

AUTHOR SB
COMPANY Eurobica Corporate
TITLE Matrix
VERSION V1.0
TARGET SMAPPLPLUS
DRIVE SM-Applications Plus

DESCRIPTION:
No description

```
//#####  
//#####  
//##### M A T R I X #####  
//#####  
//#####
```

Notes {

- Verifica dei \$

} //Notes

- Link To Extra Help:

- Make Visible to Library User: Yes

- Function Block Description: No description

(RampaAccOut%, RampaDecOut%, JerckOut%) = _ConvertRamp (RampaAccIn%, RampaDecIn%, JerckIn%, RampType%) {

FBbody

SELECT RampType%

CASE 0 // RAMPA VELOCITA' TRADIZIONALE

Temp1=FLOAT(#1.06)*65536*1000.0/60.0

RampaAccOut%=INT(Temp1/RampaAccIn%)

RampaDecOut%=INT(Temp1/RampaDecIn%)

Temp2%=MAX(JerckIn%,1)

IF Temp2% > RampaAccIn%/2 THEN Temp2%=RampaAccIn%/3

JerckOut%=INT(LIMIT((1000.0*FLOAT(RampaAccOut%)/Temp2%),2147000000))

CASE 1 // RAMPA VELOCITA' SLEW

RampaAccOut%=INT(1000000000.0/FLOAT(RampaAccIn%))

RampaDecOut%=INT(1000000000.0/FLOAT(RampaDecIn%))

ENDSELECT

} //(RampaAccOut%, RampaDecOut%, JerckOut%) = _ConvertRamp (RampaAccIn%, RampaDecIn%, JerckIn%, RampType%)

- Link To Extra Help:

- Make Visible to Library User: Yes

- Function Block Description: No description

(VirtualEncoder%, ActualSpeed%) = _VirtualEncoder (Enable%, RefRpm%, AccMs%, DecMs%, MaxSpeed%) {

RampEnable% = 1

Inc% =1000

ActSpd% =0

ActAcc% =0

ActPos% =0

FBbody

Ref% = RefRpm% * 65536 / 60

(Acc%, Dec%, Dummy%) = _ConvertRamp (AccMs%, DecMs%, 0, 0)

Ref2% = iMux2 (0 , Ref% , Enable%)

(ActSpd%,ActAcc%,Jerk%) = SRAMP (Ref2%,ActSpd%,ActAcc%,Acc%,Dec%,10000000,10000000)

(ActPos%,StpPos%) = VEL2POS (ActPos%,ActSpd%,ActAcc%,Dec%)

Inc% = ActPos%

VirtualEncoder% = (Inc%+_Q1%)

IF ActSpd% <> 0 THEN

ActualSpeed% = ActSpd% * 60 / 65536 + 1

ELSE

ActualSpeed% = 0

ENDIF

Matrix.DPL

14/01/2016

Page 1

```

} //(VirtualEncoder%, ActualSpeed%) = _VirtualEncoder (Enable%, RefRpm%, AccMs%, DecMs%, MaxSpeed%)
//.....
// -----  PARAMETRI DRIVER  -----
Alias {
//-----
//-----

$DEFINE DRV_MaxSpeed          #01.06 // Massima Velocità del Motore
$DEFINE DRV_BipolarRef%       #01.10 // Selezione Riferimento Bipolare
$DEFINE DRV_SelRef%           #01.14 // Riferimento di velocità da velocità digitali
$DEFINE DRV_SelDgtRef%        #01.15 // Selezione velocità digitale
//-----

$DEFINE DRV_DisRamp%          #02.02 // Disabilitazione Rampe
$DEFINE DRV_EnFastRamp%       #02.04 // Abilitazione Rampe Fast
$DEFINE PrSPD_AccCn           #02.11 // Rampa Accelerazione s/100Hz
$DEFINE PrSPD_DecCn           #02.21 // Rampa Decelerazione s/100Hz
//-----

$DEFINE DRV_ActMotorRpm       #03.02 // Velocità Attuale del Motore

$DEFINE DRV_SpdKp             #03.10
$DEFINE DRV_SpdKi             #03.11
$DEFINE DRV_SpdKd             #03.12

$DEFINE DRV_EncConnect%       #03.26 // Selettore Retrazione Velocità (dove è collegato l'Encoder)
$DEFINE DRV_Enc_Alim%         #03.36 // Alimentazione Encoder Motore
$DEFINE DRV_EncType%          #03.38 // Tipo di Encoder Motore per la Chiusura del LOOP DI POSIZIONE
$DEFINE DRV_EncAutodetect%    #03.41 // Abilitazione Autodetect Encoder Motore
//-----

$DEFINE DRV_ActMotorCorrent    #04.02 // Assorbimento Attuale del Motore
$DEFINE DRV_LimCorrent         #04.05 // Limite di Corrente Motore
$DEFINE DRV_LimSimCorrent      #04.07 // Limite di Corrente Simmetrica
$DEFINE DRV_SelDriveMode%      #04.11 // Selezione Modo Operativo Drive
$DEFINE DRV_CorrentFilter      #04.12
$DEFINE DRV_CorrentKp%         #04.13
$DEFINE DRV_CorrentKi%         #04.14
//-----

$DEFINE DRV_SelStartMode%      #06.04 // Selezione della Logica di Avviamento/Comando
$DEFINE DRV_Enabled%           #06.15 // Abilitazione Software
$DEFINE DRV_HwEnable%          #06.29 // Abilitazione Hardware
$DEFINE DRV_Forward%           #06.30 // Run Forward
$DEFINE DRV_Reverse%           #06.32 // Run Reverse
$DEFINE DRV_SelSeqLatch%       #06.40 // Abilitazione Sequencer Latching
//-----

$DEFINE DRV_FbTermT05          #07.01 // Stato AI T05 (Volt)
$DEFINE DRV_FbTermT07          #07.02 // Stato AI T07
$DEFINE DRV_FbTermT08          #07.03 // Stato AI T08
$DEFINE DRV_CompTermT05        #07.07 // Compensazione Terminale T05
$DEFINE DRV_ScaleTermT05       #07.08 // Scalatura Terminale T05
$DEFINE DRV_InvTermT05%        #07.09 // Inversione Terminale T05
$DEFINE DRV_ParamTermT05       #07.10 // Associazione Parametro Terminale T05
$DEFINE DRV_TypTermT07%        #07.11 // Tipo Terminale T07 (0=0.20mA,4=4.20mA,6=Volt)
$DEFINE DRV_ScaleTermT07       #07.12 // Scalatura Terminale T07
$DEFINE DRV_InvTermT07%        #07.13 // Inversione Terminale T07
$DEFINE DRV_ParamTermT07       #07.14 // Associazione Parametro Terminale T07
$DEFINE DRV_TypTermT08%        #07.15 // Tipo Terminale T08 (0=0.20mA,4=4.20mA,6=Volt)
$DEFINE DRV_ScaleTermT08       #07.16 // Scalatura Terminale T08
$DEFINE DRV_InvTermT08%        #07.17 // Inversione Terminale T08
$DEFINE DRV_ParamTermT08       #07.18 // Associazione Parametro Terminale T08
$DEFINE DRV_ParamTermT09       #07.19 // Associazione Parametro Terminale T09 (AO)
$DEFINE DRV_ScaleTermT09       #07.20 // Scalatura Terminale T09
$DEFINE DRV_TypTermT09%        #07.21 // Tipo Terminale T09 (0=Volt,1=0.20mA,2=4.20mA)
$DEFINE DRV_ParamTermT10       #07.22 // Associazione Parametro Terminale T10 (AO)
$DEFINE DRV_ScaleTermT10       #07.23 // Scalatura Terminale T10
$DEFINE DRV_TypTermT10%        #07.24 // Tipo Terminale T10 (0=Volt,1=0.20mA,2=4.20mA)
$DEFINE DRV_OffsTermT05        #07.30 // Offset Terminale T05
$DEFINE DRV_OffsTermT07        #07.31 // Offset Terminale T07

```

```

$DEFINE DRV_OffsTermT08          #07.32 // Offset Terminale T08
//-----
$DEFINE DRV_FbTermT24%           #08.01 // Stato DI o DO T24
$DEFINE DRV_FbTermT25%           #08.02 // Stato DI o DO T25
$DEFINE DRV_FbTermT26%           #08.03 // Stato DI o DO T26
$DEFINE DRV_FbTermT27%           #08.04 // Stato DI T27
$DEFINE DRV_FbTermT28%           #08.05 // Stato DI T28
$DEFINE DRV_FbTermT29%           #08.06 // Stato DI T29
$DEFINE DRV_FbTermRele%          #08.07 // Stato DO T41 (Relè)
$DEFINE DRV_FbTermT22%           #08.08 // Stato DO T22
$DEFINE DRV_FbTermT31%           #08.09 // Stato DI T31 (Hardware Enable%)
$DEFINE DRV_InvTermT31%           #08.10 // Inversione Terminale T31 (Hardware Enable%)
$DEFINE DRV_InvTermT24%           #08.11 // Inversione Terminale T24
$DEFINE DRV_InvTermT25%           #08.12 // Inversione Terminale T25
$DEFINE DRV_InvTermT26%           #08.13 // Inversione Terminale T26
$DEFINE DRV_InvTermT27%           #08.14 // Inversione Terminale T27
$DEFINE DRV_InvTermT28%           #08.15 // Inversione Terminale T28
$DEFINE DRV_InvTermT29%           #08.16 // Inversione Terminale T29
$DEFINE DRV_InvTermTRele%         #08.17 // Inversione Terminale T41 (Relè)
$DEFINE DRV_InvTermT22%           #08.18 // Inversione Terminale T22
$DEFINE DRV_ParamTermT24          #08.21 // Associazione Parametro Terminale T24
$DEFINE DRV_ParamTermT25          #08.22 // Associazione Parametro Terminale T25
$DEFINE DRV_ParamTermT26          #08.23 // Associazione Parametro Terminale T26
$DEFINE DRV_ParamTermT27          #08.24 // Associazione Parametro Terminale T27
$DEFINE DRV_ParamTermT28          #08.25 // Associazione Parametro Terminale T28
$DEFINE DRV_ParamTermT29          #08.26 // Associazione Parametro Terminale T29
$DEFINE DRV_ParamTermRele         #08.27 // Associazione Parametro Terminale T41 (Relè)
$DEFINE DRV_ParamTermT22          #08.28 // Associazione Parametro Terminale T22
$DEFINE DRV_LogicaPNP%            #08.29 // Logica Positiva PNP (da T24 a T29)
$DEFINE DRV_DoTermT24%            #08.31 // Definizione DI o DO Terminale T24 (0=DI 1=DO)
$DEFINE DRV_DoTermT25%            #08.32 // Definizione DI o DO Terminale T25 (0=DI 1=DO)
$DEFINE DRV_DoTermT26%            #08.33 // Definizione DI o DO Terminale T26 (0=DI 1=DO)
$DEFINE DRV_DisSetAutT28eT29%     #08.39 // Disabilitazione Impostazione Automatica T28 e T29
//-----
$DEFINE DRV_DriveParSave%         #10.00 // Salvataggio Parametri
$DEFINE DRV_Ok%                   #10.01 // Drive OK
$DEFINE DRV_ZeroSpeed%            #10.03 // Segnale di Velocità Zero
$DEFINE DRV_AtSpd%                #10.06 // Segnale di Velocità Raggiunta
$DEFINE DRV_Reset%                #10.33 // Reset delle Anomalie del Drive
$DEFINE DRV_ParReset%             #10.38 // Reset Drive
//-----
$DEFINE DRV_IdModSlot1%           #15.01 // ID Modulo Opzionale inserito nello Slot 1

// SM Universal Encoder Plus
$DEFINE SmUEP_Spd%                #15.03 // Velocità (giri/min)
$DEFINE SmUEP_RevCnt%             #15.04 // Contagiri
$DEFINE SmUEP_ActPos%             #15.05 // Posizione Attuale
$DEFINE SmUEP_ActPosFn%           #15.06 // Posizione Fine Attuale

// Modulo Profibus
$DEFINE DRV_Dp_Address%           #15.03 // Indirizzo Profibus
$DEFINE DRV_Dp_Spd%               #15.04 // Velocità rete Profibus
$DEFINE DRV_Dp_NumWords%          #15.05 // Quantità Word in Scambio con S7
$DEFINE DRV_Dp_OutParam01%        #15.10 // Parametro in Uscita
$DEFINE DRV_Dp_OutParam02%        #15.11 // Parametro in Uscita
$DEFINE DRV_Dp_OutParam03%        #15.12 // Parametro in Uscita
$DEFINE DRV_Dp_OutParam04%        #15.13 // Parametro in Uscita
$DEFINE DRV_Dp_OutParam05%        #15.14 // Parametro in Uscita
$DEFINE DRV_Dp_OutParam06%        #15.15 // Parametro in Uscita
$DEFINE DRV_Dp_OutParam07%        #15.16 // Parametro in Uscita
$DEFINE DRV_Dp_OutParam08%        #15.17 // Parametro in Uscita
$DEFINE DRV_Dp_OutParam09%        #15.18 // Parametro in Uscita
$DEFINE DRV_Dp_OutParam10%        #15.19 // Parametro in Uscita
$DEFINE DRV_Dp_InParam01%         #15.20 // Parametro in Ingresso
$DEFINE DRV_Dp_InParam02%         #15.21 // Parametro in Ingresso
$DEFINE DRV_Dp_InParam03%         #15.22 // Parametro in Ingresso

```

```

$DEFINE DRV_Dp_InParam04%      #15.23 // Parametro in Ingresso
$DEFINE DRV_Dp_InParam05%      #15.24 // Parametro in Ingresso
$DEFINE DRV_Dp_InParam06%      #15.25 // Parametro in Ingresso
$DEFINE DRV_Dp_InParam07%      #15.26 // Parametro in Ingresso
$DEFINE DRV_Dp_InParam08%      #15.27 // Parametro in Ingresso
$DEFINE DRV_Dp_InParam09%      #15.28 // Parametro in Ingresso
$DEFINE DRV_Dp_InParam10%      #15.29 // Parametro in Ingresso
//-----
$DEFINE DRV_IdModSlot2%         #16.01 // ID Modulo Opzionale inserito nello Slot 2 (per SM App usare il
menù #81.xx)
//-----
$DEFINE DRV_IdModSlot3%         #17.01 // ID Modulo Opzionale inserito nello Slot 3 (per SM App usare il
menù #81.xx)
//-----
// $DEFINE FRZ_EnDiExtEncoder%   #16.38 // Abilitazione dell'ingresso Dig come comando di Freeze Encoder
di Riferimento
// $DEFINE FRZ_SelFnExtEncoder%  #16.41 // Freeze su Fronte Negativo Encoder di Riferimento
//                               #70.xx // Usare _Px% etc. anzichè #70 (velocizzo la scansione)
//-----
$DEFINE DRV_SmAppSwRel          #81.02 // Release Software SM Application
$DEFINE DRV_SetCtSync%          #81.06 // CtSync: 25=CtMaster 26=CtSlave
$DEFINE DRV_ClockThick%         #81.11 // Tempo di Campionamento della Task Clock
$DEFINE DRV_Pos0Thick%          #81.12 // Tempo di Campionamento della Task Pos0
$DEFINE DRV_EnAutoRun%          #81.13 // Abilitazione Autorun
$DEFINE DRV_EnTripErr%          #81.14 // Abilitazione Trip per Errori Run Time
$DEFINE DRV_EnAutSavPar%        #81.20 // Abilitazione Salvataggio dei Menù 20,70,71,74,75 allo
Spegnimento del drive
$DEFINE DRV_EnAutSavMenu20%     #81.21 // Abilitazione Salvataggio del Menù 20
$DEFINE DRV_CtNetTokenRing3%    #81.22 // Se è presente un'unica rete CTNet lasciare a 0
$DEFINE DRV_CtNetAddress%       #81.23 // da 1 a 255, 0 disabilita la CtNet locale
$DEFINE DRV_CtNetSpd%           #81.24 // 0=5 Mbit/s 1=2.5 Mbit/s 2=1.25Mbit/s
$DEFINE DRV_CtNetMst%           #81.25 // Solo Master: 1005=dove 100ms per le comunicazioni lente e 5ms
per le veloci

$DEFINE FRZ_EnDiRealEncoder%     #81.42 // Abilitazione dell'ingresso Dig In0 come comando di Freeze
Encoder Retroazione
$DEFINE FRZ_SelFnRealEncoder%    #81.43 // Freeze su Fronte Negativo Encoder di Retroazione
$DEFINE DRV_EnPos0Priority%      #81.44 // Imposta la Task Pos0 come Task Prioritaria
$DEFINE DRV_DebugInfo_Line%     #81.48 // Riga arresto da Debug Information
$DEFINE DRV_DebugInfo_Err%      #81.50 // Errore arresto da Debug Information

$DEFINE DRV_EvntCmdWrd%         #85.01 // Parola di Comando Task Event

$DEFINE DI_FtcSlv%              #86.01 // Stato DI T10
$DEFINE DI_FtcMst%              #86.02 // Stato DI T11
$DEFINE DRV_SpTermSmT12%        #86.03 // Comando DO T12
$DEFINE DRV_SpTermSmT13%        #86.04 // Comando DO T13

$DEFINE DRV_DebugInfo_Task%     #88.02 // Task arresto da Debug Information

$DEFINE FRZ_RichRealEncoder_%    #90.18 // Richiesta e Feedback Freeze Encoder di Retroazione
$DEFINE FRZ_RichExtEncoder%      #90.28 // Richiesta e Feedback Freeze Encoder di Riferimento // $ (work
presente: Debug FTC Mst Simulata)
$DEFINE DRV_RealEncoder%         #90.31 // Contatore Encoder Motore
$DEFINE DRV_ExtEncoder%          #90.32 // Contatore Encoder Esterno // $ (work presente: Debug Encoder
$DEFINE FRZ_PosEncRealEncoder_%  #90.33 // Posizione Freeze Encoder di Retroazione
$DEFINE FRZ_PosEncExtEncoder%    #90.34 // Posizione Freeze Encoder di Riferimento // $ (work presente:
Debug Encoder Simulato)
$DEFINE Marker_Flag_Pulse%       #90.41 // Marker dell'Encoder Master
$DEFINE DRV_RefEncMarker%        #90.42 // Tacca di zero Encoder Motore
$DEFINE DRV_RefEncSource%        #90.43 // Selezione Encoder Reference
$DEFINE DRV_FbEncSource%         #90.44 // Selezione Encoder Feedback
$DEFINE Marker_Flag_Enable%      #90.45 // Abilitazione Rilevamento Marker dell'Encoder Master
$DEFINE DRV_RefEncMarkerEn%      #90.46 // Abilitazione Lettura Tacca di zero Encoder Motore
$DEFINE FRZ_EnExtEncoder%        #90.47 // Abilitazione Freeze Encoder di Riferimento
$DEFINE FRZ_EnRealEncoder%       #90.48 // Abilitazione Freeze Encoder di Retroazione

```

```

$DEFINE DRV_EnQckRefSpd%      #91.01 // Abilita l'aggiornamento veloce del Riferimento di Velocità
$DEFINE DRV_SpdRef%           #91.02 // Riferimento di velocità da POSLOOP
$DEFINE DRV_MaxIntSpeed%      #91.05 // Velocità Massima del Motore

} //Alias
// ----- PARAMETRI PROGRAMMA -----
Alias {

    $DEFINE Gain%              #18.11 // Guadagno Proporzionale Loop di Posizione
    $DEFINE FdFrwd%            #18.12 // Feed Forward Loop di Posizione
    $DEFINE Jerck%             #18.13 // Jerck
    $DEFINE Max_Error%         #18.14 // Errore Posizionamento
    $DEFINE Modality%          #18.15 // Modalità Operativa

    $DEFINE Default_Parameter% #18.31 // 0=Abilitazione Caricamento dei Parametri di Default
    $DEFINE SwEnable%          #18.32 // Enable Software
    $DEFINE Jog_CW%            #18.33 // Jog Avanti
    $DEFINE Jog_CCW%           #18.34 // Jog Indietro
    $DEFINE Jog_InProg%        #18.35 // Jog in corso
    $DEFINE Hom_Enable%        #18.36 // Enable Homing
    $DEFINE Hom_Run%           #18.37 // Start Home
    $DEFINE Hom_InProg%        #18.38 // Home in corso
    $DEFINE Hom_Ok%            #18.39 // Home Eseguito

    $DEFINE CT_Status_00%      #19.31 // Uscita Digitale Parametro T24
    $DEFINE CT_Status_01%      #19.32 // Uscita Digitale Parametro T25
    $DEFINE DI_HMI_01%         #19.33 // Ingresso Digitale Parametro T26
    $DEFINE DI_HMI_02%         #19.34 // Ingresso Digitale Parametro T27
    $DEFINE DI_SaAut%          #19.35 // Ingresso Digitale Parametro T28: Selettore Automatico
// $ (work presente: Debug Cmnd Simulati)
    $DEFINE DI_SbJog_Hom%      #19.36 // Ingresso Digitale Parametro T29: Pulsante Jog - Hom
// $ (work presente: Debug Cmnd Simulati)
    $DEFINE EnStaz%            #19.40 // Ciclo Automatico in Corso

    $DEFINE Jog_SpdRpm%        #20.01 // Velocità di Movimentazione Jog in Rpm
    $DEFINE Jog_AccDecMs%      #20.02 // Accelerazione-Decelerazione JOG in millisecondi
    $DEFINE Hom_SpdRpm%        #20.03 // Velocità di Homing in Rpm
    $DEFINE Hom_AccDecMs%      #20.04 // Accelerazione-Decelerazione Home in millisecondi
    $DEFINE Egb_AccMs%         #20.05 // Accelerazione EGB
    $DEFINE Egb_DecPosMs%      #20.06 // Decelerazione EGB Sgancio in Posizione
    $DEFINE Egb_DecSpdMs%      #20.07 // Decelerazione EGB Sgancio in Velocità
    $DEFINE Egb_RcInizSpdRpm%  #20.08 // Velocità Recupero Iniziale Slave
    $DEFINE Egb_RcInizAccDecMs% #20.09 // Accelerazione e Decelerazione Recupero Iniziale Slave

    $DEFINE Egb_RcLavAccDecMs% #20.10 // Accelerazione e Decelerazione Recupero Lavoro Slave
    $DEFINE Egb_RcFinalSpdRpm% #20.11 // Velocità Recupero Finale Slave
    $DEFINE Egb_RcFinalAccDecMs% #20.12 // Accelerazione e Decelerazione Recupero Finale Slave
    $DEFINE QuotaFtcMst%       #20.13 // Distanza tra Ftc Master e Punto Accoppiamento (mm)
    $DEFINE QuotaFtcSlv%       #20.14 // Distanza tra Ftc Slave e Punto Accoppiamento (mm)
    $DEFINE QuotaStandBySlv%    #20.15 // Distanza tra Ftc Slave e Punto StandBy (mm)
    $DEFINE QuotaAccopp%        #20.16 // Quota durata Accoppiamento (Slave) (mm)
    $DEFINE QuotaOfst%          #20.17 // Quota Sfasamento Master/Slave (0.1mm)
    $DEFINE QuotaOscuramSlv%    #20.18 // Quota Oscuramento Slave (0.1mm)
    $DEFINE QuotaOscuramMst%    #20.19 // Quota Oscuramento Master (0.1mm)

    $DEFINE Ratio_EncIntS%      #20.21 // Parte Intera Rapporto Encoder Motore (5 cifre)
    $DEFINE Ratio_EncDecS%      #20.22 // Parte Decimale Rapporto Encoder Motore (5 cifre)
    $DEFINE Ratio_EncIntM%      #20.23 // Parte Intera Rapporto Encoder Esterno (5 cifre)
    $DEFINE Ratio_EncDecM%      #20.24 // Parte Decimale Rapporto Encoder Esterno (5 cifre)
    $DEFINE Info_DebugLineIm%   #20.25 // Immagine Ritentiva Riga arresto da Debug Information
    $DEFINE Info_DebugErrIm%    #20.26 // Immagine Ritentiva Errore arresto da Debug Information
    $DEFINE Info_DebugTaskIm%   #20.27 // Immagine Ritentiva Task arresto da Debug Information

    $DEFINE Sim_LunghM%         #20.30 // Simulatore Pezzi
    $DEFINE Sim_SpazioM%        #20.31 // Simulatore Pezzi

```

```

$DEFINE Sim_LunghS%           #20.32      // Simulatore Pezzi
$DEFINE Sim_SpazioS%          #20.33      // Simulatore Pezzi
//-----

} //Alias
//-----
Initial {

    CALL InizParametriDrive:
    CALL InizParametriProgramma:
    CALL Calcoli:
//-----
// Parametri Default Programma
    IF Default_Parameter%=0 THEN
        CALL DefParamProgramma:           // Caricamento dei Parametri di Default
        DRV_DriveParSave%=1001           // Salvataggio dei Parametri
        DELAY(5)                          // Ritardo di 0,5 sec.
        DRV_ParReset%=100                // Reset del Drive
        REINIT                           // Reinizializzazione del Sistema
        DELAY(5)                          // Ritardo di 0,5 sec.
        DRV_DriveParSave%=1070           // Reinizializzazione SM-Application
        DELAY(5)                          // Ritardo di 0,5 sec.
        DRV_ParReset%=100                // Reset del Drive
    ENDIF
//-----
// Senso Rotazione Motore
    DRV_Forward%=1
    DRV_Reverse%=0
//-----
// Inizializzazione Contatore di Posizione
    Encoder_PPR%=65536

} //Initial
DefParamProgramma: {

    Gain%=9000
    FdFrwd%=989
    Max_Error%=100

    Ratio_EncIntS%=218
    Ratio_EncDecS%=45333330
    Ratio_EncIntM%=32
    Ratio_EncDecM%=73222537

    Jog_SpdRpm%=200
    Jog_AccDecMs%=500

    Hom_SpdRpm%=300
    Hom_AccDecMs%=500

    Egb_AccMs%=25
    Egb_DecPosMs%=25
    Egb_DecSpdMs%=30
    Egb_RcInizSpdRpm%=1000
    Egb_RcInizAccDecMs%=25
    Egb_RcLavAccDecMs%=100
    Egb_RcFinalSpdRpm%=2000
    Egb_RcFinalAccDecMs%=50

    QuotaFtcMst%=900
    QuotaFtcSlv%=370
    QuotaStandBySlv%=270
    QuotaAccopp%=50
    QuotaOfst%=0
    QuotaOscuramSlv%=100

```

```

QuotaOscuramMst%=100

Default_Parameter%=1

} //DefParamProgramma:
InizParametriDrive: {

// Logica di funzionamento
DRV_MaxSpeed=3000.0           // Massima Velocità del Motore
DRV_Enabled%=0                // Abilitazione Software

DRV_BipolarRef%=1             // Selezione Riferimento Bipolare
DRV_SelRef%=3                  // Riferimento di velocità da velocità digitali
DRV_SelDgtRef%=1              // Selezione velocità digitale 1
DRV_DisRamp%=0                // Disabilitazione Rampe
DRV_EnFastRamp%=0             // Rampe Fast
DRV_LimSimCorrent=DRV_LimCorrent // Limiti di Corrente Simmetrici
DRV_SelDriveMode%=0           // Selezione Controllo in Velocità
DRV_SelStartMode%=4           // Disabilita Logiche di Start / Stop da Input Digitali
DRV_SelSeqLatch%=0            // Abilitazione Sequencer Latching

// Riferimento e Feedback Encoder Motore
DRV_Enc_Alim%=0               // Alimentazione dell'Encoder=5V.
DRV_EncType%=3                // Impostazione dell'Encoder=AbServo
DRV_EncAutodetect%=1          // Abilitazione Autodetect dell'Encoder

// Encoder Motore
DRV_EncConnect%=0             // Selettore Retrazione Velocità su Connettore Drive
DRV_RefEncSource%=2           // Riferimento Encoder Slot 2 (5=non Configurato)
DRV_FbEncSource%=DRV_EncConnect% // Feedback Encoder Uguale a Selettore Retrazione Velocità

// Task Manager Register
DRV_SetCtSync%=1              // CtSync: 25=CtMaster 26=CtSlave
DRV_ClockThick%=0             // Tempo di Campionamento della Task Clock= 1 millisecondo
DRV_Pos0Thick%=2              // Tempo di Campionamento della Task Pos0 = 1 millisecondo
DRV_EnAutoRun%=1              // Auto-Run Enable
DRV_EnTripErr%=1              // Abilitazione Trip per Errori Run Time
DRV_EnAutSavPar%=1            // Abilitazione Salvataggio dei Parametri allo Spegnimento del
DRV_EnAutSavMenu20%=1         // Abilitazione Salvataggio del Menù 20
DRV_EnPos0Priority%=1          // Imposta la Task Pos0 come Task Prioritaria
DRV_EnQckRefSpd%=1            // Abilita l'aggiornamento veloce del Riferimento di Velocità
DRV_SpdRef%=0                 // Riferimento di velocità da POSLOOP
DRV_MaxIntSpeed%=INT(DRV_MaxSpeed) // Velocità Massima del Motore

// Inizializzazione IO
DRV_LogicaPNP%=1              // Logica Positiva PNP (da T24 a T29)
DRV_SpTermSmT12%=0            // Comando DO T12
DRV_SpTermSmT13%=0            // Comando DO T13
DRV_InvTermT24%=0             // Inversione Terminale T24
DRV_ParamTermT24=19.31         // Associazione Parametro Terminale T24
DRV_DoTermT24%=1              // Definizione DI o DO Terminale T24 (0=DI 1=DO)
DRV_InvTermT25%=0             // Inversione Terminale T25
DRV_ParamTermT25=19.32         // Associazione Parametro Terminale T25
DRV_DoTermT25%=1              // DefinizioDRV_CtNetMst%ne DI o DO Terminale T25 (0=DI 1=DO)
DRV_InvTermT26%=0             // Inversione Terminale T26
DRV_ParamTermT26=19.33         // Associazione Parametro Terminale T26
DRV_DoTermT26%=0              // Definizione DI o DO Terminale T26 (0=DI 1=DO)
DRV_InvTermT27% = 0           // Inversione Terminale T27
DRV_ParamTermT27=19.34         // Associazione Parametro Terminale T27
DRV_DisSetAutT28eT29%=1       // Disabilitazione Impostazione Automatica T28 e T29
DRV_InvTermT28%=0             // Inversione Terminale T28
DRV_ParamTermT28=19.35         // Associazione Parametro Terminale T28
DRV_InvTermT29%=0             // Inversione Terminale T29
DRV_ParamTermT29=19.36         // Associazione Parametro Terminale T29
DRV_InvTermTRele%=0           // Inversione Terminale T41 (Relè)
DRV_ParamTermRele=00.00        // Associazione Parametro Terminale T41 (Relè)

```

		Matrix.DPL	14/01/2016
			Page 7

```

DRV_InvTermT22%=1 // Inversione Terminale T22
DRV_ParamTermT22=00.00 // Associazione Parametro Terminale T22
DRV_InvTermT31%=0 // Inversione Terminale T31 (Hardware Enable%)
DRV_InvTermT05%=0 // Inversione Terminale T05
DRV_ParamTermT05=00.00 // Associazione Parametro Terminale T05
DRV_TypTermT07%=6 // Tipo Terminale T07 (0=0.20mA,4=4.20mA,6=Volt)
DRV_InvTermT07%=0 // Inversione Terminale T07
DRV_ParamTermT07=00.00 // Associazione Parametro Terminale T07
DRV_TypTermT08%=6 // Tipo Terminale T08 (0=0.20mA,4=4.20mA,6=Volt)
DRV_InvTermT08%=0 // Inversione Terminale T08
DRV_ParamTermT08=00.00 // Associazione Parametro Terminale T08
DRV_ParamTermT09=03.02 // Associazione Parametro Terminale T09 (AO)
DRV_TypTermT09%=3 // Tipo Terminale T09 (0=Volt,1=0.20mA,2=4.20mA)
DRV_ParamTermT10=04.02 // Associazione Parametro Terminale T10 (AO)
DRV_TypTermT10%=3 // Tipo Terminale T10 (0=Volt,1=0.20mA,2=4.20mA)

// Inizializzazione CtNet
DRV_CtNetTokenRing3%=0 // Se è presente un'unica rete CTNet lasciare a 0
DRV_CtNetAddress%=0 // da 1 a 255, 0 disabilita la CtNet locale
DRV_CtNetSpd%=1 // 0=5 Mbit/s 1=2.5 Mbit/s 2=1.25Mbit/s
DRV_CtNetMst%=0 // Solo Master: 1020=dove 100ms per le comunicazioni lente e 20ms
per le veloci

// Inizializzazione Freeze Encoder Motore
// FRZ_EnDiRealEncoder%=1 // Abilitazione dell'ingresso Dig In0 come comando di Freeze
Encoder Retroazione
// FRZ_SelFnRealEncoder%=0 // Freeze su Fronte Negativo Encoder di Retroazione
// FRZ_EnRealEncoder%=1 // Abilitazione Freeze Encoder di Retroazione
// FRZ_RichRealEncoder%=0 // Richiesta e Feedback Freeze Encoder di Retroazione

// Inizializzazione Freeze Encoder Esterno
FRZ_EnDiExtEncoder%=1 // Abilitazione dell'ingresso Dig come comando di Freeze Encoder
di Riferimento
FRZ_SelFnExtEncoder%=0 // Freeze su Fronte Negativo Encoder di Riferimento
FRZ_EnExtEncoder%=1 // Abilitazione Freeze Encoder di Riferimento
FRZ_RichExtEncoder%=0 // Richiesta e Feedback Freeze Encoder di Riferimento

//-----

} //InizParametriDrive:
InizParametriProgramma: {

// Simulazione Master
VrtMst_En%=0
VrtMst_Run%=0
VrtMst_AccMs%=500
VrtMst_DecMs%=500
VrtMst_RefRpm%=0
VrtMst_Encoder%=0
VrtMst_ActSpdRpm%=0
VrtMst_RefMtMin%=0

//-----
// Variabili di Sistema
SwEnable%=0 // Variabile Definita in Menu Esterno
Modality%=0 // Variabile Definita in Menu Esterno
ModalityChng%=0
SelectProfile%=0
SelectProfileChng%=0
BackCycleMsec%=0
DisableProfibus%=0
CtSync_Spare%=0
CtSync_Coeff%=1 // Coefficiente di ritardo CtSync (default 1 ciclo)
CtSync_Status%=0

Info_SmAppSwRel%=INT(DRV_SmAppSwRel*100)

```

		Matrix.DPL	14/01/2016
			Page 8

```

Info_Commissa%=0
//-----
// Variabili di JOG
Jog_CW%=0           // Variabile Definita in Menu Esterno
Jog_CCW%=0          // Variabile Definita in Menu Esterno
Jog_InProg%=0        // Variabile Definita in Menu Esterno
// Jog_SpdRpm%=100    // Variabile Definita in Menu Esterno
// Jog_AccDecMs%=500  // Variabile Definita in Menu Esterno
//-----
// Variabili Homing
Hom_Step%=0
Hom_Enable%=0        // Variabile Definita in Menu Esterno
Hom_Run%=0           // Variabile Definita in Menu Esterno
Hom_InProg%=0        // Variabile Definita in Menu Esterno
Hom_Ok%=0            // Variabile Definita in Menu Esterno
// Hom_SpdRpm%=100    // Variabile Definita in Menu Esterno
// Hom_AccDecMs%=500  // Variabile Definita in Menu Esterno
Hom_FrzPos%=0
Hom_Px%=0
//-----
// Variabili Albero Elettrico
Egb_Step%=0
Egb_AtSpeed%=0
// Egb_AccMs%=5000    // Variabile Definita in Menu Esterno
// Egb_DecMs%=5000    // Variabile Definita in Menu Esterno

// Rigido
// Egb_RcAccMs%=500    // Variabile Definita in Menu Esterno
// Egb_RcDecMs%=500    // Variabile Definita in Menu Esterno
// Egb_RcSpdRpm%=500   // Variabile Definita in Menu Esterno
Egb_RcMstEncoderDlt%=0
Egb_RcSlvEncoderDlt%=0
Egb_RcMstEncoderIniz%=0
Egb_RcSlvEncoderIniz%=0

// Applicazione
Egb_RcEn%=0
Egb_RcFault%=0
Egb_RcActPosFrz%=0
Egb_RcLavParamEn%=0
Egb_DistRcSlvMstCn%=0

// Offset
// QuotaOfst%=0        // Variabile Definita in Menu Esterno
QuotaOfstMCn%=0
QuotaOfstSCnIm%=0
//-----
// Variabili Moduli
Mst_Enc%=0
Mst_Delta%=0
Mst_ImEnc%=0

Mtr_Enc%=0
Mtr_Delta%=0
Mtr_ImEnc%=0

PzInS%=0
PzOutS%=0
PzOutSChng%=0
PzInM%=0
PzOutM%=0

UsrSlv_EncW%=0
DIM DistFpSlvCn%[10]
DIM UsrSlv_En%[10]
DIM UsrSlv_Enc%[10]

```

```

DIM UsrSlv_Delta%[10]
DIM UsrSlv_ImEnc%[10]
DIM UsrSlv_EncFrz%[10]

UsrMst_Res%=0
UsrMst_EncW%=0
DIM DistFpMstCn%[10]
DIM UsrMst_En%[10]
DIM UsrMst_Enc%[10]
DIM UsrMst_Delta%[10]
DIM UsrMst_ImEnc%[10]
DIM UsrMst_EncFrz%[10]
//-----
// Variabili Loop di Posizione
PsLp_Ref%= 0
PsLp_Enable%=0
PsLp_ErrPos%=0
PsLp_PidOut%=0
// Gain%=9000 // Variabile Definita in Menu Esterno
// FdFrwd%=989 // Variabile Definita in Menu Esterno
// Max_Error%=100 // Variabile Definita in Menu Esterno
//-----
// Variabili Profilo di Velocità
PrSPD_SpdCn%=0
PrSPD_AccCn=0
PrSPD_DecCn=0
//-----
// Variabili Profilo di Posizione
PrPOS_Ref%=0
PrPOS_ActAcc%=0
PrPOS_ActPos%=0
PrPOS_ActSpd%=0
PrPOS_SpdCn%=0
PrPOS_AccCn%=0
PrPOS_DecCn%=0
PrPOS_Jerck%=0
PrPOS_InProg%=0
PrPOS_ToTarget%=0
//-----
// Variabili Profilo Albero Elettrico
// PrEGB_Den%=1
// PrEGB_Num%=1
PrEGB_AccDec%=0
PrEGB_ActPos%=0
PrEGB_CoeffRamp%=0
PrEGB_ActSpd%=0
PrEGB_FloatInc%=0
PrEGB_MstEncInc=0
PrEGB_MstEnc%=0
PrEGB_MstRif%=0
PrEGB_TrgtRamp%=0
PrEGB_RampInit%=1
PrEGB_EndAccDec%=0
PrEGB_MstStopMod%=0
PrEGB_RampEnable%=0
PrEGB_RcRef%=0
PrEGB_RcActAcc%=0
PrEGB_RcActPos%=0
PrEGB_RcActSpd%=0
PrEGB_RcSpdCn%=0
PrEGB_RcAccCn%=0
PrEGB_RcDecCn%=0
//-----
// Variabili Matrix

// IO

```

		Matrix.DPL	14/01/2016
			Page 10

```

EnAut%=0
EnStaz%=0
// DI_SaAut%=0 // $ (work assente: Debug Cmnd Simulati)
// DI_SbJog_Hom%=0 // $ (work assente: Debug Cmnd Simulati)
SbStart%=0
SaAutChng%=0
Asse_Ready%=0
Fault_MancanzaBusta%=0
//-----

// HP
Hp_Gen%=0
Hp_Step%=0
Hp_Dt%=0
Hp_Ok%=0
Hp_StepEn%=0
Hp_GenFlag%=0

// Last
Clk_100mS%=0
Clk_100mSim%=0
Clk_200mS%=0
Clk_200mSim%=0
Clk_400mS%=0
Clk_400mSim%=0
Clk_500mS%=0
Clk_500mSim%=0
Clk_800mS%=0
Clk_800mSim%=0
Clk_1S%=0
Clk_1Sim%=0
Clk_2S%=0
Clk_2Sim%=0
//-----

// Varie
Sim_FtcM%=0
Sim_EncM%=0
Sim_FtcS%=0
Sim_EncS%=0
FRZ_RichRealEncoder%=0
FRZ_RichRealEncoderFp%=0
FRZ_RichExtEncoderFp%=0
// DRV_ExtEncoder%=0 // $ (work assente: Debug Encoder Simulato)
// FRZ_PosEncExtEncoder%=0 // $ (work assente: Debug Encoder Simulato)

Dbg_01%=0
Dbg_02%=0
Dbg_03%=0
Dbg_04%=0
Dbg_05%=0
Dbg_06%=0
Dbg_07%=0
Dbg_08%=0
Dbg_09%=0
Dbg_10%=0

Dbg_11%=1
Dbg_12%=0
Dbg_13%=0
Dbg_14%=0
Dbg_15%=0
Dbg_16%=0
Dbg_17%=0
Dbg_18%=0
Dbg_19%=0
Dbg_20%=0

```

		Matrix.DPL	14/01/2016
			Page 11

```

    Dbg_21%=0
    Dbg_22%=0
    Dbg_23%=0
    Dbg_24%=0
    Dbg_25%=0
    Dbg_26%=0
    Dbg_27%=0
    Dbg_28%=0
    Dbg_29%=0
    Dbg_30%=0

} //InizParametriProgramma:
//.....
Background {
Top:
    StartTime_msec%=TIME
    CALL Main:
    CALL Last:
//-----
    CALL Calcoli:
    CALL Visualizzazioni:
//  CALL Debug:      // $ (work assente: Debug FTC)
//-----
// Enable Software
    DRV_Enabled%= TON((SwEnable% & DRV_HwEnable%),250)
    IF DRV_Enabled%=0 THEN
        Hom_Ok%=0
        Modality%=0
        SelectProfile%=0
    ENDIF

BackCycleMsec%=(TIME - StartTime_msec%)
GOTO Top:
} //Background
Calcoli: {

    // Quantità Impulsi per Unità di Misura Motore
    CnUnitSlv=Ratio_EncIntS%+(Ratio_EncDecS%/100000000.0)

    // Quantità Impulsi per Unità di Misura Encoder Esterno
    CnUnitMst=Ratio_EncIntM%+(Ratio_EncDecM%/100000000.0)

    // Calcolo Ratio EGB
    PrEGB_Num%=INT (CnUnitSlv*1000000)
    PrEGB_Den%=INT (CnUnitMst*1000000)

//-----
    PsLp_Gain%=Gain% * 10
    PsLp_FdFrwd%=FdFrwd%
//-----

    // TRASFORMAZIONE VELOCITA' E RAMPE IN COUNT

    // Modalità JOG
    Jog_SpdRpmMl%=Jog_SpdRpm%*1000
    Jog_AccCn=FLOAT (Jog_AccDecMs%)/DRV_MaxSpeed
    Jog_DecCn=FLOAT (Jog_AccDecMs%)/DRV_MaxSpeed

    // Modalità HOMING
    Hom_SpdSpdRpmMl%=Hom_SpdRpm%*1000
    Hom_PosSpdCn%=Hom_SpdRpm% * Encoder_PPR% / 60
    Hom_SpdAccCn=FLOAT (Hom_AccDecMs%)/DRV_MaxSpeed
    Hom_SpdDecCn=FLOAT (Hom_AccDecMs%)/DRV_MaxSpeed
    (Hom_PosAccCn%, Hom_PosDecCn%,Hom_Jerck%)=_ConvertRamp (Hom_AccDecMs%, Hom_AccDecMs%, Jerck%, 0)

```

		Matrix.DPL	14/01/2016
			Page 12

```

// Modalità ALBERO ELETTRICO
(Egb_AccCn%, Dummy%, Dummy%) = _ConvertRamp (Egb_AccMs%, 1, 0, 1)
// Rampa Accelerazione Albero Elettrico
(Egb_DecPosCn%, Egb_DecSpdCn%, Dummy%) = _ConvertRamp (Egb_DecPosMs%, Egb_DecSpdMs%, 0, 1)
// Rampe Decelerazione Albero Elettrico
Egb_RcInizSpdCn% = Egb_RcInizSpdRpm% * Encoder_PPR% / 60
// Velocità Recupero Iniziale
(Egb_RcInizAccCn%, Egb_RcInizDecCn%, Dummy%) = _ConvertRamp (Egb_RcInizAccDecMs%, Egb_RcInizAccDecMs%, 0,
0) // Rampe Recupero Iniziale
(Egb_RcLavAccCn%, Egb_RcLavDecCn%, Dummy%) = _ConvertRamp (Egb_RcLavAccDecMs%, Egb_RcLavAccDecMs%, 0, 0)
// Rampe Recupero Lavoro
Egb_RcLavSpazioCn% = INT(0.5 * PrEGB_ActSpd% * PrEGB_ActSpd% / Egb_RcLavDecCn%)
// Calcolo Spazio Necessario Per Rampa Recupero Lavoro
Egb_RcFinalSpdCn% = Egb_RcFinalSpdRpm% * Encoder_PPR% / 60
// Velocità Recupero Finale
(Egb_RcFinalAccCn%, Egb_RcFinalDecCn%, Dummy%) = _ConvertRamp (Egb_RcFinalAccDecMs%,
Egb_RcFinalAccDecMs%, 0, 0) // Rampe Recupero Finale
Egb_QuotaAggancioCn% = INT(((QuotaFtcMst%-(QuotaFtcSlv%-QuotaStandBySlv%))*10) * CnUnitMst)
// Quota di Aggancio Albero Elettrico (mm)

// Master Simulato
VrtMst_RefRpm% = INT(VrtMst_RefMtMin% / 0.19886281497 ) //
Impostazione Velocità in m/min (Circonferenza Ruota in metri)

// PARAMETRI PROGRAMMA
QuotaFtcMstCn% = INT(QuotaFtcMst%*CnUnitMst*10) // Distanza
tra Ftc Master e Punto di Accoppiamento (mm)
QuotaFtcSlvCn% = INT(QuotaFtcSlv%*CnUnitSlv*10) // Distanza
tra Ftc Slave e Punto di Accoppiamento (mm)
QuotaStandBySlvCn% = INT(QuotaStandBySlv%*CnUnitSlv*10) // Distanza
tra Ftc Slave e Punto di StandBy (mm)
QuotaAccoppCn% = INT(QuotaAccopp%*CnUnitSlv*10) // Distanza
tra Inizio e Fine Accoppiamento (mm)
IF QuotaOfst%<0 THEN QuotaOfst%=0
QuotaOfstSCn% = INT(QuotaOfst%*CnUnitSlv) // Quota di
Sfasamento Slave(0.1mm)
QuotaOscuramSlvCn% = INT(QuotaOscuramSlv%*CnUnitSlv) // Quota di
Oscuramento Slave(0.1mm)
QuotaOscuramMstCn% = INT(QuotaOscuramMst%*CnUnitMst) // Quota di
Oscuramento Slave(0.1mm)

} //Calcoli:
Visualizzazioni: {

// Stato Asse
IF DRV_Ok%=1 AND Asse_Ready%=0 AND Fault_MancanzaBusta%=0 THEN
CT_Status_00%=1
CT_Status_01%=1
ENDIF
IF Asse_Ready%=1 AND DRV_Ok%=1 AND Fault_MancanzaBusta%=0 THEN
CT_Status_00%=0
CT_Status_01%=1
ENDIF
IF Fault_MancanzaBusta%=1 AND DRV_Ok%=1 THEN
CT_Status_00%=1
CT_Status_01%=0
ENDIF
IF DRV_Ok%=0 THEN
CT_Status_00%=0
CT_Status_01%=0
ENDIF

// Lettura Quota Motore (0.1mm)

```

		Matrix.DPL	14/01/2016
			Page 13

```

Rd_ActPosS%=INT (Mtr_Enc%/CnUnitSlv)

// Lettura Quota Encoder Esterno (0.1mm)
Rd_ActPosM%=INT (Mst_Enc%/CnUnitMst)

// Lettura Quota Correzione Slave-Master (0.1mm)
DistRcSlvMst%=INT (Egb_DistRcSlvMstCn%/CnUnitSlv)

} //Visualizzazioni:
Debug: {
  //-----
  IF Sim_FtcM% = 1 THEN
    IF Sim_EncM% > ((Sim_LunghM%+Sim_SpazioM%)*10*CnUnitMst) THEN
      Sim_EncM%=0
    ELSEIF Sim_EncM% < (Sim_LunghM%*10*CnUnitMst) THEN
      DRV_SpTermSmT13%=1
    ELSEIF Sim_EncM% > (Sim_LunghM%*10*CnUnitMst) THEN
      DRV_SpTermSmT13%=0
    ENDIF
  ELSE
    Sim_EncM%=0
  ENDIF
  FRZ_RichExtEncoder%=DI_FtcMst% // $ ( Debug FTC Mst)

  IF Sim_FtcS% = 1 THEN
    IF Sim_EncS% > ((Sim_LunghS%+Sim_SpazioS%)*10*CnUnitSlv) THEN
      Sim_EncS%=0
    ELSEIF Sim_EncS% < (Sim_LunghS%*10*CnUnitSlv) THEN
      DRV_SpTermSmT12%=1
    ELSEIF Sim_EncS% > (Sim_LunghS%*10*CnUnitSlv) THEN
      DRV_SpTermSmT12%=0
    ENDIF
  ELSE
    Sim_EncS%=0
  ENDIF
} //-----
} //Debug:
//.....
Main: {

// Inizializzazione
// Condizioni di Ripristino Programma
IF DI_SaAut% <> SaAutChng% THEN Hp_Gen%=1
SaAutChng%=DI_SaAut%

// Richiesta Ripristino Programma
IF Hp_Gen%=1 THEN CALL InitPLC:

// Dopo 1 Secondo dalla Richiesta di Ripristino Avvio l'Homing
Hp_Dt%=TON(!Hp_Gen%,500)
IF Hp_Dt%=1 THEN CALL Hp:

// Ciclo Automatico
SbStart% = DI_SaAut% & DI_SbJog_Hom%
EnAut% = (SbStart% | EnAut% ) & DI_SaAut% & !Hp_Gen%
EnStaz% = TON(EnAut%,50)

// Ciclo Manuale
SbMan% = TON(!DI_SaAut% & DI_SbJog_Hom%,500)
IF SbMan%=1 THEN RichMan%=1
IF RichMan%=1 THEN SwEnable%=1
EnMan% = RichMan% & TON(DRV_Enabled%,500)
IF EnMan%=1 AND DI_SbJog_Hom%=1 THEN
  Modality%=1

```

```

        Jog_CW%=1
    ELSE
        Jog_CW%=0
    ENDIF

// Segnalazioni ad HMI

} //Main:
InitPLC: {

    SwEnable%=0
    EnAut%=0
    Hp_Ok%=0
    Hp_Gen%=0
    Hp_GenFlag%=0
    Hp_Step%=0

    Asse_Ready%=0

    EnMan%=0
    RichMan%=0
    Jog_CW%=0

    Egb_RcEn%=0

    PzInS%=0
    PzOutS%=0
    UstrSlv_En%[1]=0
    UstrSlv_En%[2]=0
    UstrSlv_En%[3]=0
    UstrSlv_En%[4]=0
    UstrSlv_En%[5]=0
    UstrSlv_En%[6]=0
    UstrSlv_Enc%[1]=0
    UstrSlv_Enc%[2]=0
    UstrSlv_Enc%[3]=0
    UstrSlv_Enc%[4]=0
    UstrSlv_Enc%[5]=0
    UstrSlv_Enc%[6]=0
    DistFpSlvCn%[1]=0
    DistFpSlvCn%[2]=0
    DistFpSlvCn%[3]=0
    DistFpSlvCn%[4]=0
    DistFpSlvCn%[5]=0
    DistFpSlvCn%[6]=0

    PzInM%=0
    PzOutM%=0
    UstrMst_En%[1]=0
    UstrMst_En%[2]=0
    UstrMst_En%[3]=0
    UstrMst_En%[4]=0
    UstrMst_En%[5]=0
    UstrMst_En%[6]=0
    UstrMst_Enc%[1]=0
    UstrMst_Enc%[2]=0
    UstrMst_Enc%[3]=0
    UstrMst_Enc%[4]=0
    UstrMst_Enc%[5]=0
    UstrMst_Enc%[6]=0
    DistFpMstCn%[1]=0
    DistFpMstCn%[2]=0
    DistFpMstCn%[3]=0
    DistFpMstCn%[4]=0
    DistFpMstCn%[5]=0

```

```

DistFpMstCn%[6]=0

} //InitPLC:
//-----
Hp: {

// CASE
IF EnStaz%=1 AND Hp_StepEn%=1 THEN
    Hp_Step%=Hp_Step%+1
    Hp_StepEn%=0
ENDIF
SELECT Hp_Step%
//-----
CASE 0 // INIZIALIZZAZIONE
    Hp_StepEn%=1
//-----
CASE 1 // RILASCIO PULSANTE HOMING
    IF DI_SbJog_Hom%=0 THEN Hp_StepEn%=1
//-----
CASE 2 // ABILITAZIONE HARDWARE
    SwEnable%=1
    Fault_MancanzaBusta%=0
    IF TON(DRV_Enabled%,300)=1 THEN Hp_StepEn%=1
//-----
CASE 3 // START HOMING
    Modality%=2
    Hom_Enable%=1
    Hom_Run%=1
    IF Hom_InProg%=1 THEN
        Hom_Run%=0
        Hp_Step%=4
    ENDIF
//-----
CASE 4 // ATTESA FINE HOMING
    IF Hom_Ok%=1 THEN Hp_StepEn%=1
//-----
CASE 5 // MODALITA' EGB
    Modality%=4
    Hp_Ok%=1
    Hp_StepEn%=1
//-----
CASE 6 // END
    Hp_End%=1
//-----
ENDSELECT
} //Hp:
//-----
Last: {

// Clock di Sistema
Clk_100mS%=TON(!Clk_100mSIm%,100)
Clk_100mSIm%=TON(Clk_100mS%,100)
Clk_200mS%=TON(!Clk_200mSIm%,200)
Clk_200mSIm%=TON(Clk_200mS%,200)
Clk_400mS%=TON(!Clk_400mSIm%,400)
Clk_400mSIm%=TON(Clk_400mS%,400)
Clk_500mS%=TON(!Clk_500mSIm%,500)
Clk_500mSIm%=TON(Clk_500mS%,500)
Clk_800mS%=TON(!Clk_800mSIm%,800)
Clk_800mSIm%=TON(Clk_800mS%,800)
Clk_1S%=TON(!Clk_1SIm%,1000)
Clk_1SIm%=TON(Clk_1S%,1000)
Clk_2S%=TON(!Clk_2SIm%,2000)
Clk_2SIm%=TON(Clk_2S%,2000)
//-----

```

		Matrix.DPL	14/01/2016
			Page 16

```

-----
} //Last:
//.....
Pos0 {
Pos0_Tim%=0

// CALL MasterSimulato:    // $ (work assente)

// Modalità
IF DRV_Enabled%=0 OR Modality% <> ModalityChng% THEN Call InitModality:
ModalityChng%=Modality%
CALL Moduli:
SELECT Modality%
CASE 1
CALL Jog:
CASE 2
CALL Homing:
CASE 4
IF Hom_Ok%=1 THEN CALL AlberoElettrico:
ENDSELECT
CALL Profili:

} //Pos0
MasterSimulato: {
//-----
-----

// (VrtMst_Encoder%, VrtMst_ActSpdRpm%) = _VirtualEncoder (VrtMst_Run%, VrtMst_RefRpm%, VrtMst_AccMs%,
VrtMst_DecMs%, DRV_MaxIntSpeed%)
IF VrtMst_Run% =1 THEN DRV_ExtEncoder%= DRV_ExtEncoder% - (VrtMst_RefRpm%/2)
//-----
-----

} //MasterSimulato:
//-----
Moduli: {

// Modulo Encoder Motore
Mtr_Delta%=DRV_RealEncoder%-Mtr_ImEnc%
Mtr_Enc%=Mtr_Enc%+Mtr_Delta%
Mtr_ImEnc%=DRV_RealEncoder%
//-----
// Modulo Master (Albero Elettrico)
Mst_Delta%=DRV_ExtEncoder%-Mst_ImEnc%
Mst_Enc%=Mst_Enc%-Mst_Delta%
Mst_ImEnc%=DRV_ExtEncoder%
//-----
// User Modulo Encoder SLV Pz 1
IF UstrSlv_En%[1]=1 THEN
UstrSlv_Enc%[1]= DRV_RealEncoder%-UstrSlv_EncFrz%[1]
IF PzOutS%=2 THEN
UstrSlv_Enc%[1]=0
UstrSlv_En%[1]=0
ENDIF
UstrSlv_Enc%[7]=UstrSlv_Enc%[1]
ENDIF
//-----
// User Modulo Encoder SLV Pz 2
IF UstrSlv_En%[2]=1 THEN
UstrSlv_Enc%[2]= DRV_RealEncoder%-UstrSlv_EncFrz%[2]
IF PzOutS%=3 THEN
UstrSlv_Enc%[2]=0
UstrSlv_En%[2]=0
ENDIF
ENDIF
//-----

```

		Matrix.DPL	14/01/2016
			Page 17

```

// User Modulo Encoder SLV Pz 3
IF UsrSlv_En%[3]=1 THEN
    UsrSlv_Enc%[3]= DRV_RealEncoder%-UsrSlv_EncFrz%[3]
    IF PzOutS%=4 THEN
        UsrSlv_Enc%[3]=0
        UsrSlv_En%[3]=0
    ENDIF
ENDIF
//-----
// User Modulo Encoder SLV Pz 4
IF UsrSlv_En%[4]=1 THEN
    UsrSlv_Enc%[4]= DRV_RealEncoder%-UsrSlv_EncFrz%[4]
    IF PzOutS%=5 THEN
        UsrSlv_Enc%[4]=0
        UsrSlv_En%[4]=0
    ENDIF
ENDIF
//-----
// User Modulo Encoder SLV Pz 5
IF UsrSlv_En%[5]=1 THEN
    UsrSlv_Enc%[5]= DRV_RealEncoder%-UsrSlv_EncFrz%[5]
    IF PzOutS%=6 THEN
        UsrSlv_Enc%[5]=0
        UsrSlv_En%[5]=0
    ENDIF
ENDIF
//-----
// User Modulo Encoder SLV Pz 6
IF UsrSlv_En%[6]=1 THEN
    UsrSlv_Enc%[6]= DRV_RealEncoder%-UsrSlv_EncFrz%[6]
    IF PzOutS%=1 THEN
        UsrSlv_Enc%[6]=0
        UsrSlv_En%[6]=0
    ENDIF
ENDIF
//-----
// User Modulo Encoder MST Pz 1
IF UsrMst_En%[1]=1 THEN
    UsrMst_Enc%[1]=UsrMst_EncFrz%[1]-DRV_ExtEncoder%
    IF PzOutM%=2 OR UsrMst_Res%=1 THEN
        UsrMst_Enc%[1]=0
        UsrMst_En%[1]=0
    ENDIF
    UsrMst_Enc%[7]=UsrMst_Enc%[1]
ENDIF
//-----
// User Modulo Encoder MST Pz 2
IF UsrMst_En%[2]=1 THEN
    UsrMst_Enc%[2]=UsrMst_EncFrz%[2]-DRV_ExtEncoder%
    IF PzOutM%=3 OR UsrMst_Res%=1 THEN
        UsrMst_Enc%[2]=0
        UsrMst_En%[2]=0
    ENDIF
ENDIF
//-----
// User Modulo Encoder MST Pz 3
IF UsrMst_En%[3]=1 THEN
    UsrMst_Enc%[3]=UsrMst_EncFrz%[3]-DRV_ExtEncoder%
    IF PzOutM%=4 OR UsrMst_Res%=1 THEN
        UsrMst_Enc%[3]=0
        UsrMst_En%[3]=0
    ENDIF
ENDIF
//-----
// User Modulo Encoder MST Pz 4
IF UsrMst_En%[4]=1 THEN

```

```

        UsrMst_Enc%[4]=UsrMst_EncFrz%[4]-DRV_ExtEncoder%
        IF PzOutM%=5 OR UsrMst_Res%=1 THEN
            UsrMst_Enc%[4]=0
            UsrMst_En%[4]=0
        ENDIF
    ENDIF
//-----
// User Modulo Encoder MST Pz 5
    IF UsrMst_En%[5]=1 THEN
        UsrMst_Enc%[5]=UsrMst_EncFrz%[5]-DRV_ExtEncoder%
        IF PzOutM%=6 OR UsrMst_Res%=1 THEN
            UsrMst_Enc%[5]=0
            UsrMst_En%[5]=0
        ENDIF
    ENDIF
//-----
// User Modulo Encoder MST Pz 6
    IF UsrMst_En%[6]=1 THEN
        UsrMst_Enc%[6]=UsrMst_EncFrz%[6]-DRV_ExtEncoder%
        IF PzOutM%=1 OR UsrMst_Res%=1 THEN
            UsrMst_Enc%[6]=0
            UsrMst_En%[6]=0
        ENDIF
    ENDIF
//-----
    UsrMst_Res%=0
//-----

// Calcolo Distanza Fronti e Abilitazione Modulo Slave

    IF DI_FtcSlv%=1 THEN FRZ_RichRealEncoder%=1
    IF FRZ_RichRealEncoder% > FRZ_RichRealEncoderFp% AND Modality%>0 THEN
        FRZ_PosEncRealEncoder%=DRV_RealEncoder%
        PzInS%=PzInS%+1: IF PzInS%>6 THEN PzInS%=1
        SELECT PzInS%
            CASE 1 : DistFpSlvCn%[6]=UsrSlv_Enc%[6] : UsrSlv_EncFrz%[1]=FRZ_PosEncRealEncoder% :
            CASE 2 : DistFpSlvCn%[1]=UsrSlv_Enc%[1] : UsrSlv_EncFrz%[2]=FRZ_PosEncRealEncoder% :
            CASE 3 : DistFpSlvCn%[2]=UsrSlv_Enc%[2] : UsrSlv_EncFrz%[3]=FRZ_PosEncRealEncoder% :
            CASE 4 : DistFpSlvCn%[3]=UsrSlv_Enc%[3] : UsrSlv_EncFrz%[4]=FRZ_PosEncRealEncoder% :
UsrSlv_En%[4]=1
            CASE 5 : DistFpSlvCn%[4]=UsrSlv_Enc%[4] : UsrSlv_EncFrz%[5]=FRZ_PosEncRealEncoder% :
UsrSlv_En%[5]=1
            CASE 6 : DistFpSlvCn%[5]=UsrSlv_Enc%[5] : UsrSlv_EncFrz%[6]=FRZ_PosEncRealEncoder% :
UsrSlv_En%[6]=1
        ENDSELECT
    ENDIF
    FRZ_RichRealEncoderFp%=FRZ_RichRealEncoder%

    IF DI_FtcSlv%=0 THEN
        UsrSlv_EncW%=UsrSlv_EncW%+Mtr_Delta%
        IF UsrSlv_EncW% > QuotaOscuramSlvCn% THEN FRZ_RichRealEncoder%=0
    ELSE
        UsrSlv_EncW%=0
    ENDIF
// -----
// Calcolo Distanza Fronti e Abilitazione Modulo Master
    IF FRZ_RichExtEncoder% > FRZ_RichExtEncoderFp% AND Modality%>0 THEN
        PzInM%=PzInM%+1: IF PzInM%>6 THEN PzInM%=1
        // FRZ_PosEncExtEncoder%=DRV_ExtEncoder% // $ (work assente: Debug FTCv Mst Simulato)
        SELECT PzInM%
            CASE 1 : DistFpMstCn%[6]=INT(UsrMst_Enc%[6]*CnUnitSlv/CnUnitMst) :
UsrMst_EncFrz%[1]=FRZ_PosEncExtEncoder% : UsrMst_En%[1]=1
            CASE 2 : DistFpMstCn%[1]=INT(UsrMst_Enc%[1]*CnUnitSlv/CnUnitMst) :
UsrMst_EncFrz%[2]=FRZ_PosEncExtEncoder% : UsrMst_En%[2]=1
            CASE 3 : DistFpMstCn%[2]=INT(UsrMst_Enc%[2]*CnUnitSlv/CnUnitMst) :

```

```

UsrMst_EncFrz%[3]=FRZ_PosEncExtEncoder% : UsrMst_En%[3]=1
CASE 4 : DistFpMstCn%[3]=INT(UsrMst_Enc%[3]*CnUnitSlv/CnUnitMst) :
UsrMst_EncFrz%[4]=FRZ_PosEncExtEncoder% : UsrMst_En%[4]=1
CASE 5 : DistFpMstCn%[4]=INT(UsrMst_Enc%[4]*CnUnitSlv/CnUnitMst) :
UsrMst_EncFrz%[5]=FRZ_PosEncExtEncoder% : UsrMst_En%[5]=1
CASE 6 : DistFpMstCn%[5]=INT(UsrMst_Enc%[5]*CnUnitSlv/CnUnitMst) :
UsrMst_EncFrz%[6]=FRZ_PosEncExtEncoder% : UsrMst_En%[6]=1
ENDSELECT
ENDIF
FRZ_RichExtEncoderFp% = FRZ_RichExtEncoder%

IF DI_FtcMst%=0 THEN
UsrMst_EncW%=UsrMst_EncW%-Mst_Delta%
IF UsrMst_EncW% > QuotaOscuramMstCn% THEN FRZ_RichExtEncoder%=0
ELSE
UsrMst_EncW%=0
ENDIF
// -----
//Inizializzazione Distanza Fronti
IF UsrSlv_En%[PzOutS%+1] > QuotaFtcSlvCn% THEN // Reset Variabili Distanza
Fronti Pezzi Slave già accoppiati
PzOutS%=PzOutS%+1: IF PzOutS%>6 THEN PzOutS%=1
SELECT PzOutS%
CASE 1 : DistFpSlvCn%[6]=0
CASE 2 : DistFpSlvCn%[1]=0
CASE 3 : DistFpSlvCn%[2]=0
CASE 4 : DistFpSlvCn%[3]=0
CASE 5 : DistFpSlvCn%[4]=0
CASE 6 : DistFpSlvCn%[5]=0
ENDSELECT
Egb_RcEn%=1
ENDIF

IF UsrMst_En%[PzOutM%+1] > (QuotaFtcMstCn%+QuotaOfstMCn%) THEN // Reset Variabili Distanza
Fronti Pezzi Master già accoppiati
PzOutM%=PzOutM%+1: IF PzOutM%>6 THEN PzOutM%=1
SELECT PzOutM%
CASE 1 : DistFpMstCn%[6]=0
CASE 2 : DistFpMstCn%[1]=0
CASE 3 : DistFpMstCn%[2]=0
CASE 4 : DistFpMstCn%[3]=0
CASE 5 : DistFpMstCn%[4]=0
CASE 6 : DistFpMstCn%[5]=0
ENDSELECT
ENDIF
//-----
-----

// IF Sim_FtcM% = 1 THEN Sim_EncM%=Sim_EncM%-Mst_Delta% // $ (work assente: Debug
// IF Sim_FtcS% = 1 THEN Sim_EncS%=Sim_EncS%+Mtr_Delta% // $ (work assente: Debug
FTC)
// -----
IF PzOutS% <> PzOutSChng% THEN // $ solo verifica tutto ok
(work assente)
Egb_RcFault%=0
IF PrEGB_RcRef% <> PrEGB_RcActPos% THEN Egb_RcFault%=1
PzOutSChng%=PzOutS%
ENDIF

} //Moduli:
//-----
InitModality: {
//-----
-----

Jog_InProg%=0

```

```

Hom_Run%=0
Hom_Step%=0
Hom_Enable%=0
Hom_InProg%=0

Pos_Ok%=0
Pos_Run%=0
Pos_Step%=0
Pos_InProg%=0
Pos_Enable%=0

Egb_Step%=0
Egb_AtSpeed%=0
//-----

} //InitModality:
Jog: {
//-----

    SelectProfile%=1                                // Profilo di Velocità
    PrSPD_AccCn=Jog_AccCn                            // Accelerazione Profilo di Velocità
    PrSPD_DecCn=Jog_DecCn                            // Decelerazione Profilo di Velocità
    IF Jog_CW%=1 AND Jog_CCW%=0 THEN                  // Jog Avanti
        PrSPD_SpdCn%=Jog_SpdRpmM1%
        Jog_InProg%=1
    ELSEIF Jog_CW%=0 AND Jog_CCW%=1 THEN              // Jog Indietro
        PrSPD_SpdCn%=Jog_SpdRpmM1%*-1
        Jog_InProg%=1
    ELSE
        PrSPD_SpdCn%=0                                // Velocità
        Jog_InProg%=0
    ENDIF

} //Jog:
Homing: {

// Arresto Homing in assenza di Enable Homing
IF Hom_Enable%=0 THEN
    Hom_Step%=0
    Hom_Run%=0
ENDIF
Hom_Case: :SELECT Hom_Step%
//-----
CASE 0
    PrSPD_SpdCn%=0
    PrPOS_Ref%=PsLp_Ref%
    IF Hom_Run%=1 THEN Hom_Step%=1
//-----
CASE 1                                // INIZIALIZZAZIONE
    Hom_Ok%=0
    Hom_InProg%=1
    SelectProfile%=1
    PrSPD_AccCn=Hom_SpdAccCn
    PrSPD_DecCn=Hom_SpdDecCn
    Hom_Step%=2
//-----
CASE 2                                // RICERCA DEL SENSORE
    PrSPD_SpdCn%=Hom_SpdSpdRpmM1%          // Velocità Ricerca Sensore
    IF UsrSlv_Enc[1] > 1 THEN
        Hom_FrzPos%=Mtr_Enc%
        Hom_Step%=3                        // all'FP Proximity
    ENDIF
//-----

```

		Matrix.DPL	14/01/2016
			Page 21

```

CASE 3
  PrSPD_SpdCn%=0
  IF TON(DRV_ZeroSpeed%,100)=1 THEN Hom_Step%=4
//-----
CASE 4
  SelectProfile%=2
  PrPOS_Ref%= Hom_FrzPos% + QuotaStandBySlvCn%
  PrPOS_SpdCn%=Hom_PosSpdCn%
  PrPOS_AccCn%=Hom_PosAccCn%
  PrPOS_DecCn%=Hom_PosDecCn%
  IF PrPOS_Ref%=PsLp_Ref% AND ABS(PsLp_ErrPos%) <=Max_Error% THEN Pos0_Tim%=1
  IF TON(Pos0_Tim%,100)=1 THEN Hom_Step%=5
//-----
CASE 5
  SelectProfile%=0
  Mtr_Enc%= UsrSlv_Enc %[1]
  Hom_Step%=6
//-----
CASE 6
  IF Hom_Run%=0 THEN
    Hom_Ok%=1
    Hom_InProg%=0
    Hom_Step%=0
  ENDIF
ENDIF

ENDSELECT
} //Homing:
AlberoElettrico: {
  Egb_Case: :SELECT Egb_Step%
//-----
CASE 0
  Asse_Ready%=1
  PrEGB_MstStopMod% = 0
  Egb_RcLavParamEn% = 0
  PrEGB_AccDec%=Egb_AccCn%
  Egb_Step%=1
  GOTO Egb_Case:
//-----
CASE 1
  QuotaOfstMCn% = INT(QuotaOfst%*CnUnitMst)
  IF UsrMst_Enc %[1] > (Egb_QuotaAggancioCn% + QuotaOfstMCn%) THEN
    QuotaOfstSCnIm%=QuotaOfstSCn%
    SelectProfile% = 3
    PrEGB_RampInit% = 1
    PrEGB_TrgtRamp% = 1000000
    Egb_RcMstEncoderIniz% = Mst_Enc%
    Egb_RcSlvEncoderIniz% = PsLp_Ref%
    Egb_Step%=2
    GOTO Egb_Case:
  ENDIF
//-----
CASE 2
  al volo (con Master in movimento)
  PrEGB_RampEnable% = 1
  Master all'aggancio ad Accelerazione EGB Terminata
  IF PrEGB_CoeffRamp% = 1000000 AND Egb_AtSpeed% = 0 THEN
    PrEGB_RcSpdCn% = Egb_RcInizSpdCn%
    PrEGB_RcAccCn% = Egb_RcInizAccCn%
    PrEGB_RcDecCn% = Egb_RcInizDecCn%
    Egb_RcMstEncoderDlt% = MULDIV((Mst_Enc% - Egb_RcMstEncoderIniz%), PrEGB_Num%, PrEGB_Den%)
    Egb_RcSlvEncoderDlt% = PsLp_Ref% - Egb_RcSlvEncoderIniz%
    PrEGB_RcRef% = PrEGB_RcRef% + Egb_RcMstEncoderDlt% - Egb_RcSlvEncoderDlt%
// Rampa Accelerazione Aggancio
// Recupero della Posizione

```

```

ENDIF
//-----
IF UsrSlv_Enc%[PzOutS%] > (QuotaFtcSlvCn% + QuotaAccoppCn%) AND Egb_RcEn%=1 THEN      //
Discriminazione Comportamento a fine accoppiamento
    Egb_DistRcSlvMstCn% = DistFpSlvCn%[PzOutS%] - DistFpMstCn%[PzOutM%]
    IF DistFpSlvCn%[PzOutS%]=0 THEN                                                    // Arresto causa
Slave (EGB in Rampa di decelerazione)
        PrEGB_AccDec%=Egb_DecSpdCn%
        PrEGB_TrgtRamp% = 0
        PrEGB_RcSpdCn% = 0
        Hp_GenFlag%=1
        Fault_MancanzaBusta%=1
        Egb_Step% = 4
    ELSEIF DistFpMstCn%[PzOutM%]=0 THEN                                                // Arresto causa
Master (EGB in Posizionamento a Busta successiva)
        PzInM%=0
        PzOutM%=0
        UsrMst_Res%=1
        PrEGB_AccDec%=Egb_DecPosCn%
        PrEGB_TrgtRamp% = 0
        PrEGB_RcSpdCn% = 0
        Egb_Step% = 3
    ELSE                                                                                // Recupero
sfasamento Slave-Master
        Egb_RcLavParamEn%=1
        Egb_RcActPosFrz% = PrEGB_RcActPos%
        PrEGB_RcRef% = PrEGB_RcRef% + Egb_DistRcSlvMstCn% - (QuotaOfstSCn% - QuotaOfstSCnIm%)

        QuotaOfstSCnIm% = QuotaOfstSCn%
ENDIF
Egb_RcEn%=0
ENDIF
IF Egb_RcLavParamEn%=1 THEN
    PrEGB_RcSpdCn%=PrEGB_ActSpd%
    IF PrEGB_RcSpdCn%<0 THEN PrEGB_RcSpdCn%=0
    IF Egb_RcActPosFrz% - PrEGB_RcActPos% > Egb_RcLavSpazioCn% AND ABS(PrEGB_RcRef%- PrEGB_RcActPos%)
> Egb_RcLavSpazioCn% THEN
        PrEGB_RcAccCn%= 1638400000
        PrEGB_RcDecCn%= 1638400000
    ELSE
        PrEGB_RcAccCn% = Egb_RcLavAccCn%
        PrEGB_RcDecCn% = Egb_RcLavDecCn%
    ENDIF
ENDIF
ENDIF
//-----
CASE 3
    IF PrEGB_ActSpd%=0 THEN Pos0_Tim%=1                                                // Posizionamento
Finale (Busta successiva)
    IF TON(Pos0_Tim%,100)=1 THEN
        PrEGB_RcSpdCn% = Egb_RcFinalSpdCn%
        PrEGB_RcAccCn% = Egb_RcFinalAccCn%
        PrEGB_RcDecCn% = Egb_RcFinalDecCn%
        PrEGB_RcRef% = PrEGB_RcActPos% + QuotaStandBySlvCn% - UsrSlv_Enc%[PzOutS%+1]
        Egb_Step% = 4
    ENDIF
//-----
CASE 4
    IF PrEGB_RcActSpd%=0 AND PrEGB_ActSpd%=0 THEN                                    // Ctrl Asse fermo
        IF Hp_GenFlag%=1 THEN Hp_Gen%=1
        PrEGB_CoeffRamp% = 0
        Hp_GenFlag%=0
        Egb_Step% = 0
    ENDIF
//-----
ENDSELECT
Egb_AtSpeed% = 0                                                                    // Segnalazioni

```

		Matrix.DPL	14/01/2016
			Page 23

```

IF PrEGB_CoeffRamp% = 1000000 THEN Egb_AtSpeed% = 1

} //AlberoElettrico:
//-----
Profili: {

    // Allineamento Contatori
    IF DRV_Enabled%=0 OR SelectProfile% <> SelectProfileChng% THEN
        PsLp_Enable%=0 // All'inizializzazione disabilito il Loop di Posizione
        PsLp_Ref%=Mtr_Enc% // Allineo il Target del POSLOOP con la Posizione Attuale
        DRV_DisRamp%=0 // Disabilitazione Rampe da drive
        PrSPD_SpdCn%=0 // Opzionale: al cambio profilo imposto a 0 il Target di
Velocità del Profilo di Velocità
        PrPOS_ActSpd%=0 // Imposto a 0 la Velocità del Profilo di Posizione
        PrPOS_Ref%=Mtr_Enc% // Allineo il Target del Profilo di Posizione con la
Posizione Attuale
        PrPOS_ActPos%=Mtr_Enc% // Allineo la Posizione Attuale del Profilo di Posizione
con la Posizione Attuale
        PrEGB_ActPos%=Mtr_Enc% // Allineo la Posizione Attuale del Recupero del Profilo
di Albero Elettrico con la Posizione Attuale
        PrEGB_MstEnc%=Mst_Enc% // Allineo l'immagine dell'Encoder Master per
inizializzare il modulo Master Albero Elettrico
        PrEGB_RampInit%=1 // Ripristino Rampe Albero Elettrico
        PrEGB_RcActSpd%=0 // Imposto a 0 la Velocità del Profilo di Recupero Albero
Elettrico
        PrEGB_RcRef%=0 // Imposto a 0 il Target del Profilo di Recupero Albero
        PrEGB_RcActPos%=0 // Imposto a 0 la posizione Attuale del Profilo di
Recupero Albero Elettrico
    ENDIF
    SelectProfileChng%=SelectProfile%
    SELECT SelectProfile%
//-----
CASE 1 // Profilo di Velocità
    IF TON(DRV_DisRamp%,5)=1 THEN DRV_SpdRef%=PrSPD_SpdCn%
    DRV_DisRamp%=1
    GOTO EndProfile:
//-----
CASE 2 // Profilo di Posizione
    //
    (PrPOS_InProg%,PrPOS_ActPos%,PrPOS_ActSpd%,PrPOS_ActAcc%,PrPOS_ToTarget%)=SSRAMP(PrPOS_Ref%,PrPOS_ActPos%,P
rPOS_SpdCn%,PrPOS_AccCn%,PrPOS_Jerck%,1) // SSRAMP
    (PrPOS_ActPos%, PrPOS_ActSpd%, PrPOS_ActAcc%)=SRAMP(PrPOS_Ref%, PrPOS_ActPos%, PrPOS_ActSpd%,
PrPOS_SpdCn%, PrPOS_SpdCn%, PrPOS_AccCn%, PrPOS_DecCn%) // SRAMP
    PsLp_Ref%=PrPOS_ActPos%
//-----
CASE 3 // Profilo Albero Elettrico
    IF PrEGB_MstStopMod% = 1 THEN PrEGB_MstRif% = PrEGB_MstRif% + (Mst_Enc% - PrEGB_MstEnc%)
    IF PrEGB_RampEnable% = 1 THEN
        PrEGB_MstEncInc = FLOAT(Mst_Enc% - PrEGB_MstEnc%)
        (PrEGB_CoeffRamp%, PrEGB_EndAccDec%) = SLEW(PrEGB_TrgtRamp%, PrEGB_AccDec%, PrEGB_RampInit%)
        PrEGB_RampInit%=0
        PrEGB_FloatInc% = INT(MULDIV(PrEGB_MstEncInc, PrEGB_CoeffRamp%, 1000000))
        PrEGB_MstRif% = PrEGB_MstRif% + PrEGB_FloatInc%
    ENDIF
    PrEGB_MstEnc% = Mst_Enc%
    (PrEGB_ActSpd%, PrEGB_ActPos%) = EGB2(1, PrEGB_MstRif%, PrEGB_Num%, PrEGB_Den%, PrEGB_ActPos%) //
Albero Elettrico
    (PrEGB_RcActPos%, PrEGB_RcActSpd%, PrEGB_RcActAcc%) = SRAMP(PrEGB_RcRef%, PrEGB_RcActPos%,
PrEGB_RcActSpd%,PrEGB_RcSpdCn% , PrEGB_RcSpdCn%, PrEGB_RcAccCn%, PrEGB_RcDecCn%) // Recupero Albero
    PsLp_Ref% = PrEGB_ActPos% + PrEGB_RcActPos% //
Aggiornamento Target a Posloop
//-----
ENDSELECT // Loop di Posizione
    (PsLp_PidOut%, PsLp_ErrPos%)=POSLOOP(PsLp_Ref%, Mtr_Enc%, 0, PsLp_Gain%, 0, PsLp_FdFrwd%, 812000,
Encoder_PPR%, 2)
    DRV_SpdRef% =LIMIT(IMUX2(0, PsLp_PidOut%, PsLp_Enable%), (DRV_MaxIntSpeed% * 1000))

```

```
PsIp_Enable%=1
EndProfile:

} //Profili:
//.....
Error {

IF DRV_DebugInfo_Err% > 0 THEN
    Info_DebugLineIm%=DRV_DebugInfo_Line%
    Info_DebugErrIm%=DRV_DebugInfo_Err%
    Info_DebugTaskIm%=DRV_DebugInfo_Task%
ENDIF

} //Error
//.....
```