Interface "graph" between object and PlcPgm

interface graphnamer
SpeedOp: Sub
SpeedOp: $PlcFo.SpeedOp
RefSpeed: $PlcMan.RefSpeed

Ap $PlcFo.RefSpeed → StoAr $PlcMan.RefSpeed

Dp $PlcFo.ReadyRemote → StoDp $PlcMan.ReadyRemote

Paradigm Object
Promotes communication.

Class-BaseFcPPO5Fo-Code			
W			
04-APR-2008 16:09	Proview	JE	1(5)

Ap $PlcMan.PowerUpTime → StoAp $PlcMan.PowerUpTime.TimerTime
Dp $PlcFo.PowerOn → Wsb PowerUpTime ≥1 → Or Out2

Normally Ready after power up startup

Dp $PlcMan.ConverterReady → And StartUpOp

Power starting up

StoDp $PlcMan.PowerStartUp
Show in HMI

Converter ready for remote control

S $PlcMan.StatusWordSW → MaskToD Mask out0 out1 out2 out3 SW_1 → And Ready → StoDp $PlcFo.ReadyRemote

Bit
Bit 0 = 1 : Ready
Bit 1 = 1 : Remote control activated
Bit 2 = 1 : Error(s) not reset

And AlarmOk → StoDp $PlcMan.AlarmOk

Class-BaseFcPPO5Fo-Code			
W			
04-APR-2008 16:09	Proview	JE	2

Preset and control

Dp $PlcFo.ReadyRemote
Dp $PlcFo.Start → And Start → DoMask CI Mask BI CW → StoIo $PlcMan.ControlWordCW

Interface control converter

Dp $PlcMan.OpResp → Timer ResetTime ≥1

Bit
Bit 0 = 1 : Ok, ok, ID-offset = 1
Bit 1 = 1 : Reset(ID-offset)

S $PlcMan.StatusWordSW → MaskToD Mask out0 SW_2

Converter run

StoDp $PlcFo.Run

Bit
Bit 0 = 1 : Run

Alarm

Dp $PlcMan.AlarmLocalMod
FirstScan
Dp $PlcMan.PowerStartUp
Dp $PlcMan.ConvertOk
Dp $PlcMan.ConvertRemote
Dp $PlcFo.PowerOn → Wsb Width1 ≥1 → And Local

CSoOp Dlsup1.DetectFirst
con
Dlsup Dlsup5 Local mode...
B-alarm

Class-BaseFcPPO5Fo-Code			
W			
04-APR-2008 16:09	Proview	JE	3

FirstScan

Dp $PlcMan.AlarmTripped
Dp $PlcMan.AlarmOk
Dp $PlcMan.ConvertTripped → And Tripped

CSoOp Dlsup1.DetectFirst
con
Dlsup Dlsup1 Tripped...

Dp $PlcMan.AlarmWarning
Dp $PlcMan.AlarmOk
Dp $PlcMan.ConvertWarning → And Warning

CSoOp Dlsup1.DetectFirst
con
Dlsup Dlsup2 Warning...
B-alarm

Dp $PlcMan.AlarmError
Dp $PlcMan.AlarmOk
Dp $PlcMan.ConvertOnInitial → And Error

CSoOp Dlsup1.DetectFirst
con
Dlsup Dlsup3 Error

S $PlcMan.StatusWordSW → MaskToD Mask out1 out2 out3 out4 out5 out10 SW_2

StoDp $PlcMan.ConvertReady
StoDp $PlcMan.ConvertRun
StoDp $PlcMan.ConvertTripped
StoDp $PlcMan.ConvertOnInitial
StoDp $PlcMan.ConvertWarning
StoDp $PlcMan.ConvertRemote

Status, HMI

Show converter status in HMI

Or Out1
Or Out3 → And Initline → StoDp $PlcMan.IndWarning

Bit
Bit 0 = 1 : Ready
Bit 1 = 1 : Remote control activated
Bit 2 = 1 : Error(s) not reset

Bit
Bit 0 = 1 : Warning

StoDp $PlcMan.IndError

Class-BaseFcPPO5Fo-Code			
W			
04-APR-2008 16:09	Proview	JE	4

Dp $PlcMan.OpResPar → Timer Timer1 ≥1
Dp $PlcMan.OpInitPar → Timer Timer2 ≥1
Dp $PlcMan.OpBackupPar → Timer Timer3 ≥1

Ip $PlcMan.RequestPar
Ip $PlcMan.WriteValuePar
Ip $PlcMan.IntoPar
Ip $PlcMan.IntoPar
Ip $PlcMan.InValuePar

CAbBlk
I1 I2 I3 I4 I5 I6 I7 I8 I9 I10 I11 I12 I13 I14 I15 I16
O1 O2 O3 O4 O5 O6 O7 O8 O9 O10 O11 O12 O13 O14 O15 O16

```
if (I8.Bit.07 && I07)
  {O1 = (I8.Bit.07 && I07) << 8;
  O2 = (I8.Bit.07 && I07) << 8;}
else if (I8.Bit.07 && !I07)
  {O1 = (I8.Bit.07 && I07) << 8;
  O2 = (I8.Bit.07 && I07) << 8;}
else if (!I8.Bit.07 && I07)
  {O1 = (I8.Bit.07 && I07) << 8;
  O2 = (I8.Bit.07 && I07) << 8;}
else if (!I8.Bit.07 && !I07)
  {O1 = (I8.Bit.07 && I07) << 8;
  O2 = (I8.Bit.07 && I07) << 8;}
O3 = 0;
O4 = 0;
O5 = 0;
O6 = 0;
O7 = 0;
O8 = 0;
O9 = 0;
O10 = 0;
O11 = 0;
O12 = 0;
O13 = 0;
O14 = 0;
O15 = 0;
O16 = 0;
```

Parameter

StoIo $PlcMan.OutRPar
StoIo $PlcMan.OutIPar
StoIo $PlcMan.OutVValuePar
StoDp $PlcMan.ReadValuePar
StoDp $PlcMan.ReadCodePar
StoDp $PlcMan.ReadCodeSubPar

Request and transmitting data to the client
Request and transmitting data to the client
Write value and transmit data from client to master
Read value and transmit data to client
Code 1-8 : for all

NB!
The attribute below must always placed last in this PlcPgm.

RestOp $PlcMan.OpReset
RestOp $PlcMan.OpInitPar
RestOp $PlcMan.OpResPar
RestOp $PlcMan.OpBackupPar

Class-BaseFcPPO5Fo-Code			
W			
04-APR-2008 16:09	Proview	JE	5