Warsaw University of Technology
Electrical Engineering Department

1DA1669:A
Programmable Logic Controllers

Laboratory Project Report: XYZ

Report Author:

No.	Surname	Name	Index Number	E-mail	Lesson Date and Time
1.	Ndlovu	Mike	329956	01184436@pw.edu.pl	Th. 12:15–14:00

Date: 21 May 2025
Supervisor: Dr inż. Piotr Pura

Contents

1. Project Description

The task was to implement a centrifuge control system using Siemens TIA Portal. The system manages two centrifuges (Big and Small) with the following functionalities:

- State-based control (Idle, Filling, Accelerating, Running, Decelerating, Emptying, Cycle Check)
- Speed control with configurable acceleration/deceleration
- Cycle management (up to 2 operational cycles)
- Timer-based operations for filling/emptying

2. Project Requirements Implementation

2.1. Control Algorithm

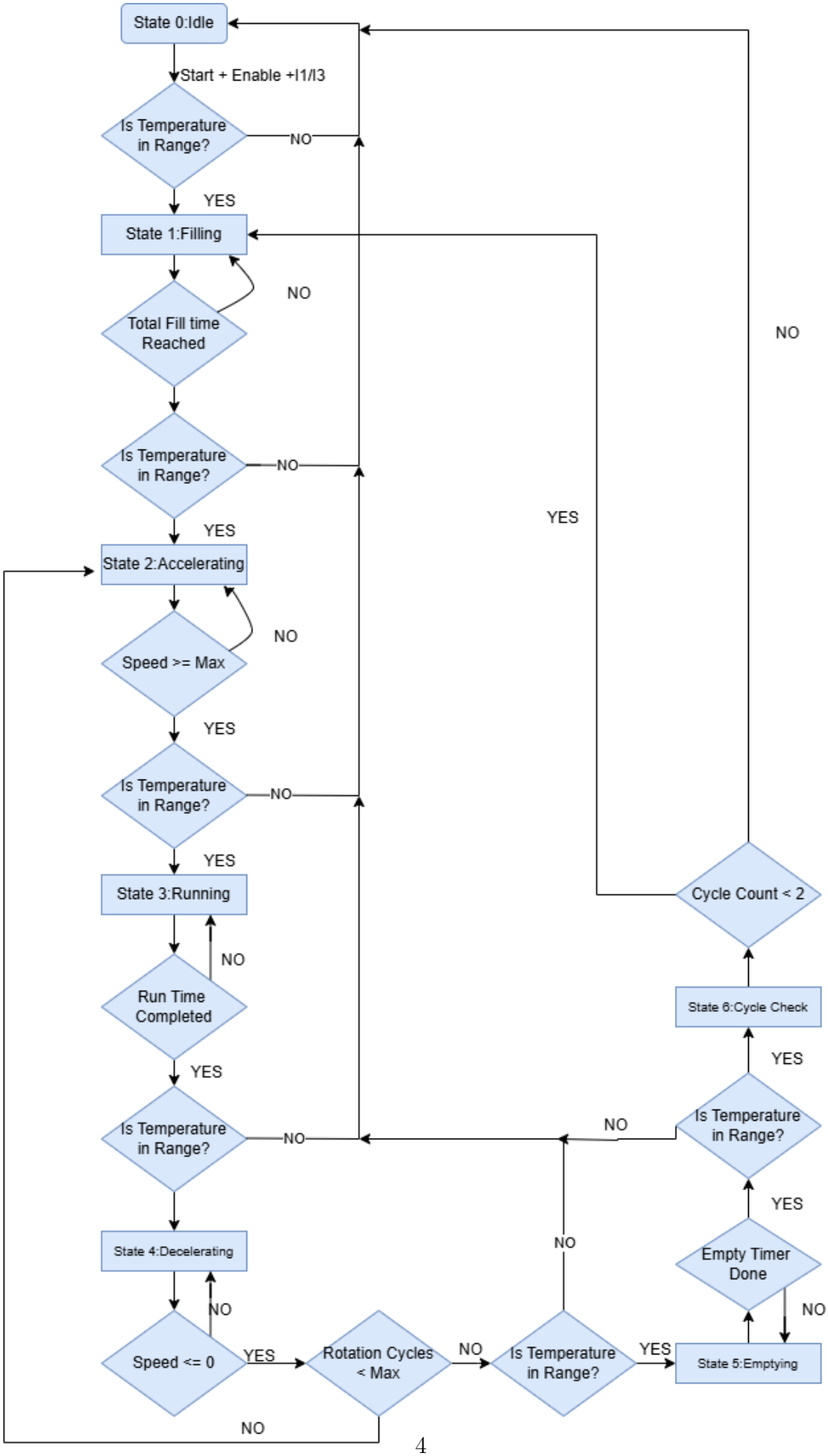

Figure 1: Centrifuge control system state flowchart

State 1 (Filling):

- Activates $FillTimer$
- Accumulates $TotalFillTime+ = FillTimer.ET$
- **Transition:** When $(TotalFillTime + FillTimer.ET) \geq FillTime$
- **Note:** Transition is blocked if temperature is out of range

State 2 (Accelerating):

- Calculates RPM using acceleration equation
- Clips speed at $MaxSpeed$
- **Transition:** $W \geq MaxSpeed$
- **Note:** Transition is blocked if temperature is out of range

State 3 (Running):

- Maintains constant $W = MaxSpeed$
- **Transition:** $RunTimer.ET \geq (RunTime - 2 \times DecelerationTime)$
- **Note:** Transition is blocked if temperature is out of range

State 4 (Decelerating):

- Reduces speed using deceleration equation
- **Transition:**
  - $W \leq 0$ & $rotation\_cycle < 3 \rightarrow$ State 2
  - $W \leq 0$ & $rotation\_cycle \geq 3 \rightarrow$ State 5
- **Note:** Transition is blocked if temperature is out of range

State 5 (Emptying):

- Activates $EmptyTimer$ ($PT = 5s$ Big, $3s$ Small)
- **Transition:** $EmptyTimer.Q = TRUE$
- **Note:** Transition is blocked if temperature is out of range

State 6 (Cycle Check):

- Manages operational cycles
- **Transition:**
  - $CycleCount < 2 \rightarrow$ State 1
  - $CycleCount \geq 2 \rightarrow$ State 0
- **Note:** Transition is blocked if temperature is out of range

2.1.1 Used variables & Inputs

longtable

Name	Type	Functionality	Range/Value
Total_Big_FillTime	TIME	Accumulated fill time for Big centrifuge	-
Big_A	REAL	Acceleration rate for Big centrifuge	5.0
Big_FillED	BOOL	Flag indicating Big centrifuge fill complete	TRUE/FALSE
Big_MaxSpeed	REAL	Maximum speed for Big centrifuge	30.0
Big_FillTime	TIME	Target fill duration for Big centrifuge	3s
Big_RunTime	TIME	Run duration at full speed for Big centrifuge	20s
Big_EmptyTime	TIME	Emptying time for Big centrifuge	5s
rotation_cycle_big	INT	Current rotation cycle count for Big centrifuge	0 to 3
rotation_max_cycles	INT	Maximum rotation cycles per run	3
Big_State	INT	Current state of Big centrifuge control	0 to 6
Big_W	REAL	Current speed of Big centrifuge	0.0 to 30.0
Big_CycleCount	INT	Completed operation cycles for Big centrifuge	0 to 2
deceleration_time_big	TIME	Deceleration time for Big centrifuge	Calculated dynamically
Total_Small_FillTime	TIME	Accumulated fill time for Small centrifuge	-
Small_A	REAL	Acceleration rate for Small centrifuge	10.0
Small_FillED	BOOL	Flag indicating Small centrifuge fill complete	TRUE/FALSE
Small_MaxSpeed	REAL	Maximum speed for Small centrifuge	50.0
Small_FillTime	TIME	Target fill duration for Small centrifuge	2s
Small_RunTime	TIME	Run duration at full speed for Small centrifuge	20s
Small_EmptyTime	TIME	Emptying time for Small centrifuge	3s
rotation_cycle_small	REAL	Current rotation cycle count for Small centrifuge	0 to 3
Small_State	INT	Current state of Small centrifuge control	0 to 6
Small_W	REAL	Current speed of Small centrifuge	0.0 to 50.0
Small_CycleCount	INT	Completed operation cycles for Small centrifuge	0 to 2
deceleration_time_small	TIME	Deceleration time for Small centrifuge	Calculated dynamically
Temp_Min	REAL	Minimum allowed temperature	20.0
Temp_Max	REAL	Maximum allowed temperature	50.0
Temperature	REAL	Current process temperature	0.0 to 100.0
Temp_Fault	BOOL	Flag indicating temperature out of range	TRUE/FALSE

Input Variables

Name	Type	Functionality	Range/Value
Start	BOOL	Start signal for centrifuge operation	TRUE/FALSE
Enable	BOOL	Enable signal to activate logic	TRUE/FALSE
I1	BOOL	Big centrifuge sensor input	TRUE/FALSE
I3	BOOL	Small centrifuge sensor input	TRUE/FALSE
Temp_Sensor	WORD	Raw temperature sensor value (0-65535)	Scaled to 0.0 to 100.0

Table 2: List of project input variables.

2.1.2 Speed Control Equations

The rotational speed W of the centrifuges is controlled using a linear acceleration and deceleration model, computed based on the elapsed time t from the timer.

Acceleration Equation During the acceleration phase, the speed increases linearly as:

$$W = A \times \frac{t}{1000} \tag{1}$$

where:

- W – Current rotational speed [RPM or rad/s]
- A – Acceleration rate [RPM/s or rad/s^2]
- t – Elapsed time from the acceleration timer [ms] [1] [2]

Clipping Condition Speed is limited to the configured maximum speed:

$$W \leq W_{max} \tag{2}$$

Run Time Compensation The run phase holds the speed for a duration reduced by twice the deceleration time:

$$RunTime_{adjusted} = RunTime - 2 \times DecelerationTime \tag{3}$$

Deceleration Equation During deceleration, the speed decreases linearly as:

$$W = W_{max} - A \times \frac{t}{1000} \tag{4}$$

Stop Condition The system stops when:

$$W \leq 0 \tag{5}$$

2.2. Project testing description

To run the centrifuge control project from start to finish, follow these steps:

1. Make sure the temperature is within the safe range defined by `Temp_Min` and `Temp_Max`.
2. If temperature is out of range, the `Temp_Fault` flag will block all operations until the temperature is corrected.
3. Press the **Enable** button to activate the control logic.
4. Press the **Start** button to begin the centrifuge sequence.
5. Provide sensor inputs:
   - **I1** – Start the Big centrifuge when the signal is TRUE.
   - **I3** – Start the Small centrifuge when the signal is TRUE.
6. Observe the process stages on the HMI:
   - Filling
   - Acceleration
   - Running at full speed
   - Deceleration
   - Emptying
   - Cycle repetition up to two times
7. If temperature exceeds the safety limits at any time, all centrifuge operations will immediately stop and timers will reset.
8. After two complete cycles, the centrifuges return to the idle state, ready for the next start.

List 2.1. Temperature Fault Detection Logic in ST language

```
"CentrifugeData".Temperature := DINT_TO_REAL(WORD_TO_DINT("Temp_Sensor"))
                                * (100.0 / 65535.0);

IF ("CentrifugeData".Temperature < "CentrifugeData".Temp_Min) OR
   ("CentrifugeData".Temperature > "CentrifugeData".Temp_Max) THEN
    "CentrifugeData".Temp_Fault := TRUE;
    // Reseting timers
    "CentrifugeData".Big_FillTimer(IN := FALSE, PT := T#50s);
    "CentrifugeData".Big_AccelTimer(IN := FALSE, PT := T#50s);
    "CentrifugeData".Big_RunTimer(IN := FALSE, PT := T#50s);
    "CentrifugeData".Big_EmptyTimer(IN := FALSE, PT := T#50s);

    "CentrifugeData".Small_FillTimer(IN := FALSE, PT := T#50s);
    "CentrifugeData".Small_AccelTimer(IN := FALSE, PT := T#50s);
```

```
    "CentrifugeData".Small_RunTimer(IN := FALSE, PT := T#50s);
    "CentrifugeData".Small_EmptyTimer(IN := FALSE, PT := T#50s);

    "CentrifugeData".Big_W := 0.0;
    "CentrifugeData".Small_W := 0.0;
    "CentrifugeData".Big_State := 0;
    "CentrifugeData".Small_State := 0;

    RETURN;  // Exiting all state logic
END_IF;
```

3. Visualization

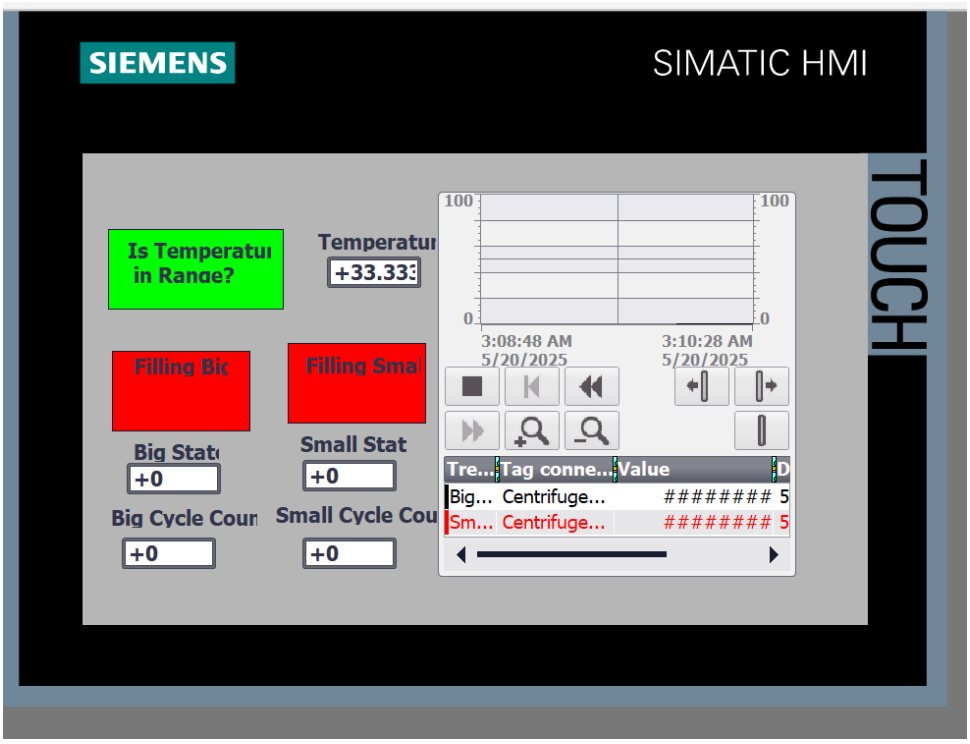

Figure 2: Initial Idle State - System reset and ready to start.

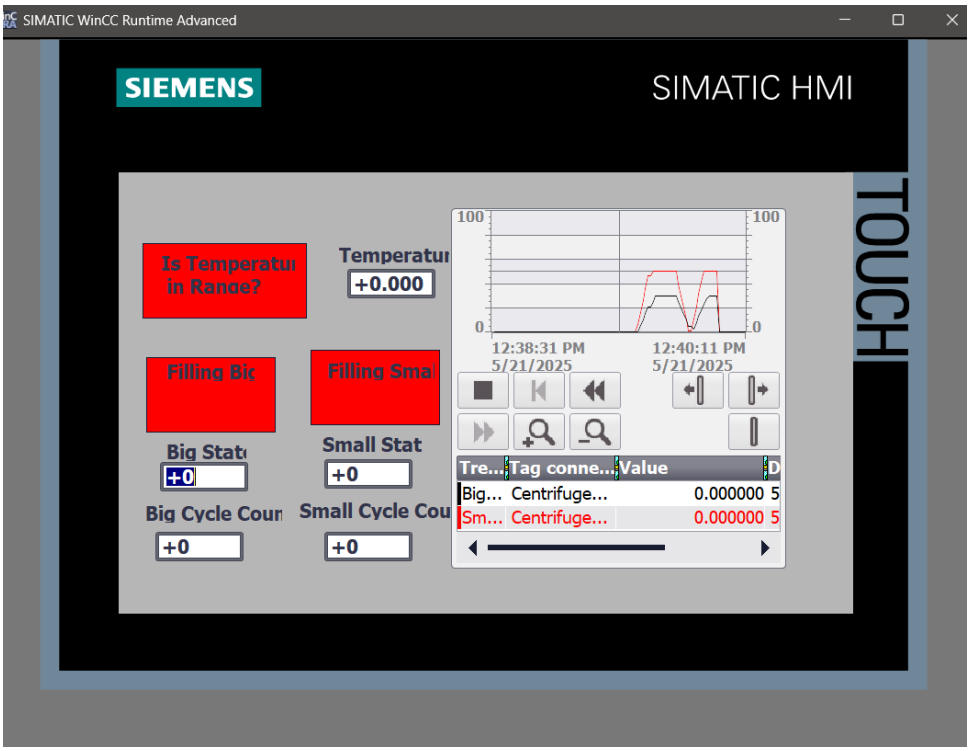

Figure 3: Temperature Below Minimum - Process blocked due to low temperature.

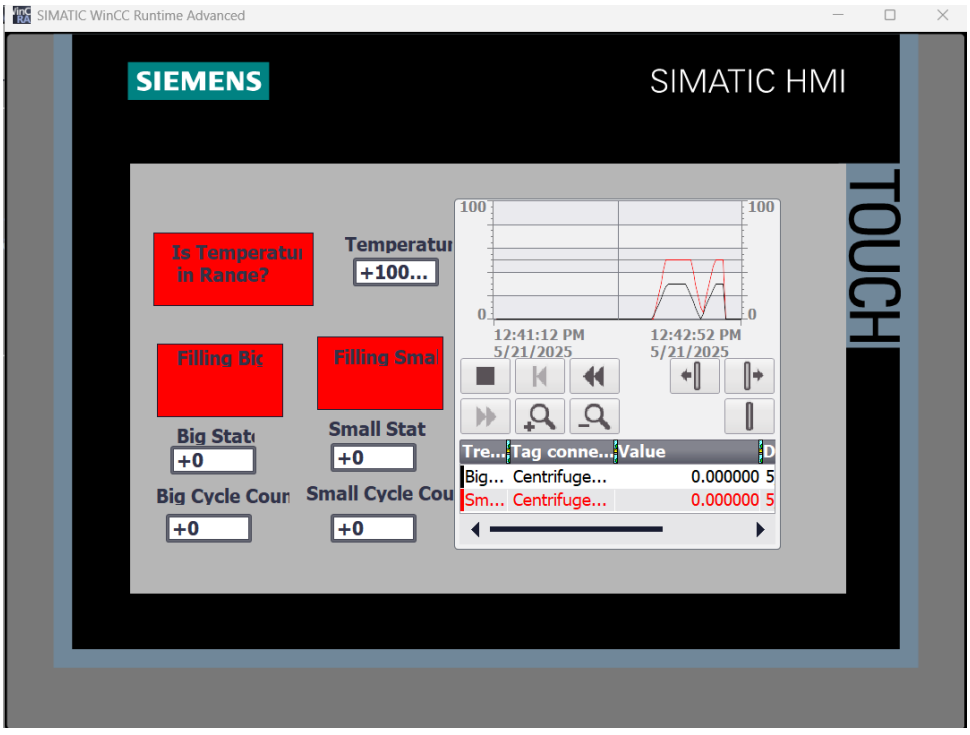

Figure 4: Temperature Above Maximum - Process blocked due to high temperature.

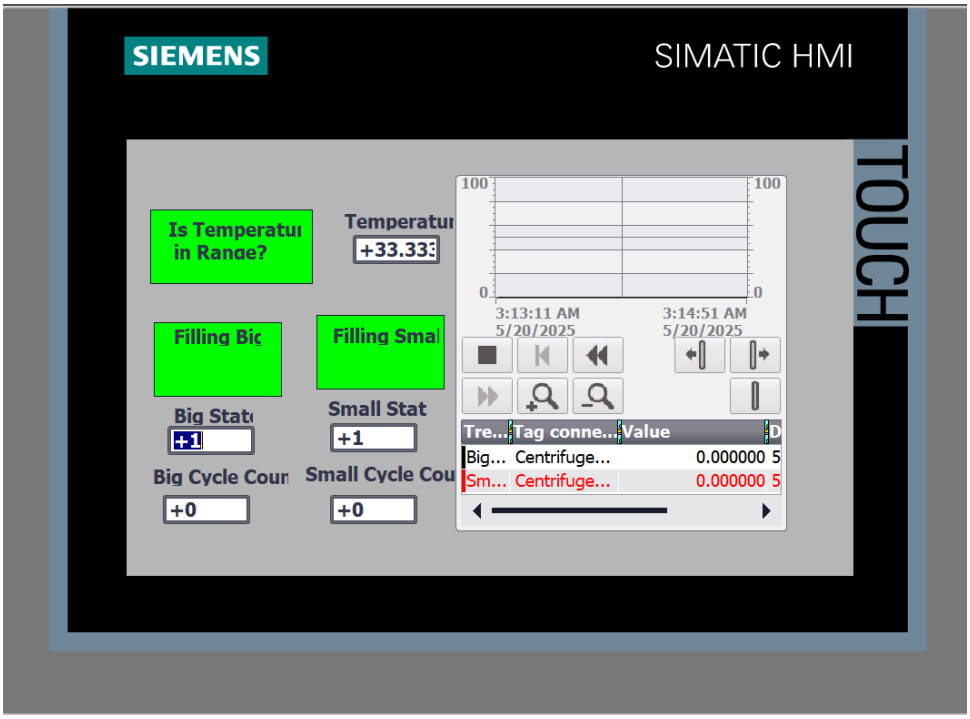

Figure 5: Filling On

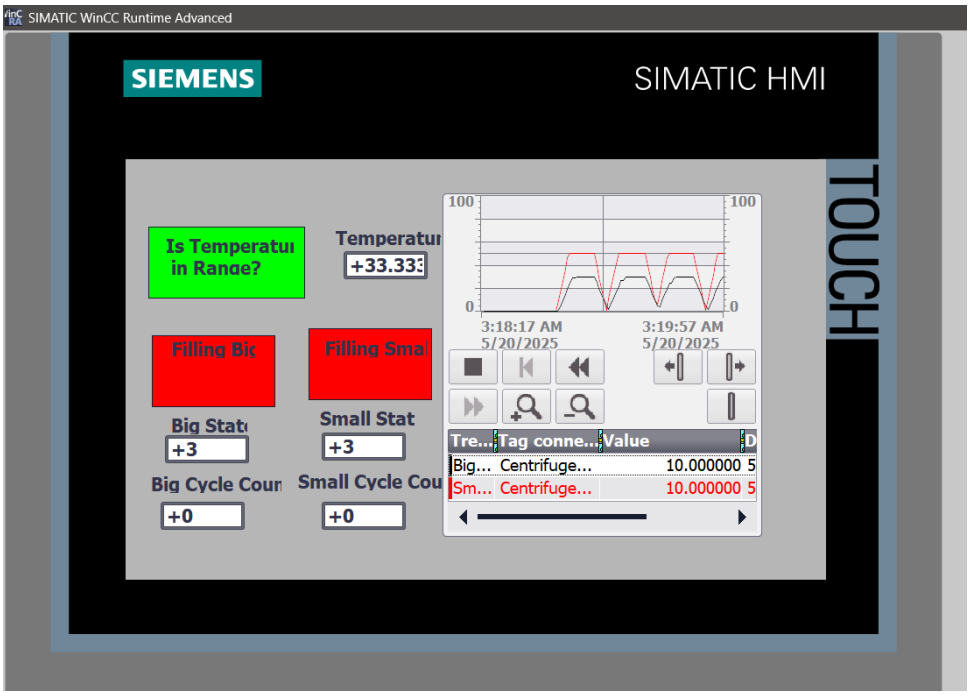

Figure 6: Running,accelerating and deceleration Repeated

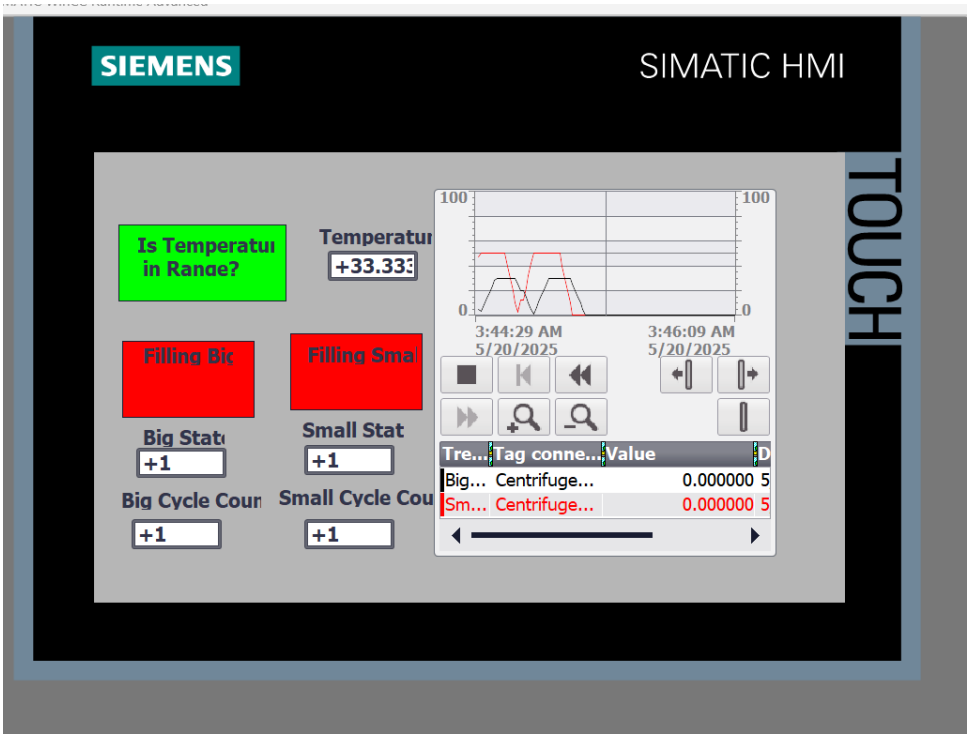

Figure 7: 1 cycle complete

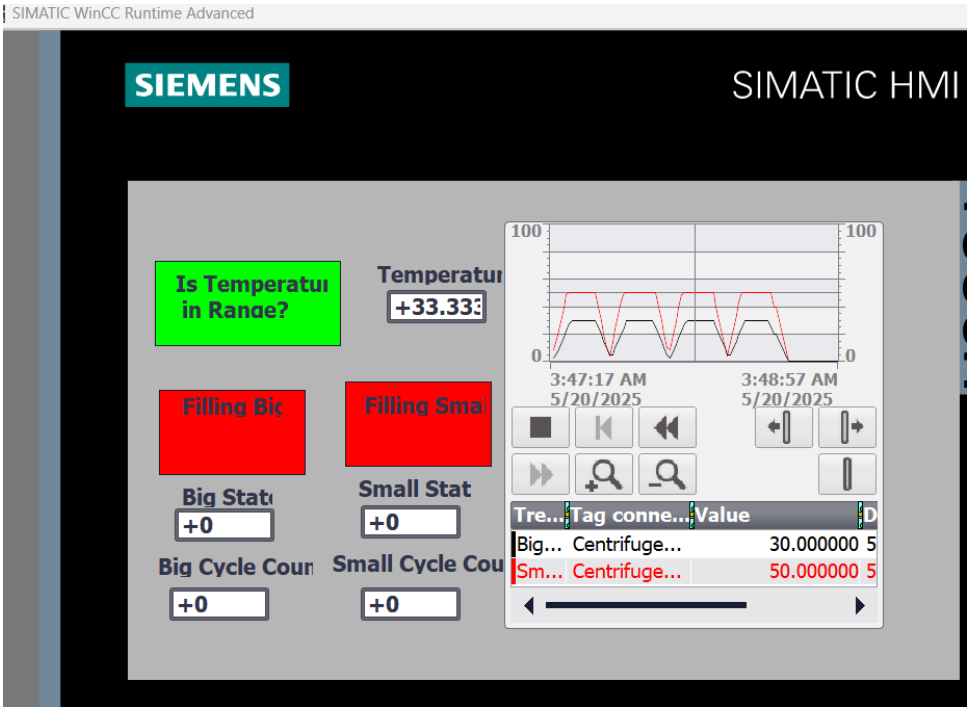

Figure 8: Second cycle complete the system resets cycle count

4. Summary

The report describes a laboratory project carried out in the PLC Control Systems course, in which a multi-phase control sequence for two sugar centrifuges (Big and Small) was implemented using a Siemens S7-1500 PLC and visualized on a TP700 Comfort HMI.

Fallowing imposed features were done:

- Fully automated control of two centrifuges, each following a 7-state cycle: idle, filling, acceleration, running, deceleration, emptying, and cycle check.
- Timed acceleration and deceleration profiles ensuring smooth ramping of motor speed.
- Independent cycle counters and state machines for Big and Small centrifuges.
- State-based control with transitions based on elapsed times and rotation cycle counts.
- Visualization of process variables and states on the HMI, including state numbers and cycle counters.

The project was extended with the following functionalities:

- **Partial Fill Monitoring via I1 and I3:** Sensor inputs I1 and I3 were added to allow the user to monitor filling conditions, enabling filling to stop at different material levels—not just when the timer completes. This offers flexibility for partial batch processing.
- **Temperature Safety Integration:** A temperature sensor input (`Temp_Sensor`) was added and scaled to a 0–100°C range. If the measured value falls outside of the safe bounds defined by `Temp_Min` and `Temp_Max`, the system:
  - Sets a `Temp_Fault` flag,
  - Immediately halts all centrifuge operation,
  - Resets timers and motor speed variables to prevent unsafe behavior.

References

[1] John R. Taylor. *Classical Mechanics*. English. Sausalito, Calif: University Science Books, Jan. 2005. ISBN: 978-1-891389-22-1.

[2] *TIA Portal Tutorial by Siemens*. en. URL: http://www.youtube.com/playlist?list=PLRtRKudOMmtESeAAeO6CeLYpYRjRz2jv4.