



Write a 61131-3 structured text program to implement an interlock to provide overfill of a vessel using a level sensor and and inlet valve.



In this example, we will create a structured text program using the IEC 61131-3 standard to implement an interlock system for a vessel. This system will prevent overfilling by using a level sensor and an inlet valve. The level sensor will provide feedback on the vessel's current fill level, while the inlet valve will control the flow of liquid into the vessel.

We will use the following variables:

- LS\_High: Level sensor high-level signal (BOOL)
- LS\_Low: Level sensor low-level signal (BOOL)
- Valve\_Open: Inlet valve open command (BOOL)
- Valve\_Close: Inlet valve close command (BOOL)
- Valve\_Status: Inlet valve current status (BOOL)

Here's the structured text program:

iecst

Copy code

```
PROGRAM OverfillProtection
VAR
    LS_High: BOOL := FALSE;
    LS_Low:  BOOL := FALSE;
    Valve_Open: BOOL := FALSE;
    Valve_Close: BOOL := FALSE;
    Valve_Status: BOOL := FALSE;
END_VAR

METHOD RunInterlock
    IF LS_High THEN
        Valve_Close := TRUE;
        Valve_Open := FALSE;
    ELSIF LS_Low THEN
        Valve_Close := FALSE;
        Valve_Open := TRUE;
    END_IF;

    IF Valve_Open AND NOT Valve_Close THEN
        Valve_Status := TRUE;    // Open the valve
    ELSIF Valve_Close AND NOT Valve_Open THEN
        Valve_Status := FALSE;  // Close the valve
    END_IF;
END_METHOD
END_PROGRAM
```

To use this program, you'll need to integrate it into your PLC system and configure the inputs (LS\_High and LS\_Low) to be connected to the appropriate level sensor signals, and the outputs (Valve\_Open and Valve\_Close) to control the inlet valve. You'll also need to call the RunInterlock method periodically (e.g., in a cyclic task) to ensure the interlock is continuously monitoring and controlling the vessel fill level.