POU: PLC_PRG

```
PROGRAM  PLC_PRG
VAR
    // Resets postition command to 0
    rPosCmd :  LREAL  :=  0 ;
    // Selection of different ball position modes from HMI
    iModeSelection :  INT ;
    // Define MB_Axis_Administration
    Axis_Admin :  MB_AXIS_ADMINISTRATION ;
END_VAR

```

```
//

//Setting of variables and calculations that can have a slow cycle time

//Calculations for the right input from the HMI
// Calc ms from nanosec for the D_part gain in the HMI
gvl . lDCtrlPID  :=  gvl . lDGain  *  1000000 ;
// Calc ms from nanosec for the I_part gain in the HMI
gvl . lICtrlPID  :=  GVL . lIGain  *  1000000 ;

// Calc ms from nanosec for the D_part gain in the HMI
gvl . lDCtrlPIDVel  :=  gvl . lDGainVel  *  1000000 ;
// Calc ms from nanosec for the I_part gain in the HMI
gvl . lICtrlPIDVel  :=  GVL . lIGainVel  *  1000000 ;

// Setting Remote to off on startup.
bRemoteOn_gb  :=  FALSE ;

//-----------------------------------------------------------------------------------
// Logic for switching between slider, sinus and square position command
// Making variables for buttons in HMI

// Reset all boolean values TO FALSE when gvl.bEnable is FALSE
IF  NOT  gvl . bEnable  THEN
    gvl . bSliderOn  :=  FALSE ;
    gvl . bSinusOn  :=  FALSE ;
    gvl . bLeftPos  :=  FALSE ;
    gvl . bRightPos  :=  FALSE ;
    gvl . bCenterPos  :=  FALSE ;
    gvl . bStepOn  :=  FALSE ;
    iModeSelection  :=  4 ;  // Set to center position mode
ELSE

// Mode selection based on the state of the boolean inputs
IF  gvl . bSliderOn  THEN
    iModeSelection  :=  0 ;  // Slider mode
    gvl . bSinusOn  :=  FALSE ;  // Reset the other boolean inputs to false
    gvl . bLeftPos  :=  FALSE ;
    gvl . bRightPos  :=  FALSE ;
    gvl . bCenterPos  :=  FALSE ;
    gvl . bStepOn  :=  FALSE ;
ELSIF  gvl . bSinusOn  THEN
    iModeSelection  :=  1 ;  // Sinusoidal mode
    gvl . bSliderOn  :=  FALSE ;  // Reset the other boolean inputs to false
    gvl . bLeftPos  :=  FALSE ;
    gvl . bRightPos  :=  FALSE ;
    gvl . bCenterPos  :=  FALSE ;
    gvl . bStepOn  :=  FALSE ;
ELSIF  gvl . bStepOn  THEN
    iModeSelection  :=  gvl . iSignalType ;
    gvl . bSliderOn  :=  FALSE ;  // Reset the other boolean inputs to false
    gvl . bSinusOn  :=  FALSE ;
    gvl . bCenterPos  :=  FALSE ;
ELSIF  gvl . bCenterPos  THEN
    iModeSelection  :=  4 ;  // Center position mode
    gvl . bSliderOn  :=  FALSE ;  // Reset the other boolean inputs to false
    gvl . bSinusOn  :=  FALSE ;
    gvl . bLeftPos  :=  FALSE ;
    gvl . bRightPos  :=  FALSE ;
    gvl . bStepOn  :=  False ;
ELSE
    iModeSelection  :=  4 ;  // If all boolean inputs are false, set to center position mode
    END_IF
END_IF

//------------------------------------------------------------------------------
// Mode selection in HMI, choose position command
CASE  iModeSelection  OF
    0 :  // Direct value from slider
        rPosCmd  :=  gvl . rSliderVal ;
    1 :  // Sinusoidal input start

        // Sinus output to ball position control PID;
        rPosCmd  :=  gvl . rSinusOutputPID ;

    2 :  // First set value
        rPosCmd  :=  gvl . rSetLeftPosition ;

    3 :  // Second set value
        rPosCmd  :=  gvl . rSetRightPosition ;

    4 :  // Home position of beam
        rPosCmd  :=  gvl . rBallHomePos ;
END_CASE

// Assign the final position command value to the gvl.rBallPosCmd variable
gvl . rBallPosCmd  :=  rPosCmd ;


//------------------------------------------------------------------------------
```

```
// Beam calibration. Set absolute position of servo motor encoder
IF  gvl . bSetRefPos  THEN
    gvl . bEnable  :=  FALSE ;
    gvl . bBeamZero  :=  FALSE ;
    gvl . rJogStep  :=  0 ;
    arAxisCtrl_gb [ 1 ] . Admin . SetAbsRef := gvl . bSetRefPos ;
    ELSE
        arAxisCtrl_gb [ 1 ] . Admin . SetAbsRef :=  FALSE ;
END_IF
```