



Write a self-contained 61131-3 function block for weight control of a tank that can be filled using vales. The weight of the tank is determined using a scale. The function block monitors the tank weight to determine if the tank is full, empty, or in between the two states. The block provides a single command that can have four values: 1. Fill the tank, 2. Stop filling the tank, 3. Start the stirrer, 4. Empty the tank. As necessary, the appropriate values are opened or closed to control the tank level. The stirrer can only work if the tank is full; otherwise, the command is ignored.



Here's an example implementation of a function block for weight control of a tank that can be filled using valves:



css Copy code

```
FUNCTION_BLOCK TankWeightControl
VAR_INPUT
    TankScaleWeight: REAL;      (* current weight of the tank on the scale *)
    TankCapacity: REAL;         (* maximum capacity of the tank *)
    FillValve: BOOL;            (* valve to control tank filling *)
    DrainValve: BOOL;           (* valve to control tank emptying *)
    Stirrer: BOOL;              (* stirrer control *)
END_VAR
VAR_OUTPUT
    TankState: INT;             (* current state of the tank *)
    ControlCommand: INT;        (* command to control tank level *)
    FillInProgress: BOOL;       (* indicates whether the tank is currently being
END_VAR

(* Define the state values for the tank *)
CONST
    EMPTY = 0;
    PARTIAL = 1;
    FULL = 2;
END_CONST

(* Define the command values *)
CONST
    FILL = 1;
    STOP_FILLING = 2;
    START_STIRRER = 3;
    EMPTY_TANK = 4;
END_CONST

VAR
    TargetWeight: REAL;         (* weight at which the fill valve should be closed *)
    FillStarted: BOOL;          (* indicates whether the tank has started filling *)
END_VAR

(* Determine the current state of the tank based on its weight *)
IF TankScaleWeight <= 0 THEN
    TankState := EMPTY;
ELSIF TankScaleWeight >= TankCapacity THEN
    TankState := FULL;
ELSE
    TankState := PARTIAL;
END_IF

(* Set the control command based on the tank state and input commands *)
CASE TankState OF
    EMPTY:
        FillInProgress := FALSE;
        ControlCommand := FILL;
        TargetWeight := TankCapacity * 0.2; (* fill the tank to 20% capacity *)
    PARTIAL:
        IF FillValve AND NOT FillInProgress THEN
            FillInProgress := TRUE;
            FillStarted := TRUE;
        END_IF;
        IF FillInProgress AND TankScaleWeight >= TargetWeight THEN
            FillInProgress := FALSE;
            ControlCommand := STOP_FILLING;
            TargetWeight := 0;
            IF Stirrer THEN
                ControlCommand := START_STIRRER;
            END_IF;
        ELSE
            ControlCommand := FILL;
        END_IF;
    FULL:
        FillInProgress := FALSE;
        ControlCommand := EMPTY_TANK;
        IF Stirrer THEN
            ControlCommand := START_STIRRER;
        END_IF;
END_CASE

END_FUNCTION_BLOCK
```

This function block takes in the current weight of the tank, the maximum capacity of the tank, and several input signals to control the filling, emptying, and stirring of the tank. It outputs the current state of the tank, a control command, and a signal indicating whether the tank is currently being filled.

The function block works by first determining the current state of the tank based on its weight. If the tank is empty, it sets the control command to fill the tank. If the tank is partially filled, it checks whether the fill valve is open and starts filling the tank if it is not already in progress. It then monitors the weight of the tank to determine when it should stop filling. When the target weight is reached, it closes the fill valve, sets the control command to stop filling, and optionally starts the stirrer. If the tank is full, it sets the control command to empty the tank, optionally starting the stirrer as well.

Note that this is just one possible implementation of a function block for tank weight control, and the exact details may vary depending on the specific requirements of your application.