

```
1  PROGRAM PLC_PRG
2  VAR
3      // Resets position command to 0
4      rPosCmd : LREAL := 0 ;
5      // Selection of different ball position modes from HMI
6      iModeSelection : INT ;
7      // Define MB_Axis_Administration
8      Axis_Admin : MB_AXIS_ADMINISTRATION ;
9  END_VAR
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
```

---

```
1  //
2
3  //Setting of variables and calculations that can have a slow cycle time
4
5  //Calculations for the right input from the HMI
6  // Calc ms from nanosec for the D_part gain in the HMI
7  gvl.lDCtrlPID := gvl.lDGain * 1000000 ;
8  // Calc ms from nanosec for the I_part gain in the HMI
9  gvl.lICtrlPID := GVL.lIGain * 1000000 ;
10
11 // Calc ms from nanosec for the D_part gain in the HMI
12 gvl.lDCtrlPIDVel := gvl.lDGainVel * 1000000 ;
13 // Calc ms from nanosec for the I_part gain in the HMI
14 gvl.lICtrlPIDVel := GVL.lIGainVel * 1000000 ;
15
16 // Setting Remote to off on startup.
17 bRemoteOn_gb := FALSE ;
18
19 //-----
20 // Logic for switching between slider, sinus and square position command
21 // Making variables for buttons in HMI
22
23 // Reset all boolean values TO FALSE when gvl.bEnable is FALSE
24 IF NOT gvl.bEnable THEN
25     gvl.bSliderOn := FALSE ;
26     gvl.bSinusOn := FALSE ;
27     gvl.bLeftPos := FALSE ;
28     gvl.bRightPos := FALSE ;
29     gvl.bCenterPos := FALSE ;
30     gvl.bStepOn := FALSE ;
31     iModeSelection := 4 ; // Set to center position mode
32 ELSE
33
34 // Mode selection based on the state of the boolean inputs
35 IF gvl.bSliderOn THEN
36     iModeSelection := 0 ; // Slider mode
37     gvl.bSinusOn := FALSE ; // Reset the other boolean inputs to false
38     gvl.bLeftPos := FALSE ;
39     gvl.bRightPos := FALSE ;
40     gvl.bCenterPos := FALSE ;
```

```
40     gvl.bStepOn := FALSE ;
41   ELSIF gvl.bSinusOn THEN
42     iModeSelection := 1 ; // Sinusoidal mode
43     gvl.bSliderOn := FALSE ; // Reset the other boolean inputs to false
44     gvl.bLeftPos := FALSE ;
45     gvl.bRightPos := FALSE ;
46     gvl.bCenterPos := FALSE ;
47     gvl.bStepOn := FALSE ;
48   ELSIF gvl.bStepOn THEN
49     iModeSelection := gvl.iSignalType ;
50     gvl.bSliderOn := FALSE ; // Reset the other boolean inputs to false
51     gvl.bSinusOn := FALSE ;
52     gvl.bCenterPos := FALSE ;
53   ELSIF gvl.bCenterPos THEN
54     iModeSelection := 4 ; // Center position mode
55     gvl.bSliderOn := FALSE ; // Reset the other boolean inputs to false
56     gvl.bSinusOn := FALSE ;
57     gvl.bLeftPos := FALSE ;
58     gvl.bRightPos := FALSE ;
59     gvl.bStepOn := False ;
60   ELSE
61     iModeSelection := 4 ; // If all boolean inputs are false, set to center
62     position mode
63     END_IF
64   END_IF
65
66   //-----
67   // Mode selection in HMI, choose position command
68   CASE iModeSelection OF
69     0 : // Direct value from slider
70       rPosCmd := gvl.rSliderVal ;
71     1 : // Sinusoidal input start
72       // Sinus output to ball position control PID;
73       rPosCmd := gvl.rSinusOutputPID ;
74     2 : // First set value
75       rPosCmd := gvl.rSetLeftPosition ;
76     3 : // Second set value
77       rPosCmd := gvl.rSetRightPosition ;
78     4 : // Home position of beam
79       rPosCmd := gvl.rBallHomePos ;
80   END_CASE
81
82   // Assign the final position command value to the gvl.rBallPosCmd variable
83   gvl.rBallPosCmd := rPosCmd ;
84
85   //-----
```

```
88      // Beam calibration. Set absolute position of servo motor encoder
89      IF gvl.bSetRefPos THEN
90          gvl.bEnable := FALSE;
91          gvl.bBeamZero := FALSE;
92          gvl.rJogStep := 0;
93          arAxisCtrl_gb[1].Admin.SetAbsRef := gvl.bSetRefPos;
94      ELSE
95          arAxisCtrl_gb[1].Admin.SetAbsRef := FALSE;
96      END_IF
97
98
99
100
101
102
103
104
105
106
107
108
109
110
```