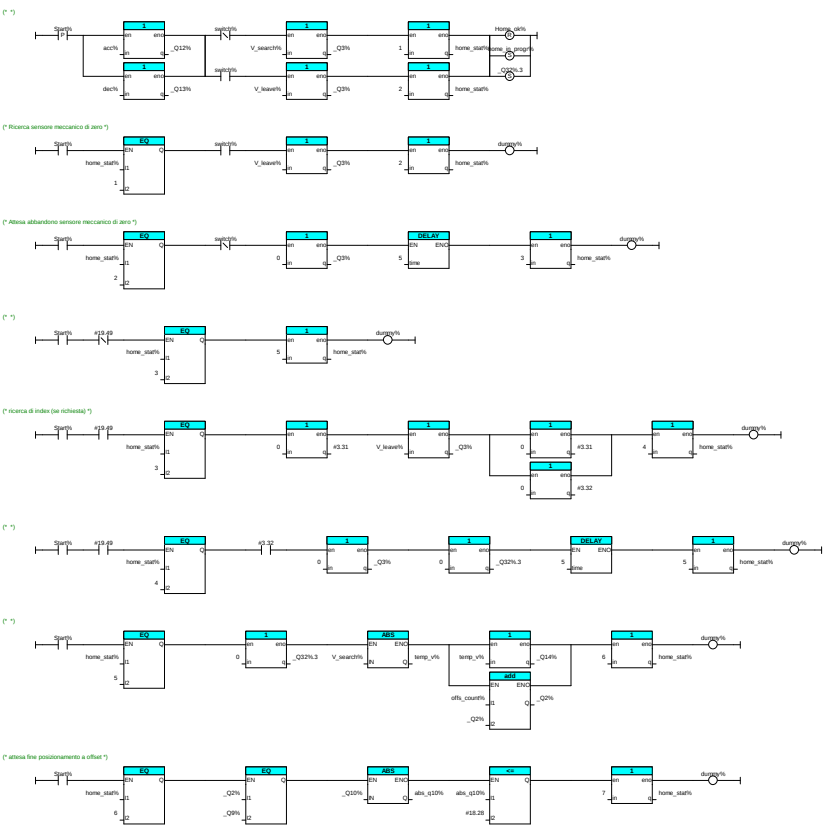
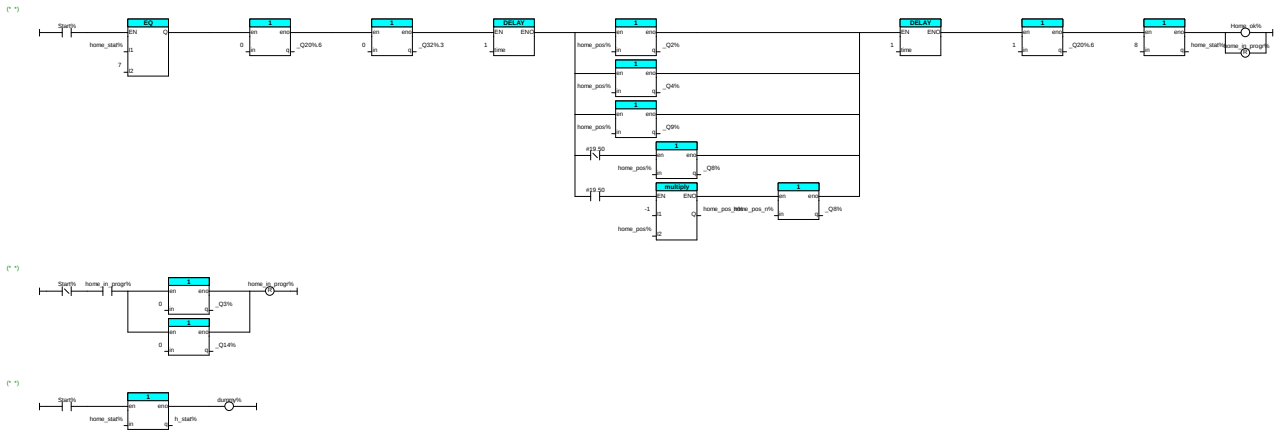


Diagram 6:



	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		15/31



} //(Home_ok%, home_in_progr%, h_stat%, Acq_ok%) = _Home_0 (Start%, V_ricerca%, V_uscita%, switch%, h_pos%, offs_home%, Acc%, Dec%, Acq_Abs_Pos%)

- Link To Extra Help:

- Make Visible to Library User: Yes

- Function Block Description: No description

(Jog_in_prog%) = _JOG (JOG_CW%, JOG_CCW%, Vel_man%, Acc%, Dec%) {

// initialisation code

FBbody

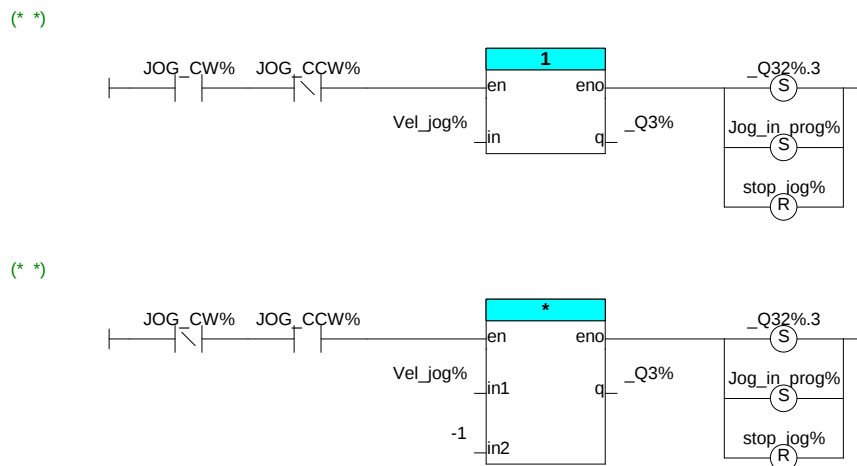
temp%=#18.30*4

Vel_jog%=Vel_man%*#18.30*4/60

_Q12%=INT(FLOAT(#1.06)*temp%*1000.0/60.0)/Acc%

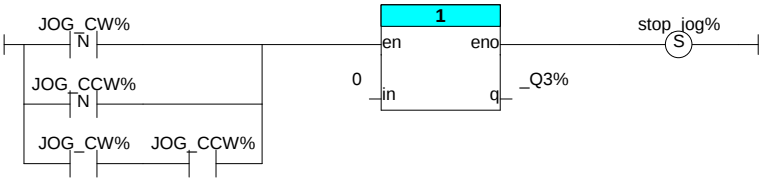
_Q13%=INT(FLOAT(#1.06)*temp%*1000.0/60.0)/Dec%

Diagram 7:

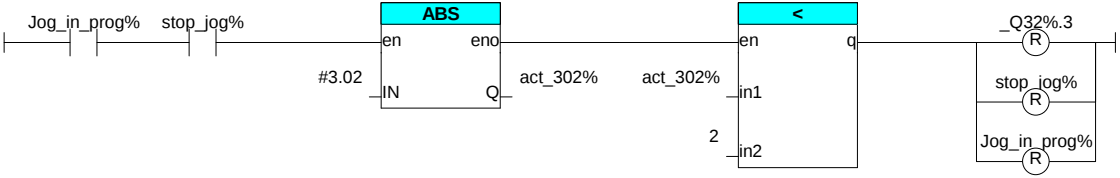


	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		16/31

(* *)



(* *)



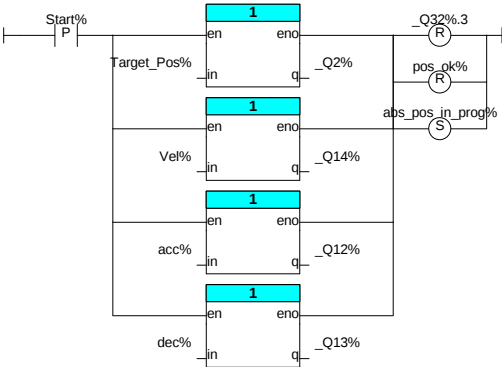
} //(Jog_in_prog%) = _JOG (JOG_CW%, JOG_CCW%, Vel_man%, Acc%, Dec%)

- Link To Extra Help:
 - Make Visible to Library User: Yes
 - Function Block Description: No description
- (pos_ok%, abs_pos_in_prog%) = _Abs_pos (Start%, Target%, Vel_Pos%, max_er%, Acc%, Dec%) {

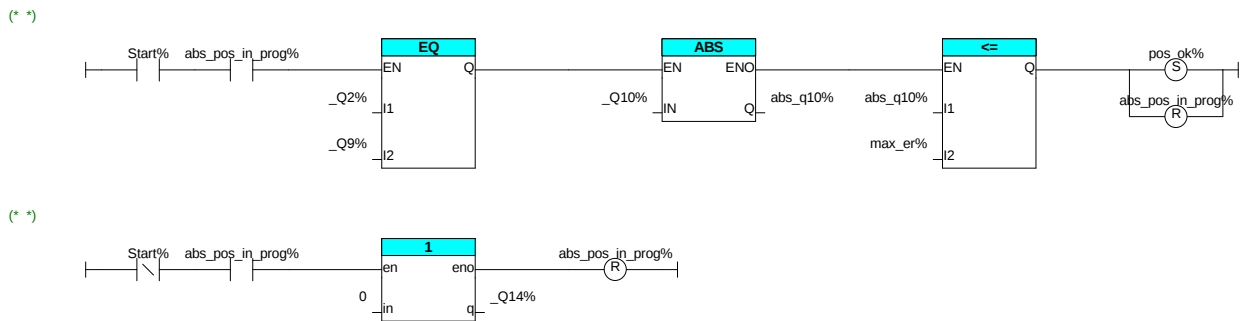
FBbody
Unit_per_giro=(FLOAT(#18.13)*10000.0+#18.14)/10000.0
temp%=#18.30*4
Count_per_unit=temp%/Unit_per_giro
Target_Pos%=(Target%*Count_per_unit)+#20.21
Vel%=Vel_Pos%*temp%/60
acc%=INT(FLOAT(#1.06)*temp%*1000.0/60.0)/Acc%
dec%=INT(FLOAT(#1.06)*temp%*1000.0/60.0)/Dec%

Diagram 8:

(* *)



	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		17/31



```

} //(pos_ok%, abs_pos_in_prog%) = _Abs_pos (Start%, Target%, Vel_Pos%, max_er%, Acc%, Dec%)

```

- Link To Extra Help:
- Make Visible to Library User: Yes
- Function Block Description: No description

```

_POSLOOP_pos () {
// initialisation code
// =====
//  inizializzazione contatore posizione e azzeramento loop
// =====

```

```

coeff_encoder%=16384/#18.30
msb% = #90.02 _0x0000FFFF
lsb% = (#90.01/65536) _0x0000FFFF
pos_abs_init% = (msb%*65536+lsb%)/(16384/#18.30)

```

```

// impostazione di _Q8% al valore iniziale
// _Q8%=0 per encoder incrementali
// _Q8%= al valore esatto se utilizzati encoder assoluti
IF #3.38=0 OR #3.38=3 THEN
    _Q8%=0
ELSE
    _Q8%=pos_abs_init%
ENDIF

```

```

    _Q2%=_Q8%
    _Q4%=_Q8%
    IF #19.50=0 THEN
        _Q9%=_Q8%
    ELSE
        _Q9%=-_Q8%
    ENDIF
    _Q19%=_Q8%

```

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		18/31

```

        _Q3%=0
//      _Q1%=0
//      _Q8%=0
//      Q9%=_Q8%
        _Q16%=812000

// Setup the system parameters and Make sure that the position controller output is OFF

        _Q32%.3 = 0
        _Q20%.1 = 1
        _Q20%.6 = 0
Act_Vel_POS%=0
Act_Ref_VEL%=0
Act_Acc_VEL%=0
Act_Pos_VEL%=_Q4%
_Q17%=0
_Q18%=0
first_posloop%=1
_Q32%.18=1

FBbody

IF _Q32%.17=1 THEN
// se _Q32%.17=1 vengono resettati i contatori
        _Q32%.3=0
        _Q8%=0
        _Q3%=0
        _Q2%=0
        _Q4%=0
        IF #19.50=0 THEN
                _Q2%=_Q8%
                _Q4%=_Q8%
                _Q9%=_Q8%
        ELSE
                _Q2%=-_Q8%
                _Q4%=-_Q8%
                _Q9%=-_Q8%
        ENDIF
        _Q32%.17=0
        first_posloop%=1
ENDIF
IF _Q32%.18=1 OR #6.29=0 OR #6.15=0 THEN
// se _Q32%.18=1 vengono i contatori vengono posti tutti = _Q8%
        _Q32%.3=0

```

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		19/31

```

IF #19.50=0 THEN
    _Q2%=_Q8%
    _Q4%=_Q8%
    _Q9%=_Q8%
ELSE
    _Q2%=-_Q8%
    _Q4%=-_Q8%
    _Q9%=-_Q8%
ENDIF
_Q3%=0
_Q32%.18=0
first_posloop%=1

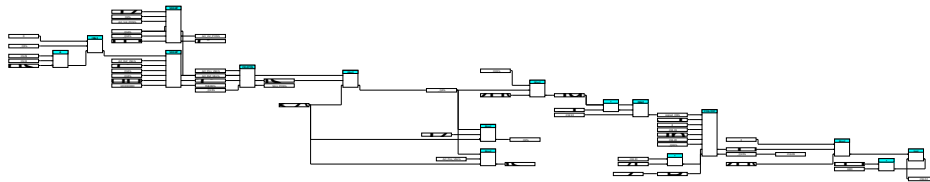
```

```

ENDIF
if first_posloop%=1 then
    _Q20%.6=0
    #91.02=0
endif

```

Diagram 9:



```

if first_posloop%=1 then
    if ABS(#19.09) < 10 then
        first_posloop%=0
        _Q20%.6=1
    else
        _Q32%.18=1
    endif
endif
endif

```

```

} // _POSLOOP_pos ()

```

- Link To Extra Help:

- Make Visible to Library User: Yes

- Function Block Description: No description

(pos_in_unit%) = _enc_step_to_un (posiz%) {

// initialisation code

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		20/31

```
FBbody
Unit_per_giro=(FLOAT(#18.13)*10000.0+#18.14)/10000.0
Count_per_unit=#18.30*4/Unit_per_giro
IF #19.50=0 THEN
    pos_in_unit%=INT(FLOAT(posiz%)/Count_per_unit)
ELSE
    pos_in_unit%=-INT(FLOAT(posiz%)/Count_per_unit)
ENDIF

} //(pos_in_unit%) = _enc_step_to_un (posiz%)
```

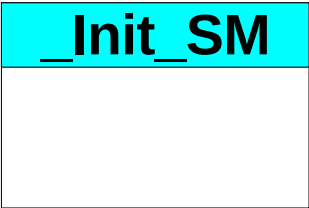
```
// =====
//      FINE USER DEFINED FUNCTION BLOCK
// =====
```

```
Initial {
Sim_acq_pos%=0
sim_FC_av%=0
sim_FC_ind%=0
Fine_Corsa_CW%=0
Fine_Corsa_CCW%=0
Sim_rst_Acq_pos%=0
rst_Acq_Pos%=0

p0%=0
// Inizializzazione variabili di sistema
///inizializzazione di default DA NON MODIFICARE

IF #15.01=301 OR #16.01=301 OR #17.01=301 THEN
// Unidrive SP + SM Application
```

Diagram 10:



```
ENDIF
```

Diagram 11:

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		21/31

_load_default

// Fine inizializzazione DA NON MODIFICARE
//

POSLOOP_enable%=1
Direzione%=0
Encoder_PPR% = 4*4096
calc_encoder%=0
JOG_CW%=0
JOG_CCW%=0
Vel_manuale%=100000

Start%=0
Target_Pos%=0
max_er%=100
Start_abs_pos%=0
Vel_abs_pos%=500000
Final_Pos%=0
max_error%=100
Start_inc_pos%=0
Start_inc1_pos%=0
Vel_inc_pos%=400000
Inc_Pos%=0
switch%=0
Start_home% =0
Home_ok%=0
home_switch%=0

Vel_search_sw%=1000
Vel_out_switch%=100
Pos_in_unit%=0
Home_torque%=0
St_2_inc_t%=0
St_2_inc_c%=0
St_2_abs_t%=0
St_2_abs_c%=0
St_torque_pos%=0
Start_modulo%=0

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		22/31

```

Inc_Pos1_OK%=0

Alb_el_rig_ON% =0
Alb_el_n_rig_ON% =0

p1%=0
Val_Modulo%=0
Init_modulo% =0

Offset%=0
Freeze_pos%=0
Limit_torque%=100
abort%=0
IF #20.22=1 THEN
    Acq_OK%=1
ENDIF
// =====
//  inizializzazione contatore posizione e azzeramento loop
// =====
step_giro_enc%=#18.30
step_giro_per_2%=step_giro_enc%*2
step_giro_per_4%=step_giro_enc%*4
coeff_encoder%=16384/step_giro_enc%
conteggio%= #90.01/(65536*coeff_encoder%)
old_conteggio%=conteggio%

// fine inizializzazione variabili blocchi operativi da non modificare


Init_m%=0
Input%=0
Val_Modulo=0

Simul_bus% =0
Sim_ter_cyc%=0
Sim_sec_cyc%=0
Simul_home%=0
Simul_abs_pos%=0
Simul_JOG_CW%=0
Simul_JOG_CCW%=0
Simul_Res_alarm%=0
Simul_switch%=0
Sim_trq_abs_pos%=0
disponibile%=0
t_abs_pos_OK%=0
Simul_incr_pos%=0

```

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		23/31

Home_OK%=0

sim_term_24%=0

sim_term_25%=0

sim_term_26%=0

sim_term_27%=0

sim_term_28%=0

sim_term_29%=0

} //Initial

Pos0 {

p0%=p0%+1

conteggio% = #90.01/(65536*coeff_encoder%)

delta_count%=conteggio%-old_conteggio%

IF delta_count% > (step_giro_per_2%-1)THEN

delta_count%=delta_count%-step_giro_per_4%

ELSEIF delta_count% < -step_giro_per_2% THEN

delta_count%=delta_count%+step_giro_per_4%

ENDIF

_Q8%=_Q8%+delta_count%

old_conteggio%=conteggio%

// =====

// fine creazione contatore posizione

// =====

} //Pos0

Background {

TOP:

// =====

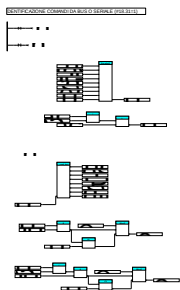
// gestione comandi

// =====

// gestione comandi da bus (#18.31=1)

Diagram 12:

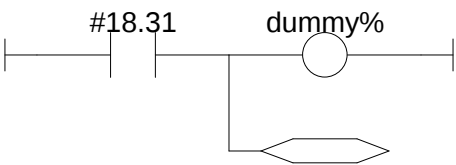
	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		24/31



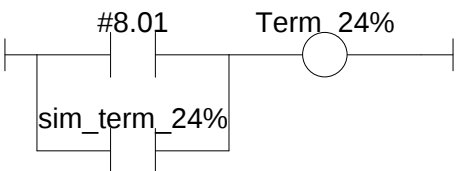
// gestione comandi da Input digitali (#18.31=0)
 // acquisizione ingressi

Diagram 13:

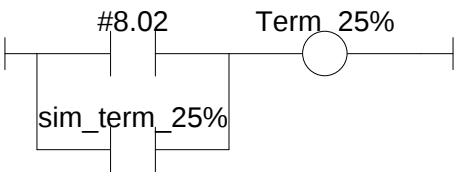
(* *)



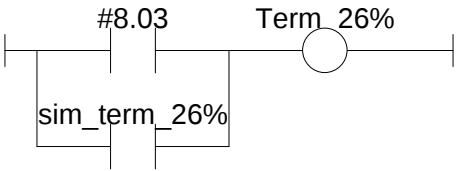
(* *)



(* *)

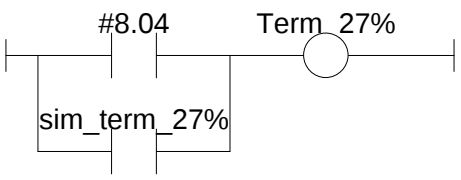


(* *)

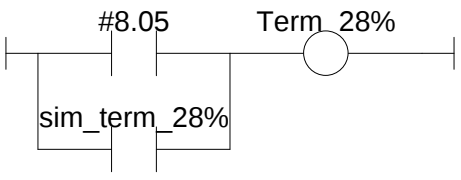


	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		25/31

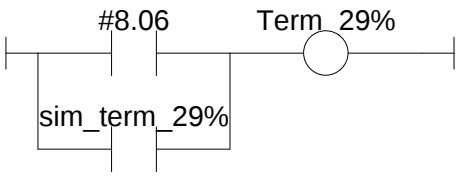
(* *)



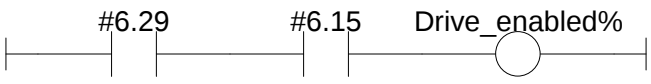
(* *)



(* *)



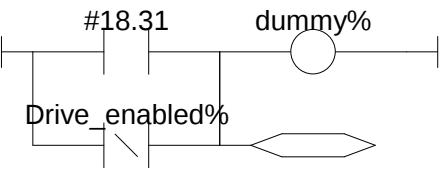
(* *)



// gestione ingressi
// *****
// e' questa la sezione da modificare per cambiare la logica di comando
// *****

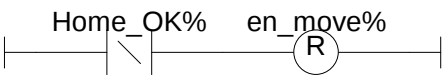
Diagram 14:

(* Ignora ingressi se comandi da bus o drive disabilitato *)

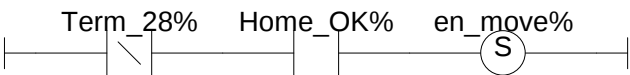


	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		26/31

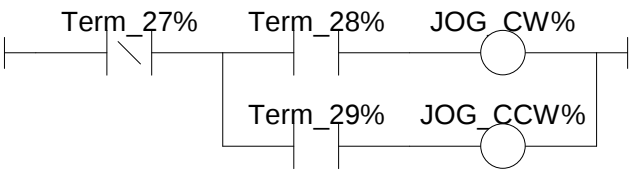
(* abilita aggancio albero elettrico sul fronte del terminale 28 dopo aver eseguito la ricerca di zero
--- da eliminare i due rung che seguono nel caso si desideri l' aggancio automaticamente trovato l' homi



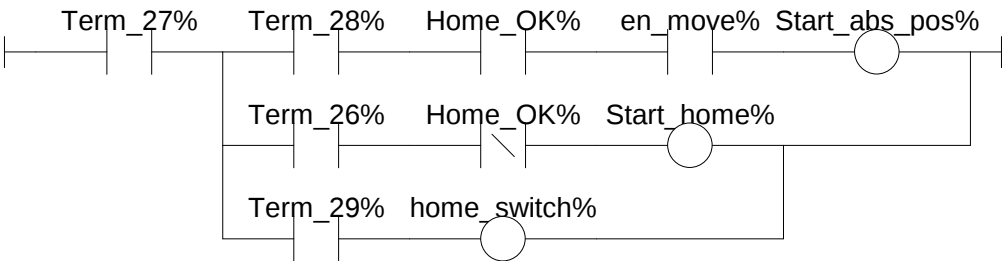
(* *)



(* *)



(* *)



// fine gestione ingressi

// =====

// fine gestione comandi

// =====

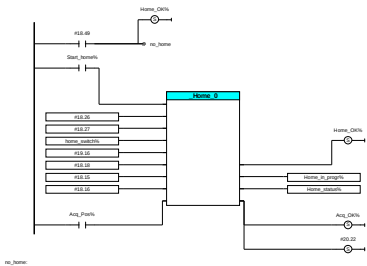
// ricerca di zero

```
IF rst_Acq_pos%=1 AND #20.22=1 THEN
    Acq_OK%=0
    #20.22=0
ENDIF
```

Diagram 15:

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		27/31

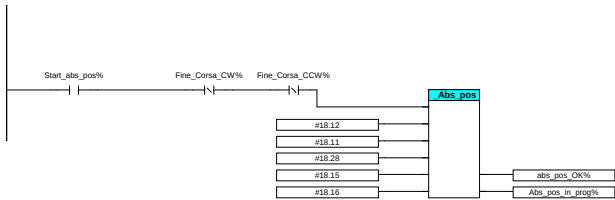
RICERCA DI ZERO



```
// movimento assoluto
IF Home_OK%=1 OR Acq_OK%=1 THEN
```

Diagram 16:

MOVIMENTO ASSOLUTO

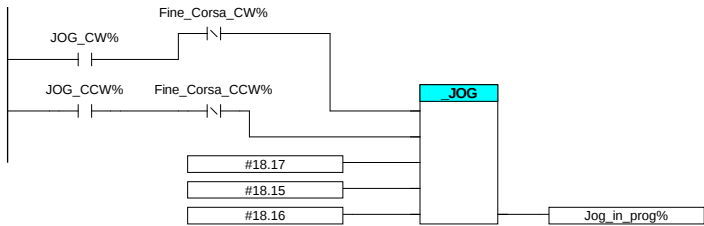


ENDIF

```
// movimentazione manuale JOG
```

Diagram 17:

MOVIMENTAZIONE MANUALE



GOTO TOP:

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		28/31

} //Background

Clock {

```
// *****  
// Loop di posizione  
// *****
```

Diagram 18:

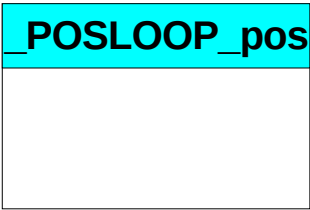
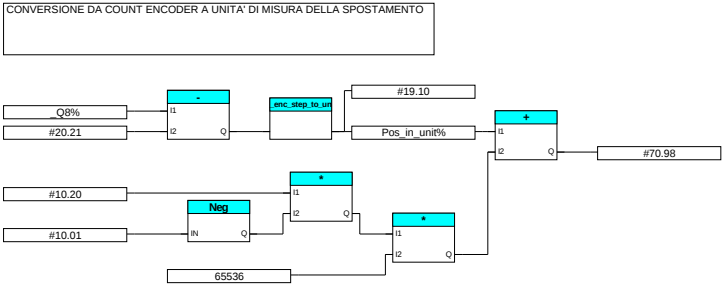


Diagram 19:



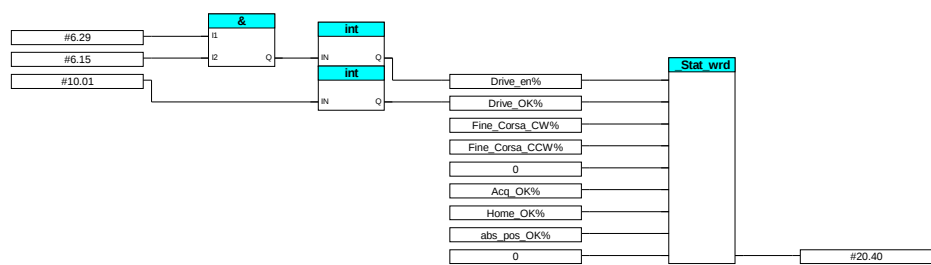
```
// =====  
//                               Fine corsa  
// =====  
IF #20.22=1 THEN  
Pos_Fine_Corsa_av%=#19.13  
Pos_Fine_Corsa_ind%=#19.14
```

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		29/31
	field2		

```
IF (Pos_in_unit% > Pos_Fine_Corsa_av%) OR (sim_FC_av%=1)THEN
    Fine_Corsa_CW%=1
ELSE
    Fine_Corsa_CW%=0
ENDIF
IF (Pos_in_unit% < Pos_Fine_Corsa_ind%)OR (sim_FC_ind%=1) THEN
    Fine_Corsa_CCW%=1
ELSE
    Fine_Corsa_CCW%=0
ENDIF
ENDIF
// =====
//               termine gestione Fine corsa
// =====

// Status Word
```

Diagram 20:



```
// impostazione uscite
// gestione Freno
```

Diagram 22:

(* Sblocca freno quando active *)



```
} //Clock
```

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		30/31

	field1	Project ASSE_Z_ASM	28/09/2005
	Version1		
	field2		31/31