**NIRMA UNIVERSITY**

**INSTITUE OF TECHNOLOGY**

**MECHANICAL ENGINEERING**

**Programmable logic controller (PLC)**

**LAB – 9**

**19BME134**

**Shrey Shah**

**Aim**

Programming with Arithmetic, Comparison Instructions.

**Objectives**

To implement different processes with PLC programming

**Theory**

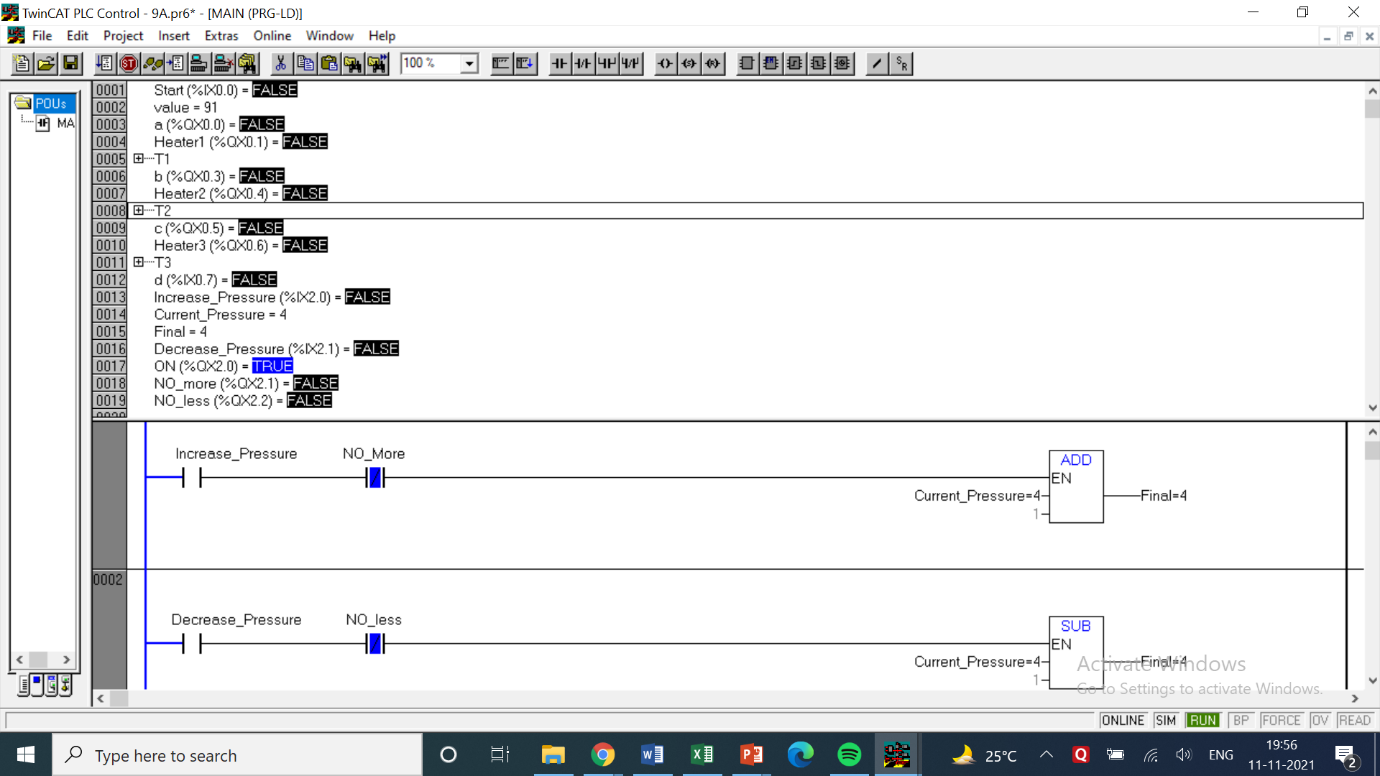
The action of the control response can be summarized as follows:

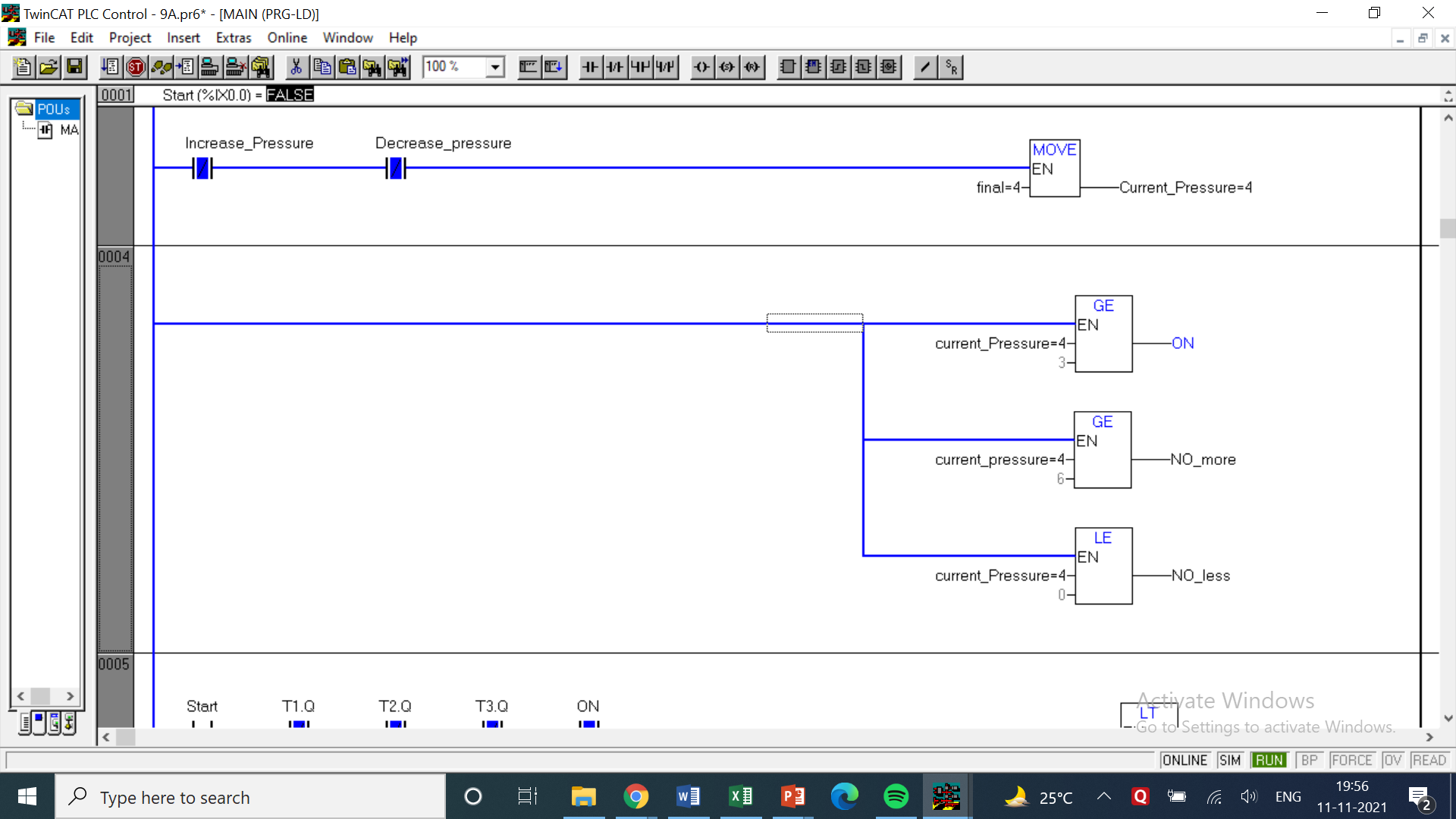
* Proportional-integral-derivative (PID) control is the most sophisticated and widely used type of process control.
* PID operations are more complex and are mathematically based. PID controllers produce outputs that depend on the magnitude, duration, and rate of change of the system error signal.
* Sudden system disturbances are met with an aggressive attempt to correct the condition. A PID controller can reduce the system error to 0 faster than any other controller.
* Programmable controllers are either equipped with PID I/O modules that produce PID control or have sufficient mathematical functions of their own to allow PID control to be carried out.

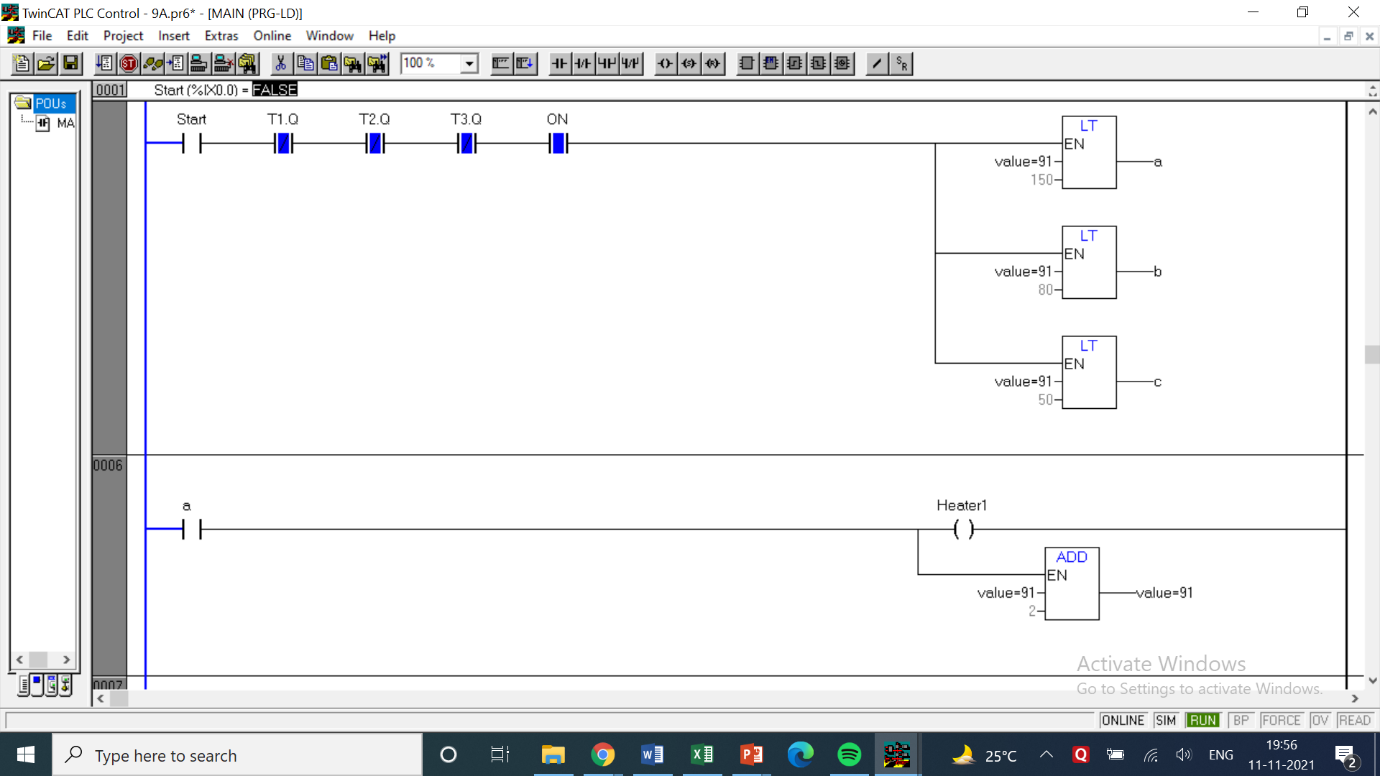
**Question 1**

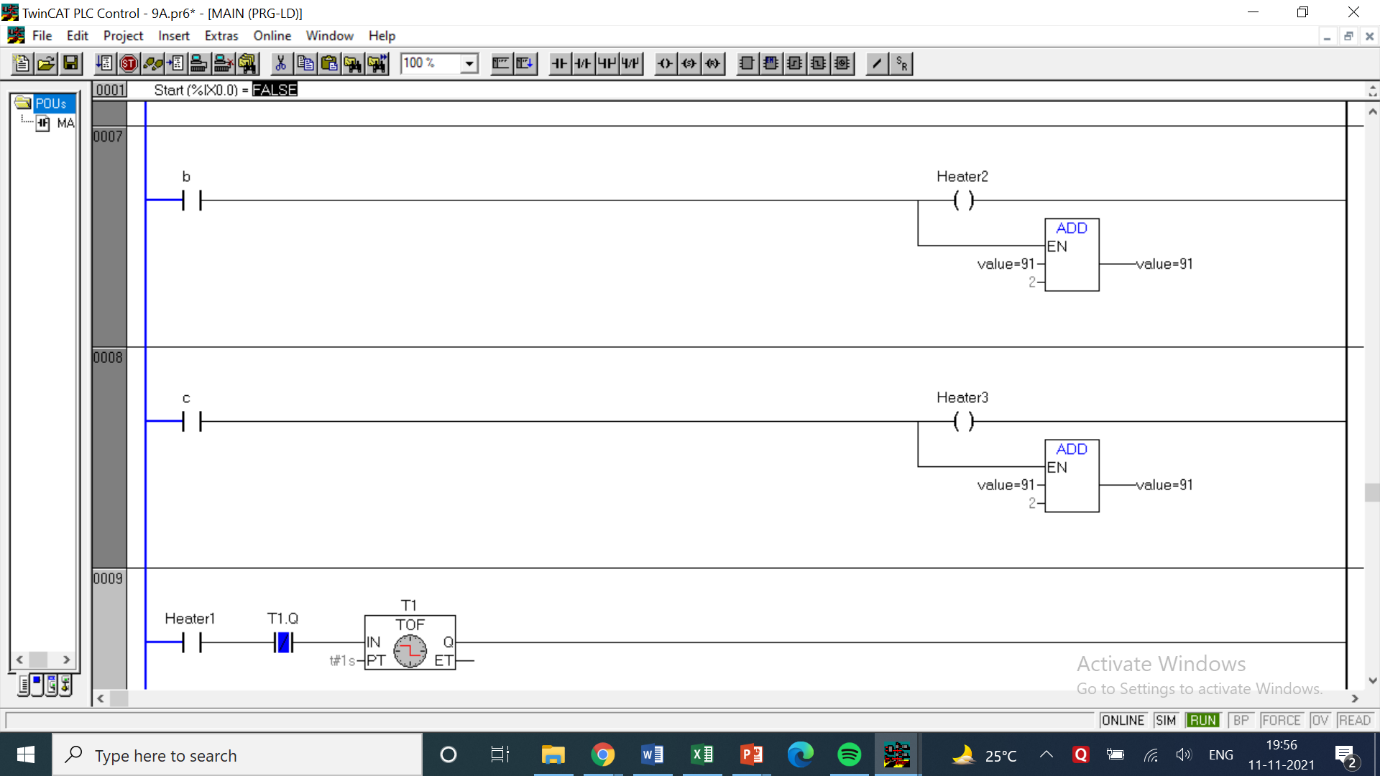
1. In a process, there are two sensors and 3 Outputs in which one is pressure sensor that is read value from 0 to 6 bar, and one temperature sensor use to control three heaters as per below condition.

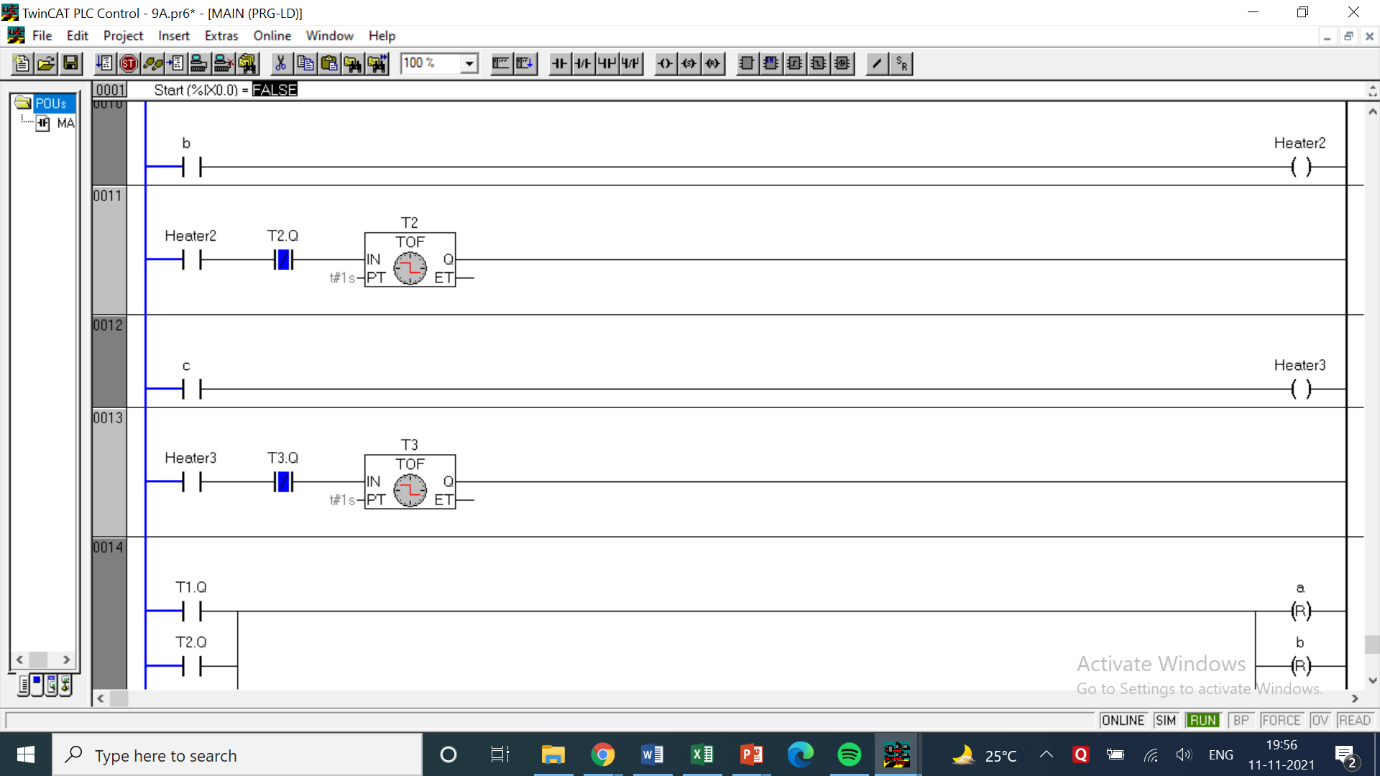
* Process is start only if pressure is grater than 3 bar.
* When temp is less than 50 o C all heaters will be ON.
* When 50 0 C < temp . 80 o C two heaeters will be ON.
* When 80 0 C < temp . 150 o C only one heater will be ON.
* When temp > 120 0 C All heaters should be OFF.









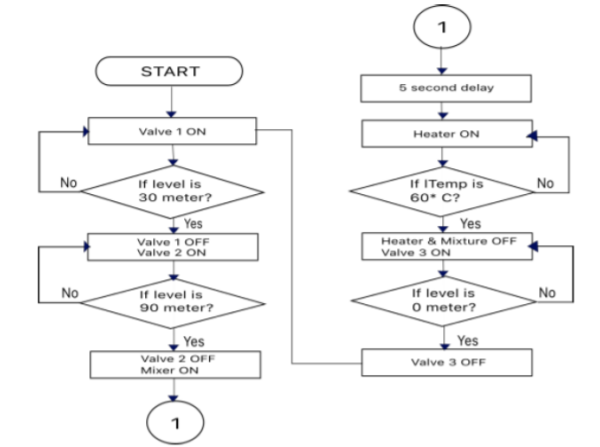


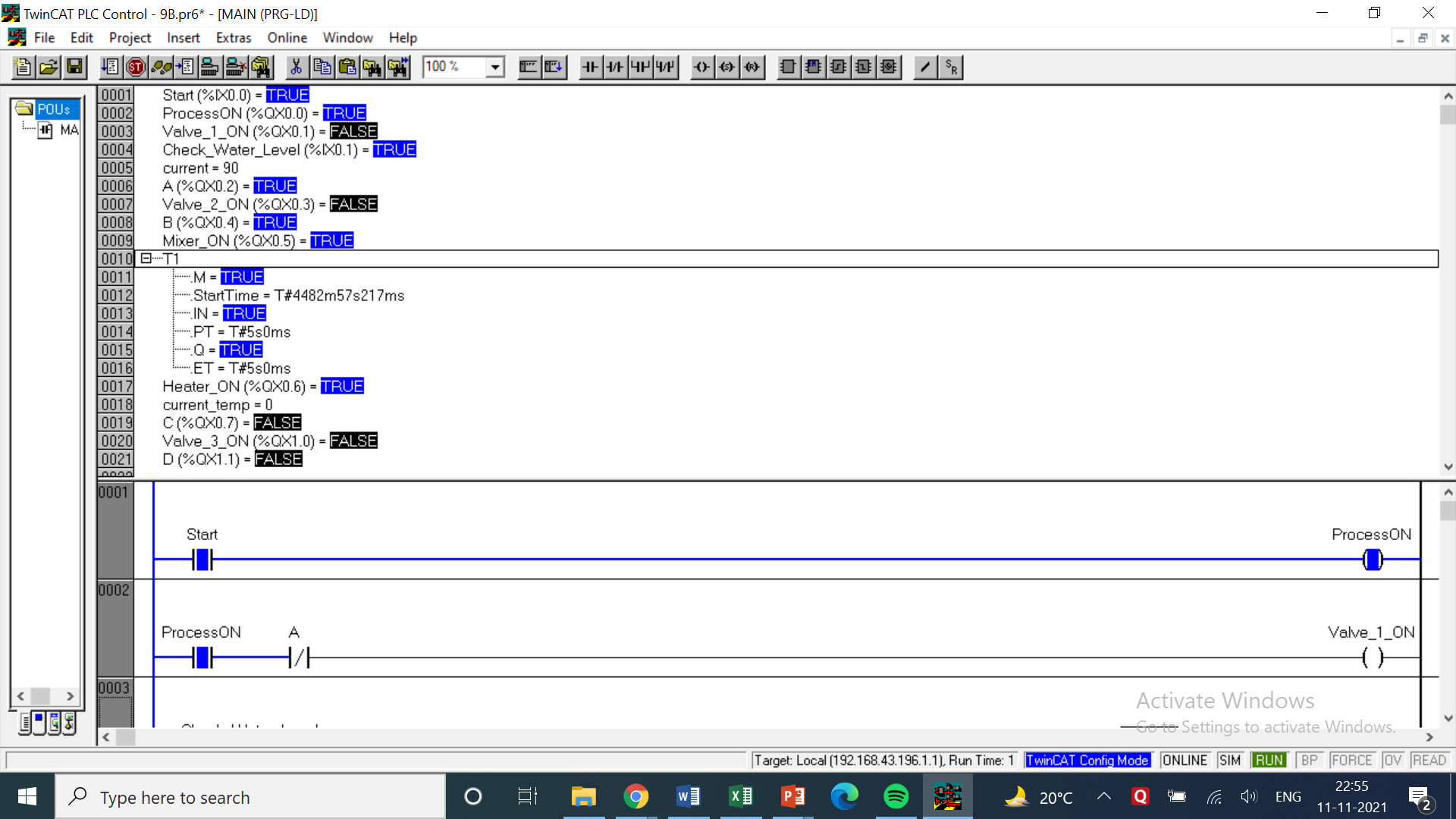
LOGIC -

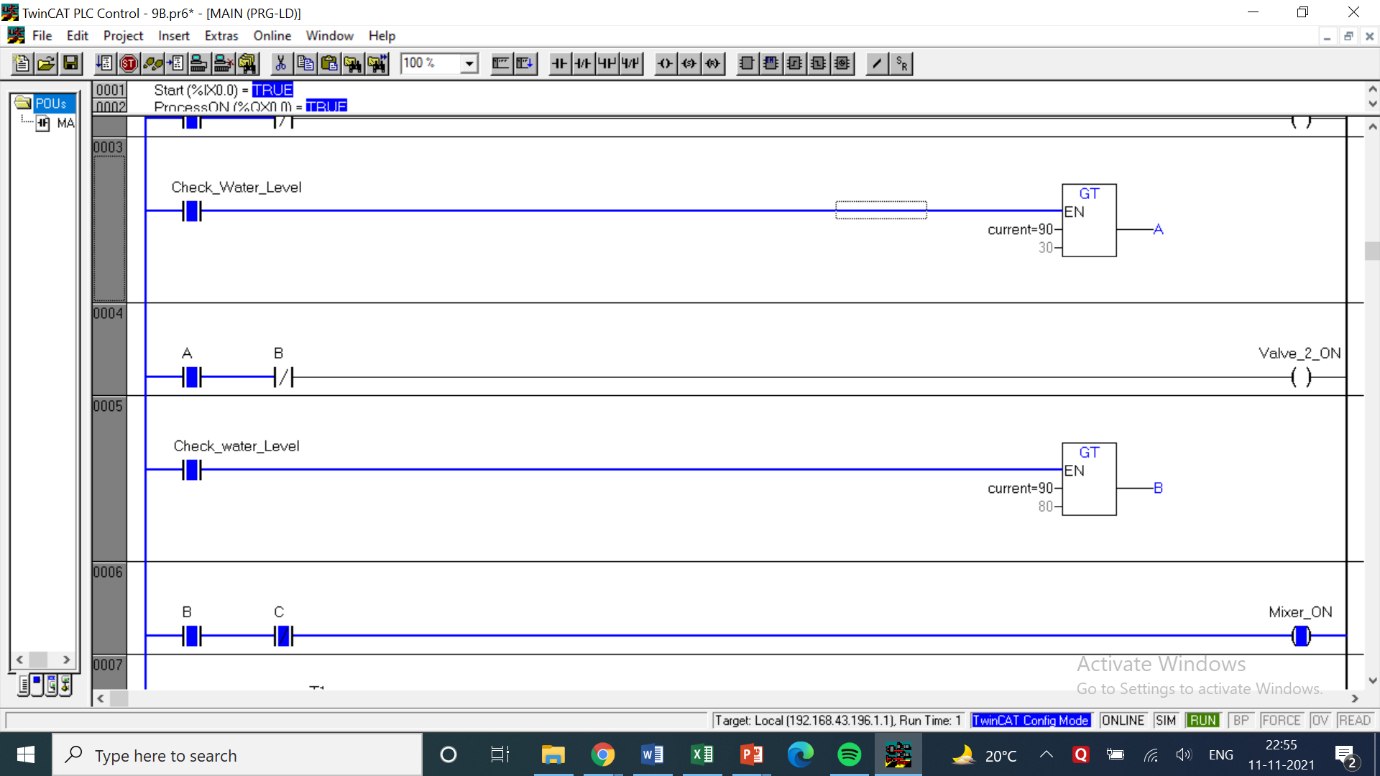
* In the first few rungs there is a small ladder logic for pressure control of the tank. Here buttons are used to increase and decrease 1 bar of pressure upon pressing the respective switch. The current pressure is always being checked by the function block and as soon as it reaches 3 bar the process can now be turned on.
* It is designed in such a way that if the pressure has reached 6 bar the button won’t be able to increase pressure further while allowing the decreasing the pressure.
* There are 3 heaters which control the heating of the fluid. Each heater has its own heating temperature, i.e, when the fluid reaches the set temperature of the respective heater the heater will turn off.
* In the following ladder logic there are 3 heaters which has the set point of 50 °C. The second heater has the set point of 80 °C. The last heater has the set point of 150 °C.
* Each heater increases the temperature of the fluid by 2 °C which means that below 50 °C, 6 °C are increased at once. Below 80 °C, 4 °C are increased at once. This helps in smoothening the curve and steady heating.
* After pressing Start button rung, it checks the temperature of the fluid and decides which heaters to be turned on. After the needed heaters are turned on. Each heater has corresponding timer attached to it which regularly updates the settings.
* In the rungs after the heater simple add temperature function block is used to add 1 °C to the current value by each heater.
* After reaching the final value all heaters are turned off.

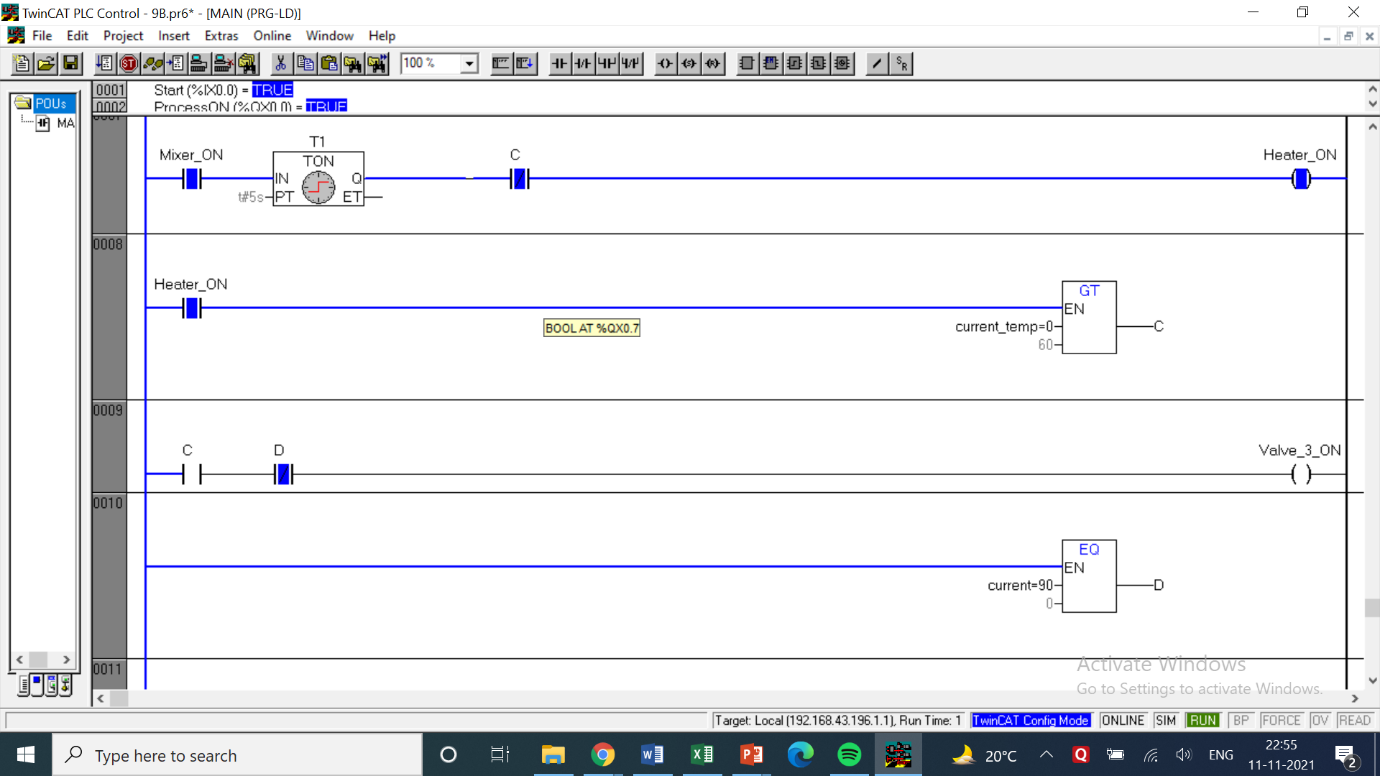
|  |  |  |  |
| --- | --- | --- | --- |
| INPUT | | OUTPUT | |
| Start | %ix0.0 | A | %qx0.0 |
| D | %ix0.7 | Heater 1 | %qx0.1 |
| Increase Pressure | %ix2.0 | B | %qx0.3 |
| Decrease Pressure | %ix2.1 | Heater 2 | %qx0.4 |
| Value (Int) | Initial = 0 | C | %qx0.5 |
| Current Pressure | Initial = 0 | Heater 3 | %qx0.6 |
| Final (Int) | Initial = NA | ON | %qx2.0 |
| Final (Int) | Initial Value = NA | No More | %qx2.1 |
| Timer T1 | Delay time = 1 sec | No Less | %qx2.2 |
| Timer T2 | Delay time = 1 sec | - | - |
| Timer T3 | Delay time = 1 sec | - | - |

1. Implement ladder logic for following process explained with flowchart.









LOGIC

* The logic for this question is VERY straight forward so much that I don’t know what to write ☹.
* So what I did was that I just added 1 rung for each of the blocks with arrows which means that every rung represents the steps in this function block with the last rung looping over the first valve using a Normally closed switch.
* This question can be somewhat inferred similar to the heater question above.

|  |  |  |  |
| --- | --- | --- | --- |
| INPUT | | OUTPUT | |
| Start | %ix0.0 | Process ON | %qx0.0 |
| Check Water Level | %ix0.1 | Valve 1 ON | %qx0.1 |
| Timer T1 | Delay Time = 5 sec | A | %qx0.2 |
| Current (Int) | Initial = 90 | Valve 2 ON | %qx0.3 |
| Current Temp | Initial = 0 | B | %qx0.4 |
| - | - | Mixer ON | %qx0.5 |
| - | - | Heater ON | %qx0.6 |
| - | - | C | %qx0.7 |
| - | - | Valve 3 ON | %qx1.0 |
| - | - | D | %qx1.1 |

COMMENTS –

* The counters in most of the questions are denoted by the letter ‘C’ followed by the number of the counter or the letter.
* In case of UP and DOWN timers their names have been specified while mentioning the use of counters. Cn.q represents the output of those counters which may be taken as NO or NC switches.
* Each question has a table of inputs and outputs which specifies which I/Os have been taken along with its addresses.
* All the timers are generally denoted by the symbol ‘tn’ where n represents the number of the timer.
* tn.q represents the output of the timer tn which can be both normally open or normally closed depending on how it is used based on the question’s requirements.

**Conclusion**

This experiment was a further application of various real time processes that can be performed using PLC programming. This lab particularly focused on comparing quantities and decide what actions are needed to be taken from the result of those comparisons and mathematical operations. It is a great resource in processes which involving regulating of a particular element.