University of Global Village (UGV), Barishal

Dept. of Electrical and Electronic Engineering (EEE)

Lab Manual

PLC & Microcontroller

Noor Md Shahriar
BSc in EEE, RUET
Senior Lecturer
Co-chairman, Dept. of EEE
University of Global Village (UGV)
874/322, C&B Road, Barishal, Bangladesh.
Contact: +8801743500587
Facebook | LinkedIn | Twitter

Reference Book:

1. "Programmable Logic Controllers: Principles and Applications" by John W. Webb and Ronald A. Reis.
2. PLC programming manuals from Siemens.

Table of Contents

Course Title:	Automation with PLC	Total Class Hour	37
Course Code:		Total Practice Hour	37
Supervised by	Noor Md Shahriar	Total Hour	85

Course Rationale

This course addresses the critical role of PLCs in industrial automation, equipping learners with the skills to design, program, and maintain efficient and reliable systems. With a focus on hands-on experience and real-world applications, it prepares participants to meet industry demands, optimize processes, and integrate advanced technologies like SCADA and IoT, driving innovation and operational excellence in automation.

Course Objectives

1. **Understand PLC Fundamentals**: Develop a solid foundation in the architecture, operation, and applications of Programmable Logic Controllers (PLCs).
2. **Develop Programming Proficiency**: Learn to program PLCs using Ladder Logic, Structured Text, and other methods to solve automation challenges.
3. **Integrate and Interface Systems**: Master the interfacing of PLCs with industrial devices, including sensors, actuators, HMIs, and SCADA systems.
4. **Apply Advanced Control Techniques**: Implement advanced functions like PID control, sequential control, and process optimization in real-time industrial systems.
5. **Design and Troubleshoot Automation Systems**: Gain hands-on experience in designing, testing, troubleshooting, and optimizing PLC-based industrial automation solutions.

Assessment Pattern

- Continuous Assessment

Bloom's Category	Tests
Imitation	12
Manipulation	8
Precision	6
Articulation	2
Naturalization	2

- Semester End Examination: (SEE):

Bloom's Category Marks (out of 30)	Tests (20)	Quiz (10)	External Participation in Curricular/Co-Curricular Activities (20)
Imitation	06	06	Bloom's Affective Domain: (Attitude or will) • **Attendance: 10** • **Viva-Voca: 5** • **Report Submission: 5**
Manipulation	04	04	
Precision	06		
Articulation	02		
Naturalization	02		

Course Learning Outcome (CLO)

CLO1	**Set up and configure PLC systems**, understanding hardware components and software setup.
CLO 2	**Develop and implement ladder logic programs** for basic PLC functions, such as logic gates, timers, and counters.
CLO 3	**Design and automate industrial systems** using PLC programming for applications like traffic lights and conveyor belts.
CLO 4	**Interface PLC with external systems**, including HMI and SCADA, for enhanced monitoring and control.
CLO 5	**Troubleshoot, debug, and optimize PLC programs**, including advanced control functions like PID control.

Course Outline

Sl. No.	Topic & Details	Class Hours	CLO Mapping
1	**Introduction to PLC Trainer and Software Setup:** Basics of PLC hardware, installation with PC, and interface overview.	2	CLO1
2	**Ladder Logic Programming:** Writing and implementing simple programs with digital inputs and outputs.	4	CLO1, CLO2
3	**Timers in PLC:** On-delay, off-delay, and retentive timers, their configuration, and implementation.	6	CLO2, CLO3
4	**Counters in PLC:** UP and DOWN counters, programming methods, and practical applications.	4	CLO2, CLO3
5	**Advanced Functions in PLC:** MOVE, control statements, math functions, and data manipulation techniques.	4	CLO3, CLO4
6	**Applications:** Conveyor system control, traffic light controller, and motor control using PLC.	6	CLO3, CLO4
7	**SCADA Integration with PLC:** Interface, tag association, and real-time operation using graphical symbols.	4	CLO4, CLO5

Course Schedule

Class No.	Topic	Teaching-Learning Strategy	Assessment Strategy	CLO Mapping
1	Introduction to PLC Trainer	Lecture and hands-on demonstration of PLC hardware and software setup.	Observation during setup tasks and Q&A on PLC basics.	CLO1
2	Writing and Implementing Ladder Logic: Digital I/O	Practical programming and simulation exercises.	Submission of ladder logic program, peer review, and instructor feedback.	CLO1, CLO2
3–4	Timer Functions in PLC	Lecture and hands-on practice of on-delay, off-delay, and retentive timers.	Class test on timer logic applications and real-time program evaluation.	CLO2, CLO3
5–6	Counter Functions in PLC	Practical exercises using up and down counters.	Submission of implemented counter programs, observation during lab sessions.	CLO2, CLO3
7	MOVE and Control Statements in PLC	Lecture with examples of math functions and data manipulation.	Class test on MOVE and control logic implementation.	CLO3, CLO4
8–9	Conveyor Control System Using PLC	Project-based learning with simulation and hardware interfacing.	Evaluation of functional conveyor system logic and report submission.	CLO3, CLO4

10–11	Traffic Light Controller Using PLC	Practical project work and debugging.	Evaluation of ladder program performance and in-class presentation.	CLO3, CLO4
12–13	SCADA Interface with PLC	Lecture on SCADA concepts, hands-on practice of tag association.	Real-time operation using SCADA interface, Q&A on interfacing challenges.	CLO4, CLO5
14–15	Capstone Project: Real-World Application of PLC	Group work, discussions, and troubleshooting.	Group project evaluation, peer review, and instructor feedback on project outcomes.	CLO4, CLO5
16	Review and Final Assessment	Revision of all topics with Q&A.	Final project presentation, assessment of individual and group contributions.	CLO1–CLO5

LIST OF EXPERIMENTS

Exp No.	NAME OF EXPERIMENTS
1	Introduction to PLC trainer & its installation with PC
2	Write and implement a simple ladder logic program using digital inputs and outputs for PLC.
3	Write and implementation a simple ladder logic program using timer. 1) On delay timer 2) Off delay timer 3) Retentive timer
4	Write and implementation a simple ladder logic program using counter. 1) UP counter 2) Down counter
5	Write program on MOVE, control statement, math function, data manipulation technique on PLC.
6	To study about conveyor control system using PLC
7	Write and implement ladder logic program to on-off the DC motor using PLC.
8	To study the traffic light controller system by using PLC.
9	Interface SCADA with PLC and associate tags with memory and I/ O and operate the PLC inputs through the switch symbol from the computer screen and view the status of the outputs using lamp and motor graphics symbols in the screen.

Lab Precaution

1. **Safety Awareness**: Always follow safety guidelines and avoid physical contact with live electrical components or circuits.
2. **Proper Handling of Equipment**: Handle the PLC hardware, cables, and tools with care to avoid damage.
3. **Power Management**: Ensure all power supplies are turned off before connecting or disconnecting any equipment.
4. **Correct Wiring**: Double-check the wiring connections as per the circuit diagram before powering the PLC to prevent short circuits.
5. **Software Use**: Use authorized software versions and verify the program logic before execution to avoid undesired operations.
6. **Environment**: Keep the workspace clean and free from liquids or debris that could cause hazards.
7. **Component Inspection**: Regularly inspect the components for any damage or wear before use.
8. **Emergency Protocols**: Familiarize yourself with the location of emergency power cut-off switches and safety equipment.
9. **Avoid Overloading**: Do not exceed the input/output ratings of the PLC or other connected devices.
10. **Personal Protection**: Wear appropriate safety gear such as insulated gloves if working with higher voltages.
11. **Supervised Use**: Always perform experiments under the supervision of a qualified instructor or technician.
12. **Documentation**: Maintain a detailed record of configurations, programs, and results for troubleshooting purposes.

Experiment no: 01

Programming And Logic Control (PLC)

AIM OF THE EXPERIMENT:

Introduction /Familiarization PLC trainer & its instruction with PC.

APPARATUS REQUIRED:

Sl. no.	Name of Equipment
1	Lab Programmable Logic Controller (PLC) kit
2	Personal Computer

THEORY:

What is PLC?

A programmable logic controller (PLC) is an industrial computer control system that continuously monitors the state of input device and makes decisions based upon a custom program to control the state of output devices.

What is inside a PLC?

- The central processing unit, the CPU, contains an internal program that tells the PLC how to perform the following functions.
- The CPU combines a microprocessor, an integrated power supply, input and output circuits, built in PROFINET, high speed motion control I/O, and on-board analog input in a compact housing to create a powerful controller.
- After you download your program, the CPU contains the logic required to monitor and control the devices in your application.
- **The CPU monitors the input and changes the outputs according to the logic of your user program**, which can include Boolean logic, counting, timing, complex math operations and communication with other intelligent devices.
- To communicate with a programming device, the CPU provides a built in PROFINET port.
- With the PROFINET network, the CPU can communicate with HMI panels or another CPU.
- To provide security for your application, every S7-1200 CPU provides password protection that allows you to configure access to the CPU functions.

- The CPU supports only a preformatted SIMATIC memory card. To insert a memory card, open the top CPU door and insert the memory card in the slot.
- Use the optional SIMATIC memory card either as a program card or as a transfer card.
- Digital inputs in the controller are 14 and voltage is 24. Digital outputs in the controller are 10 and voltage is 24.
- Back panel is available to connect extra I/O modules and communication modules.
- Analog input in the controller is two.
- Input power supply to the controller is 120/240 VAC.
- Three communication modules and eight I/O expansion modules can be used.
- Modbus communication board, output board and analog input board can be used.

***PLC Languages*:**

The function of all programming languages is to allow the user to communicate with the programmable controller via a programming device. They all convey to the system, by means of instructions, a basic control plan.

- The most common types of languages encountered in programmable controller system design is ;
    a) Ladder Diagram (LD)
    b) Function Block Diagram (FBD)

LADDER DIAGRAM (LD)

Traditional ladder logic is graphical programming language. Initially programmed with simple contacts that simulated the opening and closing relays, counters, timers, shift registers etc.

FUNCTION BLOCK DIAGRAM (FBD)

Useful for expressing the interconnection of control system algorithms and logic.

HARDWARE/SOFTWARE REQUIREMENT:

- Processor type - Intel Pentium i3, 2.5 GHZ or similar
- RAM - 4GB
- *Available hard disk space* - 10 GB on system drive C:\
- *Operating systems*- windows XP professional SP3, windows 2003 server R2 SP2, windows 7 (professional, enterprise, ultimate) SP1, windows 10 pro

- *Graphics card* - 32MB RAM 24-bit color depth
- *Screen resolution* - 1024 x 768
- *Network* – 20 M bit/s Ethernet or faster
- *Optical drive*- DVD-ROM

PROCEDURE: (PLC SETUP)

1. First select **TIA portal 14.0** and double click on it.
2. Double click on "**Create new project**" then select the "**Project name**" and select the location path to save project and then click "**Create**".
3. Double click on "**Configure a device**".
4. Click "**Add new device**". After device is added click "**Controller**" and then click on "**SIMATIC S71200**".
5. Click on "**CPU**" and "**CPU 1214 DC/DC/DC**", then select the required "**MLFB**" number and click "**Add**".
6. Now select the signal module **AQ1x12 BIT** and Add to the CPU and save the project by pressing **CTRL+S** key.
7. The select respective expansion module 3 no's of **DI 16/DQ 16x24VDC**, 1 no. of **DI 8/DQ8 x 24 VDC** and 1 no. of **AI 4 x 13 BIT/AQ 2 x 14 BIT** from the hardware catalogue.
8. Double click on the "**RJ 45**" symbol in the controller, now "**Properties**" of the controller will be open and Ethernet configuration also open, if it is not opened just click on the Ethernet address and change IP address and change IP address as required then save the project.
9. Then go to "**Downloading the program**" setting for click on the "**Compile**" icon and then click on the "**Download**" icon.
10. Then automatically the "**Extended to download device**" window open. Select the "**PN/IE**" into the type of the "**PG/PC**" interface and also select the "**Show all compatible devices**". Finally click the "**Start search**" option and then click the "**Load**" button.
11. In software synchronization before loading to a device window in opens and then clicks the "**Continue without synchronization**" option below.

12. "**Stop modules**", "**Stop all**" option and then "**Load**" option and then load option and also goes to "**Load results**" window tick the "Start all" and click the "**Finish**" option.

PROGRAMMING METHOD IN PLC:

1. Click "**PLC-1**" in the project tree then click "**Program block**" and click "**main OB1**". Now OB1 is created (object block1).
2. Now select the network and double click on "**normally open**" icon (⊣⊢).
3. Now normally open is added. Now add the "**output coil**" (– ( ) –) in the network. Then double click on the both NO contact and output coil to enter their addresses simultaneously.
4. After it click on the "**Compile**" icon then "**Download to device**" icon.
5. For make an online process goes to click the "**Monitoring on/off**" icon. Then the window shows the online mode of the PLC.

RESULT:

We studied introduction of PLC, its installation with PC, hardware components, building various blocks and determine no. of digital inputs/outputs & analog inputs/outputs.

Experiment No: 02

Write and implement a simple ladder logic program to study and verify and gate using digital inputs and outputs for PLC.

AND-GATE OPERATION

AIM OF THE EXPERIMENT:

Test the truth table of AND gate using PLC software.

APPARATUS REQUIRED:

Sl. no.	Apparatus Name
1.	PLC trainer kit
2.	Personal computer installed with PLC software
3.	Ethernet cable
4.	Patch chords

THEORY:

1) In AND_GATE operation is used to make the multiple operations of 2 inputs. Now using A&B are 2 inputs and C is the output.
2) Now generating the following formula to create the AND gate operation is given below.

Programming:-

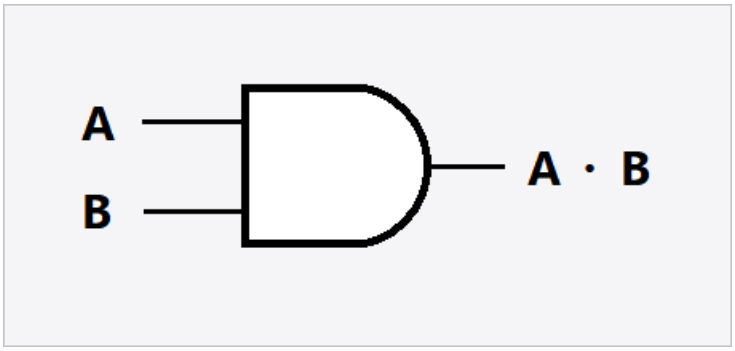
$$A.B = C$$

Truth table & Symbol of AND gate:

A	B	C

0	0	0
0	1	0
1	0	0
1	1	1

0-LOW ; 1-HIGH



PROCEDURE:

1. Click "**PLC-1**" in the project tree then click "**Program block**" and click "**Main OB1**". Now **OB1** is created (object block1).
2. Now write an **AND_GATE** operation for make a 2 inputs like A and B. It is created by using two normally open contacts. And their addresses are **A(I0.0)** and **B(I0.1).**
3. And the only one output can be used in this program that is C and their address is **(Q0.0).**
4. After saved the program go to online mode to click **GO Online** icon directly.

5. After click the online mode goes to select the **Monitoring ON/OFF** icon.

6. Finally the two inputs [(10.0),(10.1)]are goes to HIGH the output[Q0.0]will goes to HIGH.

CONCLUSION:

Thus the AND**_GATE** operation was studied successfully through **PLC Software.**

OR-GATE OPERATION

AIM OF THE EXPERIMENT:

Test the truth table of OR gate using PLC software.

APPARATUS REQUIRED:

Sl. no.	Apparatus Name
1.	PLC trainer kit
2.	Personal computer installed with PLC software
3.	Ethernet cable
4.	Patch chords

THEORY:

- In OR_GATE operation is used to make the addition operation of 2 inputs. Now using A&B are 2 inputs and C is the output.
- Now generating the following formula to create the AND_GATE operation is given below.

$$A + B = C$$

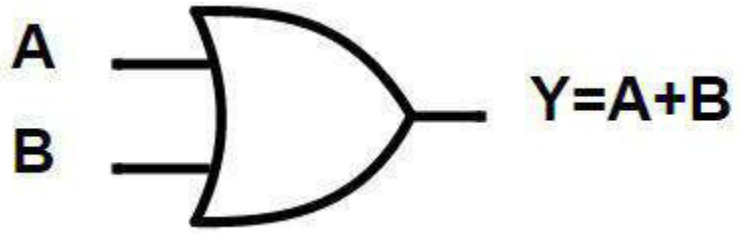
Programming:-

TRUTH TABLE & SYMBOL OF OR_GATE:

A	B	C
0	0	0
0	1	1

1	0	1
1	1	1

0-LOW ; 1-HIGH



PROCEDURE:

1. Click PLC-1 in the project tree then click **program block** and then click **main OB1.**
2. Now the (object block 1) **OB1** is created.
3. Now write an **OR_GATE** operation for make a 2 inputs like A and B.
4. It is created by using two normally open contacts parallel connection. And their addresses are A (**I0.0**) and B (**I0.1**).
5. And the only one output can be used in this program that is C and their address is (**Q0.0**).
6. Then save the program to press (CTRL+S) function it will be saved.
7. After saved the program go to online mode to click **GO Online** icon directly.
8. After click the online mode goes to select the **Monitoring ON/OFF** icon.
9. Finally the inputs [(I0.0), (I0.1)] are goes to **HIGH** the output [Q0.0] will goes to **HIGH**. It can be represented in **Green** color indication.

CONCLUSION:

Thus the **OR_GATE** operation was studied successfully through **PLC Software.**

EX_OR-GATE

AIM OF THE EXPERIMENT:

Test the truth table of EX- OR gate using PLC software.

APPARATUS REQUIRED:

Sl. no.	Apparatus Name
1.	PLC trainer kit
2.	Personal computer installed with PLC software
3.	Ethernet cable
4.	Patch chords

THEORY:

- In **EX-OR_GATE** operation is used to make the operation of inequality functions of inputs. Now using A,B and C is the output.
- Now generating the following formula to create the **EX-OR_GATE** operation is given below.

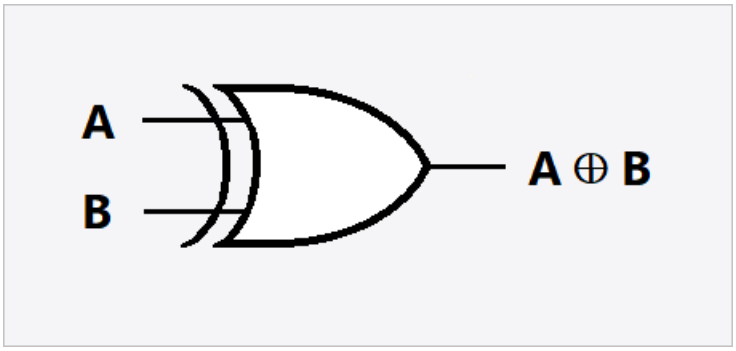
$$\overline{A}\,B + A.\overline{B} = C$$

PROGRAMMING:-

TRUTH TABLE & SYMBOL OF EX-OR_GATE:

A	B	C
0	0	0
0	1	1
1	0	1
1	1	0

0-LOW ; 1-HIGH



PROCEDURE:

1. Click **PLC-1** in the project tree then click **program block** and then click **main OB1.**
2. Now the (object block 1) **OB1** is created.
3. Now write an **EX-OR_GATE** operation for make a 2 inputs like A and B.
4. Then save the program and go to **download** into PLC.
5. After saved the program go to online mode to click **GO Online** icon directly.
6. Finally **monitoring** all the values of the inputs and outputs.
7. All the inputs are **LOW** at that time the output is **LOW**. The inequality functions of Inputs are **HIGH** the Output is **HIGH**.
8. Thus the all input values are **HIGH** at that time the Output is **HIGH**.

CONCLUSION:

Thus the **EX-0R_GATE** operation was studied successfully through **PLC Software.**

.

EX NOR-GATE

AIM OF EXPERIMENT:

Test the truth table of EX- OR gate using PLC software.

APPARATUS REQUIRED:

Sl. no.	Apparatus Name
1.	PLC trainer kit
2.	Personal computer installed with PLC software
3.	Ethernet cable
4.	Patch chords

THEORY:

- In **EXNOR_GATE** operation is used to make the inverse operation of the **EXNOR_GATE** values. Now using A,B and C is the output.
- Now generating the following formula to create the **EXNOR_GATE** operation is given below.

$$\overline{A}.\overline{B} + A.B = C$$

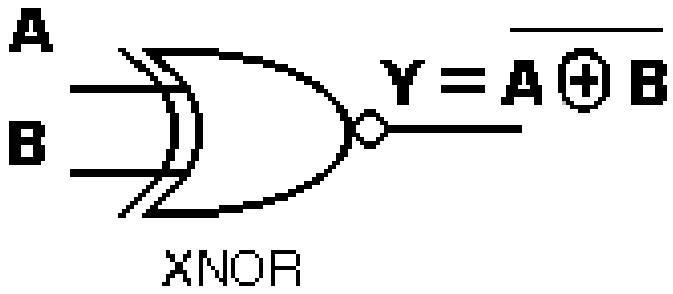
Programming:-

TRUTH TABLE & SYMBOL OF EXNOR GATE:

A	B	C
0	0	1
0	1	0

1	0	0
1	1	1

0-LOW 1-HIGH



PROCEDURE:

1. Click **PLC-1** in the project tree then click **program block** and then click **main OB1.**
2. Now the (object block 1) **OB1** is created.
3. Now write an **EXNOR_GATE** operation for make a 2 inputs like A and B.
4. Then save the program and go to **download** into PLC.
5. After saved the program go to online mode to click **GO Online** icon directly.
6. Finally **monitoring** all the values of the inputs and outputs.
7. All the inputs are **LOW** at that time the output is **HIGH**. The inequality functions of Inputs are **HIGH** the Output is **HIGH**.
8. Thus the all input values are **HIGH** at that time the Output is **HIGH**.

CONCLUSION:

Thus the **EX-N0R_GATE** operation was studied successfully through **PLC Software.**

Experiment No: 03

Write and implementation of simple ladder logic program using timer

ON DELAY TIMER

AIM OF THE EXPERIMENT:

To study the ON Delay Timer operation using PLC Software.

APPARATUS REQUIRED:

- PLC Trainer kit
- Personal Computer Installed with PLC Software
- Ethernet cable
- Patch chords

THEORY:

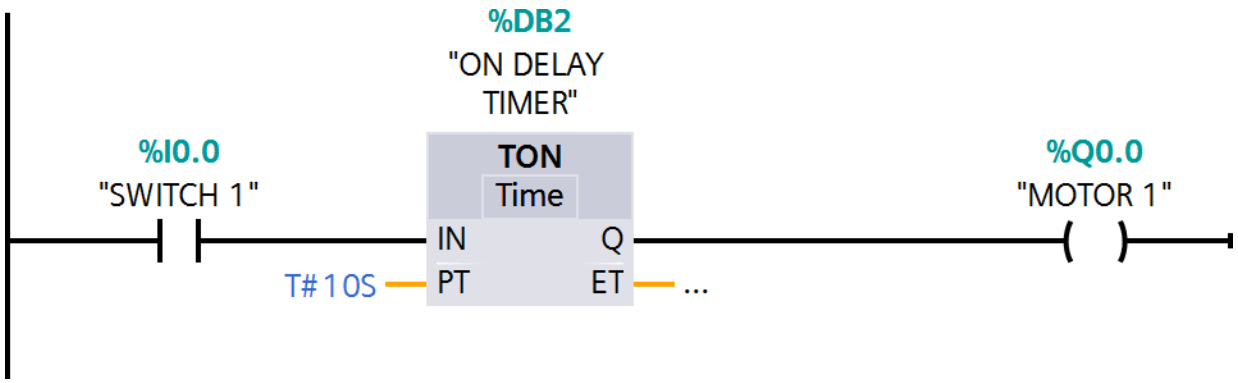
- ON Delay Timer is used to make the Delay operation in processes.
- It can control the inputs and outputs of the field instruments.

PROCEDURE:

1. Click PLC-1 in the project tree "1" then click program block "2" and click main OB1 "3" as shown in the below image.
2. Now the (Object Block1) OB1 is created.
3. Now select the On Delay Timer goes to instructions tab and click the Basic Instructions and select the Timer operations tab and then double click the TON block.
4. Now the call options block is displayed. Then change the data block name. And click the automatic option finally select the ok button.
5. Now the Timer block is created in network is shown below the diagram. And also that one input and one output coil is created in same network for testing the timer block.
6. The input can be connected with IN of the Timer block. Then output coil connected with Q of the Timer block.
7. Then given the addresses for input and outputs. And finally given the preset time for the Timer goes to double click the PT in timer block and write the values in sec.
8. Then save the program and download after that goes to online mode. And monitoring the program can be displayed in following images.
9. Then given the addresses for input and outputs. And finally given the preset time for the Timer goes to double click the PT in timer block and write the values in sec.
10. Then save the program and download after that goes to online mode and monitor the program.

11. When the status of the SWITCH 1(I0.0) changes from 0 to 1 the timer instruction will be executed and it will activate the MOTOR 1(Q0.0) after 10s delay.

LADDER LOGIC PROGRAM:-



CONCLUSION:

Thus, the ladder logic program of ON Timer was written and implemented successfully using PLC Software.

OFF DELAY TIMER

AIM OF THE EXPERIMENT:

To study the OFF-Delay Timer operation using PLC Software.

APPARATUS REQUIRED:

- PLC Trainer kit
- Personal Computer Installed with PLC Software
- Ethernet cable
- Patch chords

THEORY:

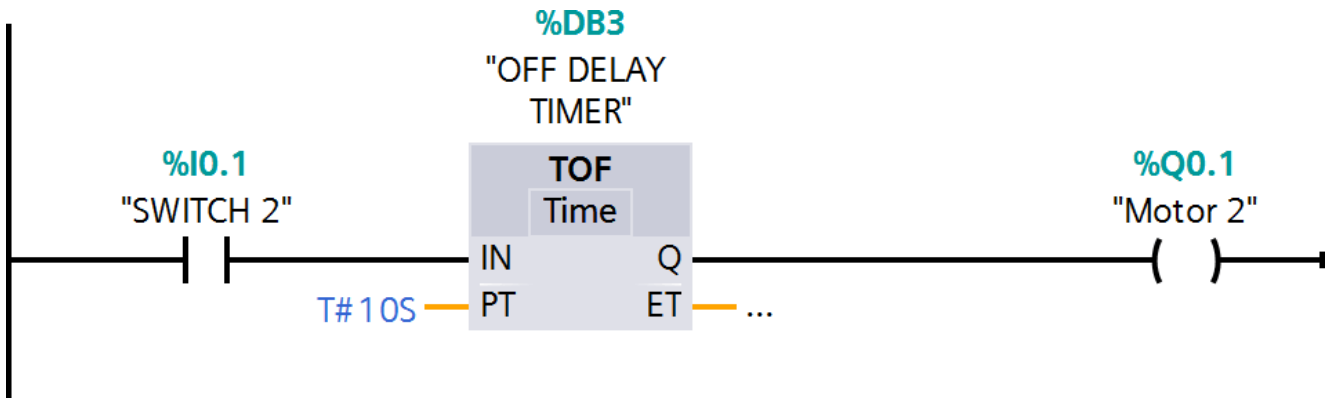
- ❖ OFF Delay Timer is used to make the Delay operation in processes.
- ❖ It can be control the inputs and outputs of the field instruments.

PROCEDURE:

1. Click PLC-1 in the project tree "1" then click program block "2" and click main OB1 "3" as shown in the below image
2. Now the below image shows (Object Block1) OB1 is created.
3. Now select the OFF-Delay Timer goes to instructions tab and click the Basic Instructions and select the Timer operations tab and then double click the TOF block.
4. Now the call options block is displayed. Then change the data block name. And click the automatic option finally select the ok button.
5. Now the Timer block is created in network. And also that one input and one output coil is created in same network for testing the timer block.
6. The input can be connected with IN of the Timer block. Then output coil is connected with Q of the Timer block.
7. Then given the addresses for input and outputs. And finally given the preset time for the Timer goes to double click the PT in timer block and write the values in seconds.
8. Then save the program and download after that goes to online mode and monitor the program.
9. When the status of the SWITCH 2(I0.1) changes from 0 to 1 the timer instruction will be executed and it will activate the MOTOR 2(Q0.1) immediately.

10. When the SWITCH 2(I0.1) status changes back to 0 then programmed time (PT) will start and after time MOTOR 2(Q0.1) will be OFF.

LADDER LOGIC PROGRAM:



CONCLUSION:

Thus, the ladder logic program of OFF Timer was written and implemented successfully using PLC Software.

RETENTIVE TIMER

AIM OF THE EXPERIMENT:

Write and implement a simple ladder logic program using Retentive timer.

APPARATUS REQUIRED:

- PLC Trainer kit
- Personal Computer Installed with PLC Software
- Ethernet cable
- Patch chords

THEORY:

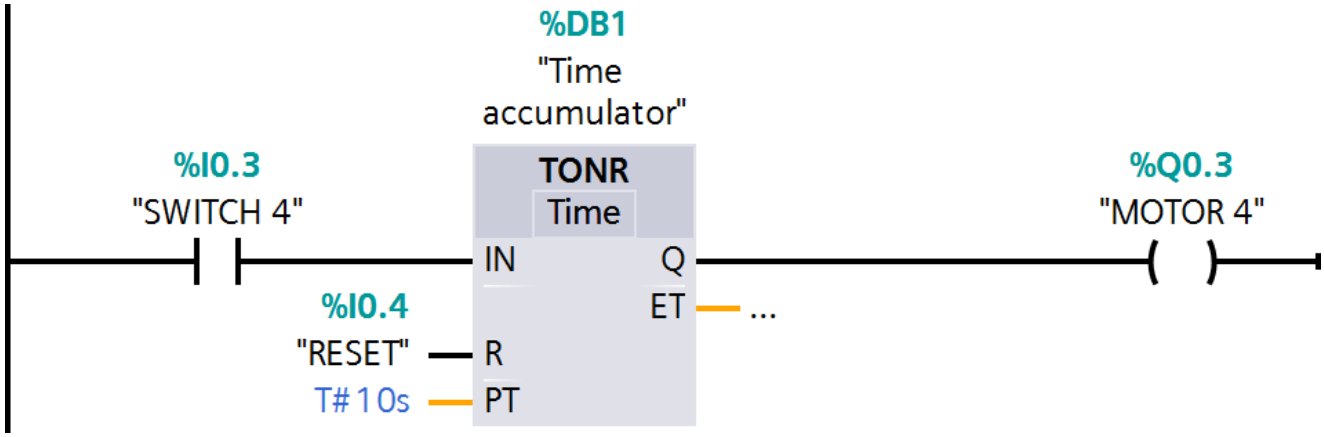
- Counts time base intervals when the instruction is true and retains the accumulated value when the instruction goes false or when power cycle occurs.
- The Retentive Timer instruction is a retentive instruction that begins to count time base intervals when rung conditions become true.
- The Retentive Timer instruction retains its accumulated value when any of the following occurs:
- Rung conditions become false.
- The processor loses power while battery backup is still maintained. And a fault occurs.

PROCEDURE:

1. Click PLC-1 in the project tree "1" then click program block "2" and click main OB1 "3" as shown in the below image
2. Now the below image shows (Object Block1) OB1 is created.
3. Now select the Accumulator timer goes to instructions tab and click the Basic Instructions and select the Timer operations tab and then double click the TONR block.
4. Now the call options block is displayed. Then change the data block name. And click the automatic option finally select the ok button.
5. Now the Timer block is created in network. And also, that one input and one output coil is created in same network for testing the timer block.
6. The input can be connected with IN of the Timer block. Then output coil is connected with Q of the Timer block.
7. Then given the addresses for input and outputs. And finally given the preset time for the Timer goes to double click the PT in timer block and write the values in seconds.
8. Then save the program and download after that goes to online mode and monitor the program.

9. When the status of the SWITCH 4(I0.3) changes from 0 to 1 the timer instruction will be executed and MOTOR 4(Q0.3) will start after 10s.
10. The MOTOR 4(Q0.2) will remain ON, even when the input status changes back to 0. The Reset (I0.4) is necessary to reset the timer or accumulated time.

LADDER LOGIC PROGRAM-



CONCLUSION:

Thus, the ladder logic program of Accumulator Timer was written and implemented successfully using PLC software.

Experiment No: 04

Write and implementation of simple ladder logic program using counter

UP COUNTER

AIM OF THE EXPERIMENT:

To study about the UP-Counter operation using PLC Software.

APPARATUS REQUIRED:

- PLC Trainer kit
- Personal Computer Installed with PLC Software
- Ethernet cable
- Patch chords

THEORY:

The counters are mainly used for counting the values from the field equipment. The UP counters can be used to counting the values in upward manner. It can be performing the incremental purpose.

PROCEDURE:

1. Click **PLC-1** in the project tree "1" then click **program block** "2" and click **main OB1** "3".
2. Now the below image shows (Object Block1) OB1 is created.
3. To select the UP- Counter block in plc goes to **Instructions** Tab and click the **Basic Instructions** then select the **CTU** block for UP Counter operation.
4. Now the UP-Counter call options block can be opened. Then change the **data block** name in that Name block. And then click the **ok** button.
5. The following details can be entered into the UP-Counter block is given below.

CU - Make the counter Input

Q- Counter Output

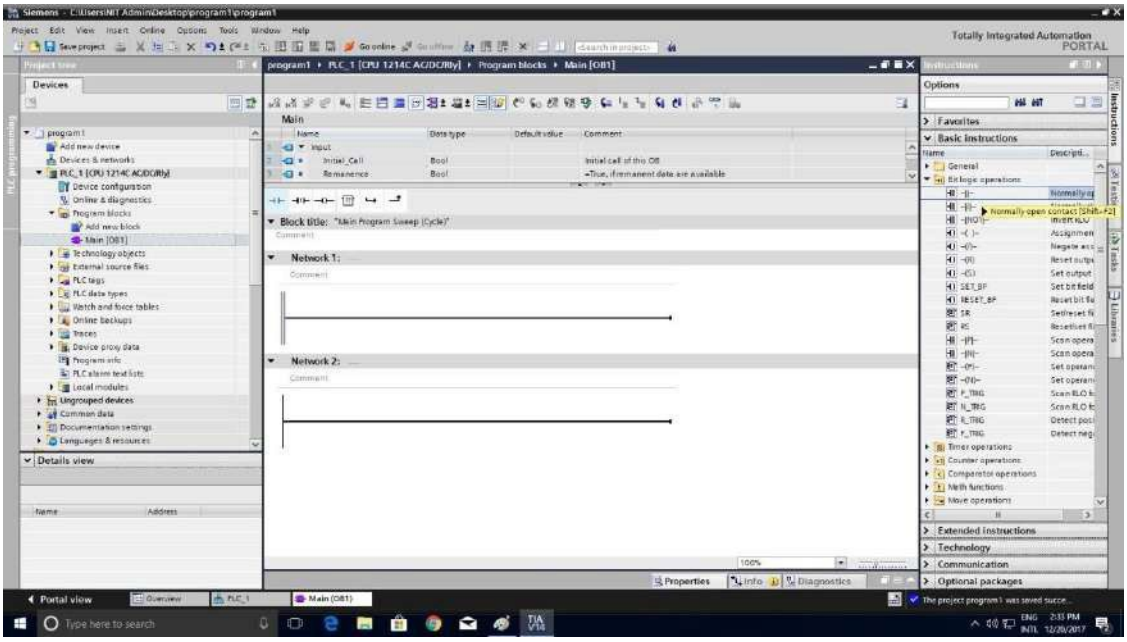
R - Reset input

CV - Count Value

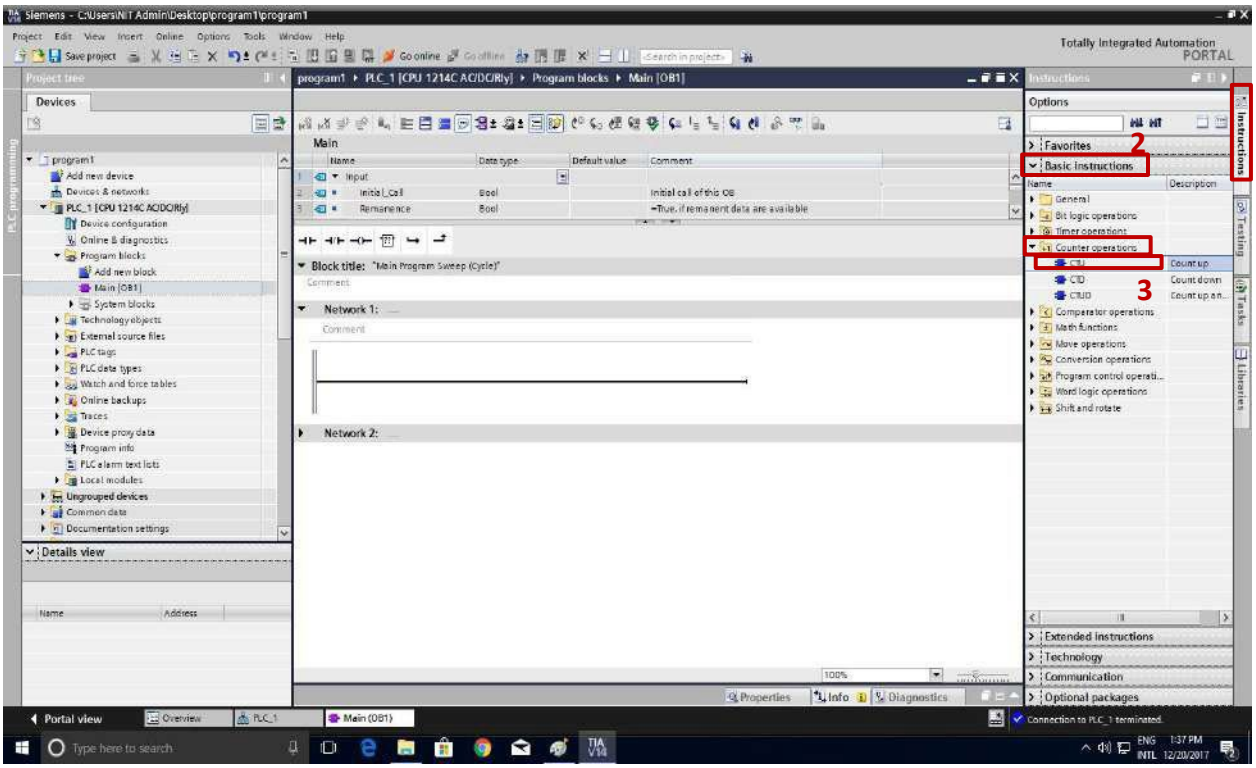
PV - Preset value (Set Value)

6. Then save the program and goes to online mode and monitoring the counter values before and after enabling the counter block.

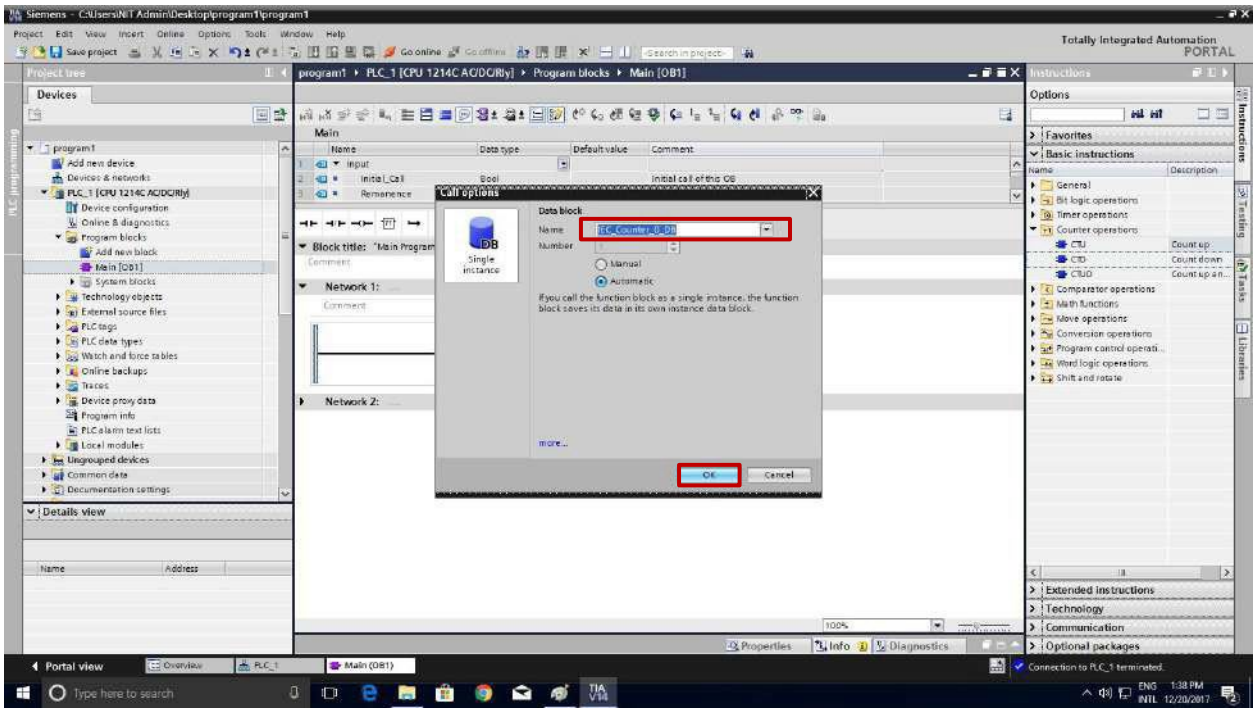
LADDER LOGIC PROGRAM:- 1.



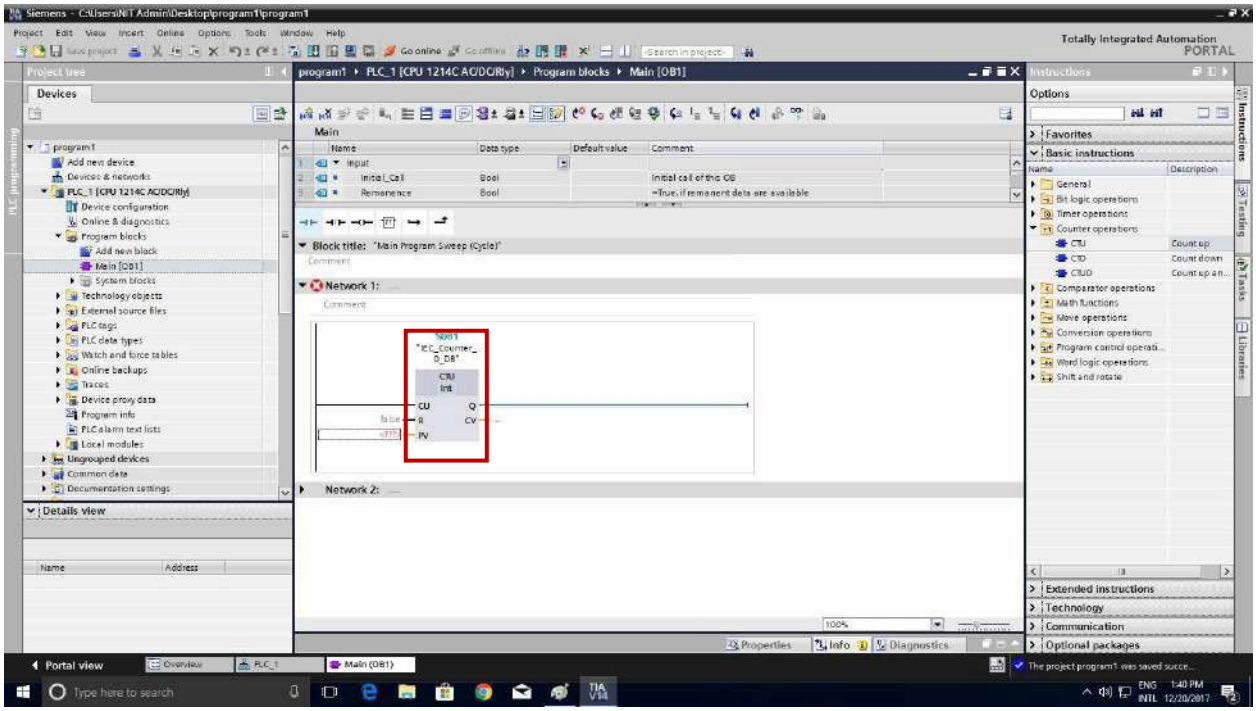
2.



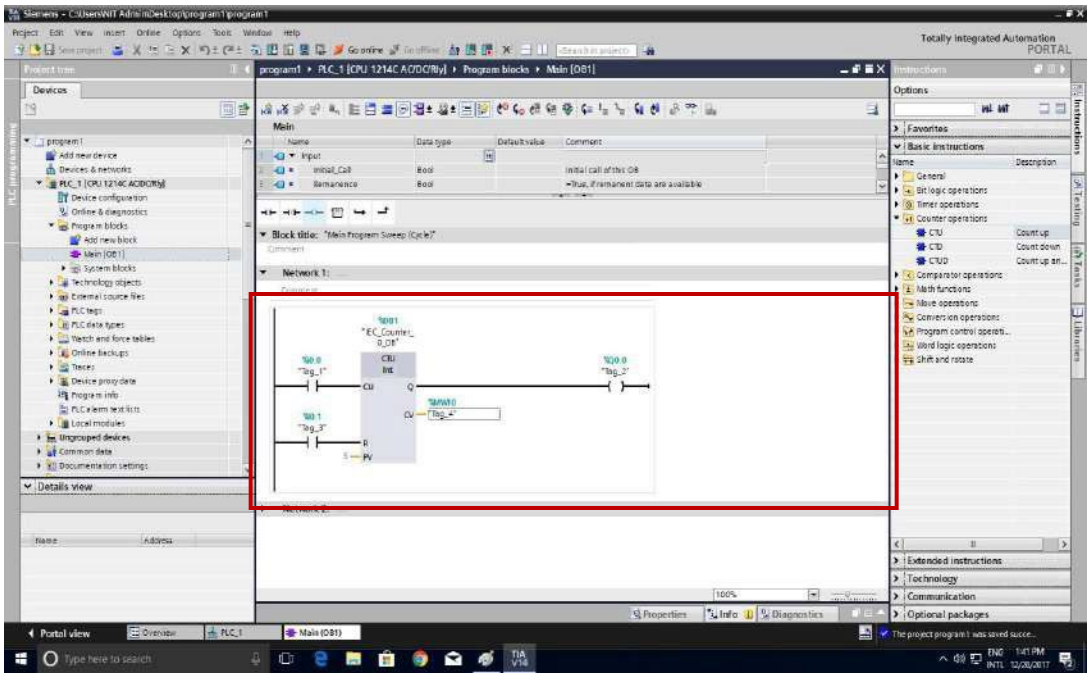
3.



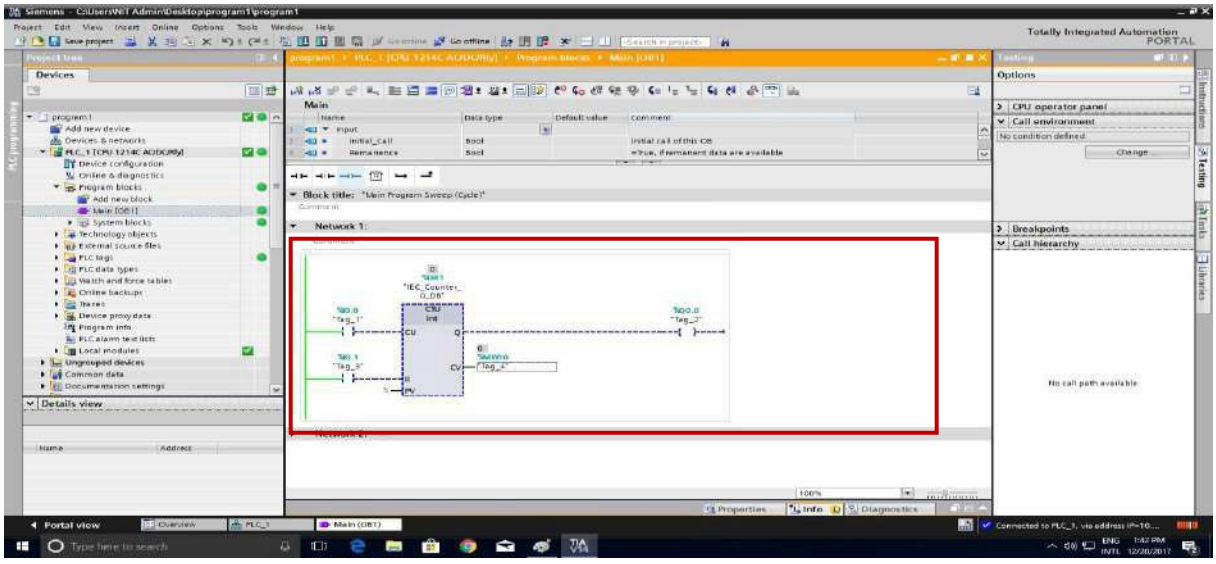
4. Now the UP-Counter block is created in network is shown below the image.



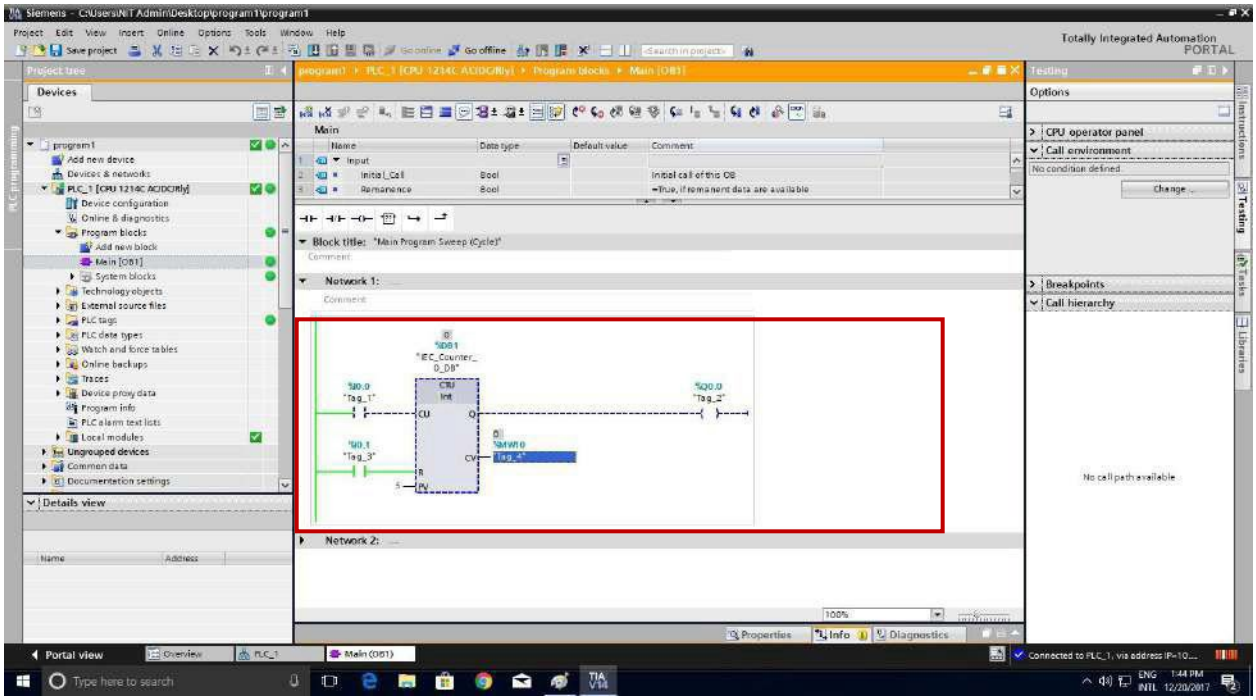
5. Before Enabling the Counter Block



6. After Enabling the Counter Block



7. After Resetting the Counter Block



CONCLUSION:

Thus, the UP-Counter operation was successfully performed using PLC software.

DOWN COUNTER

AIM OF THE EXPERIMENT:

To study about the Down Counter operation using PLC Software.

APPARATUS REQUIRED:

- PLC Trainer kit
- Personal Computer Installed with PLC Software
- Ethernet cable
- Patch chords

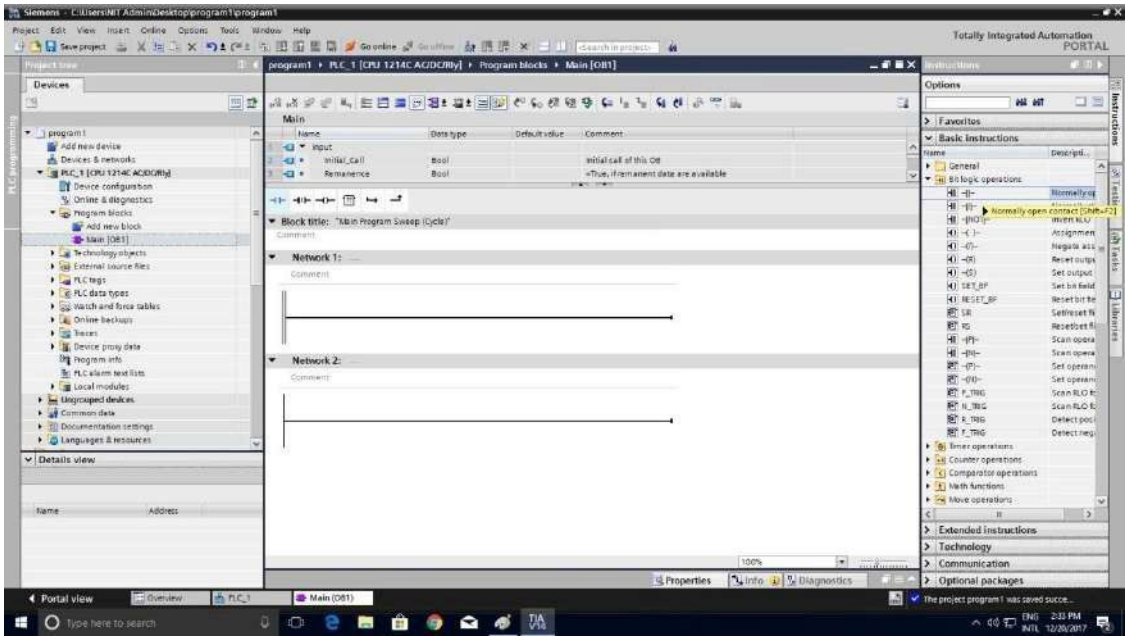
THEORY:

The counters are mainly used for counting the values from the field equipment. The DOWN Counters can be used to counting the values in down ward manner. It can be performing the decremental purpose.

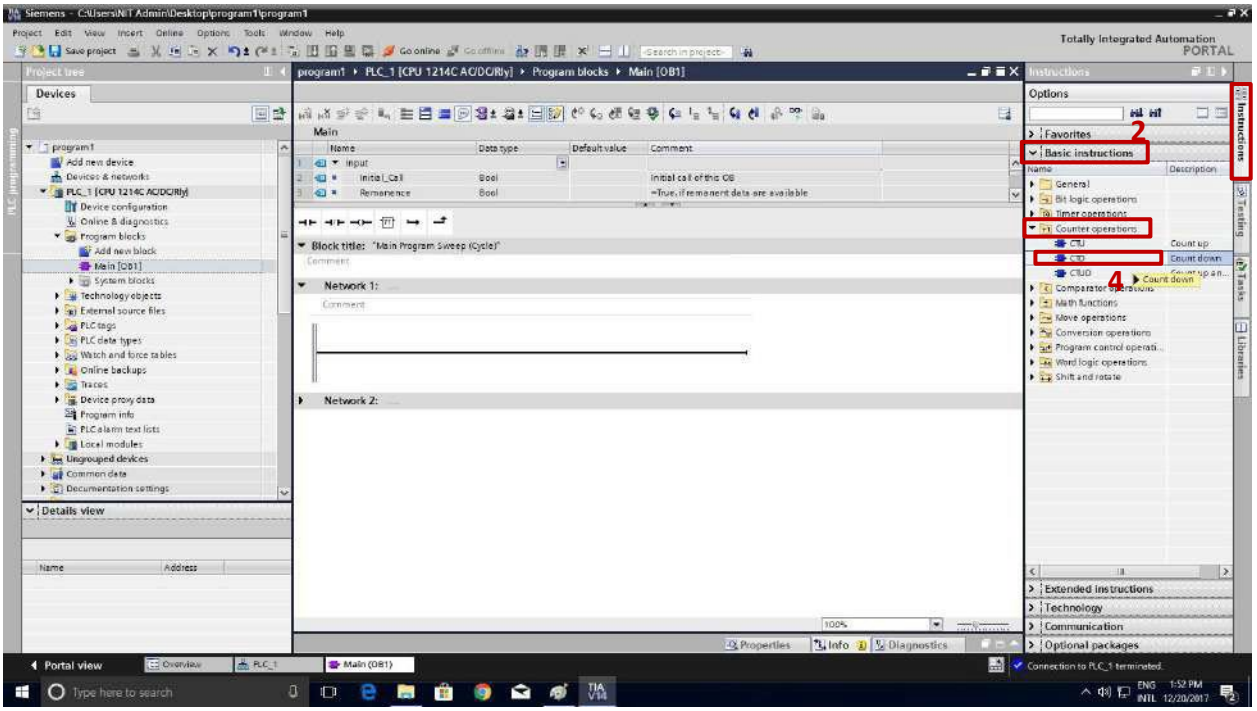
PROCEDURE:

1. Click **PLC-1** in the project tree "1" then click **program block** "2" and click **main OB1** "3".
2. Now the below image shows (Object Block1) OB1 is created.
3. To select the DOWN Counter block in plc goes to **Instructions** Tab and click the **Basic Instructions** then select the **CTD** block for DOWN Counter operation.
4. Now the DOWN Counter call options block can be opened. Then change the **data block** name in that Name block. And then click the **ok** button.
5. Now the **DOWN** Counter block is created in network.
6. The following details can be entered into the UP-Counter block is given below.
   CD - Make the counter Input
   Q- Counter Output
   LD - load Preset value (After reach the value of 0) CV
   - Count Value
   PV - Preset value (Set Value)
7. Then save the program and goes to online mode and monitoring the counter values before and after enabling the counter block.

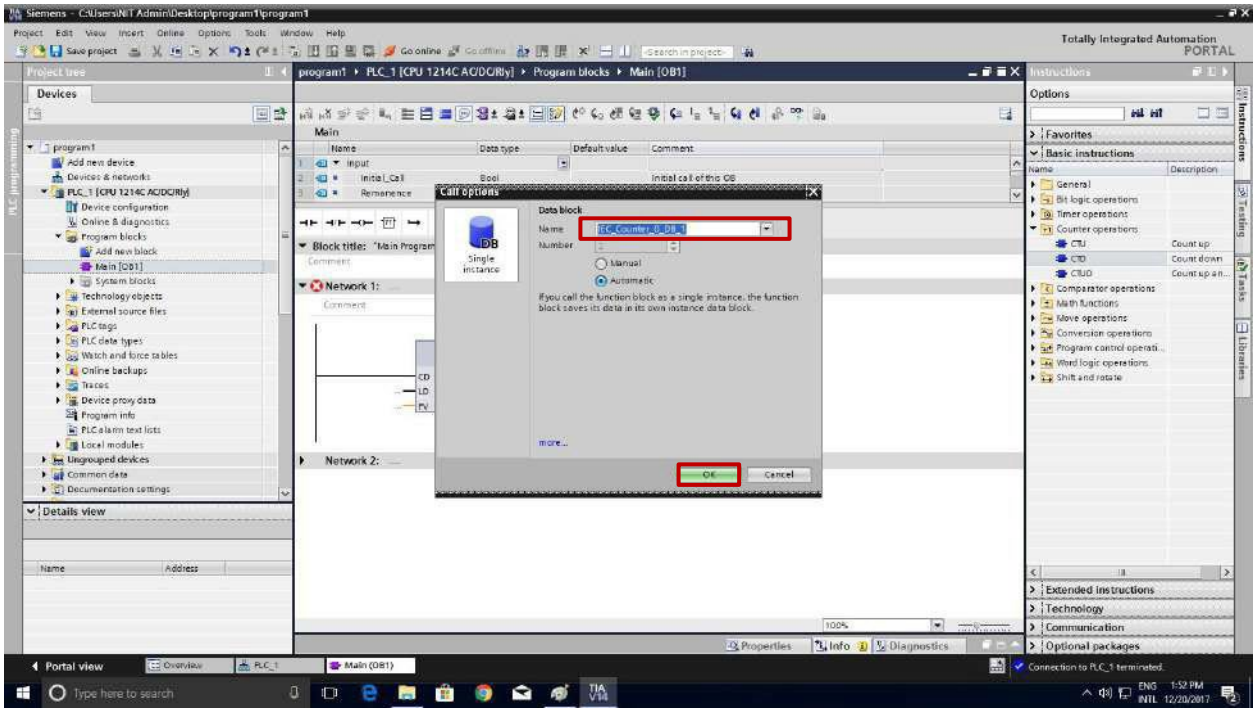
LADDER LOGIC PROGRAM:- 1.



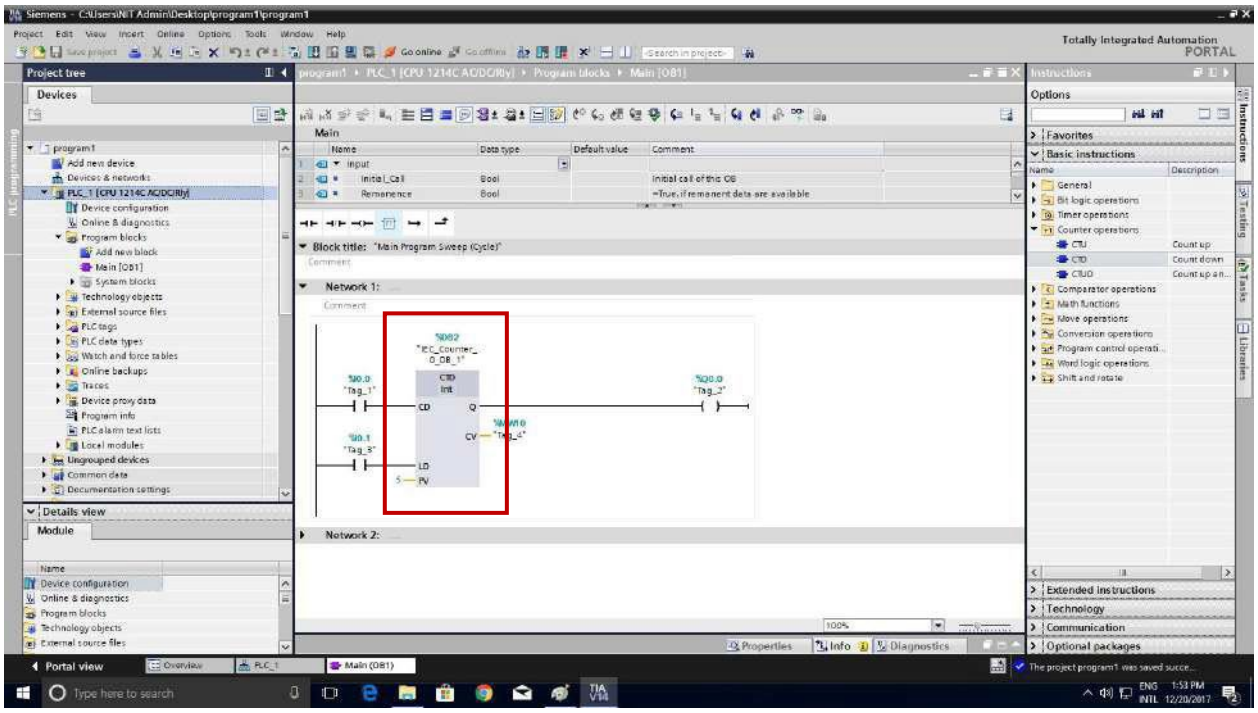
2.



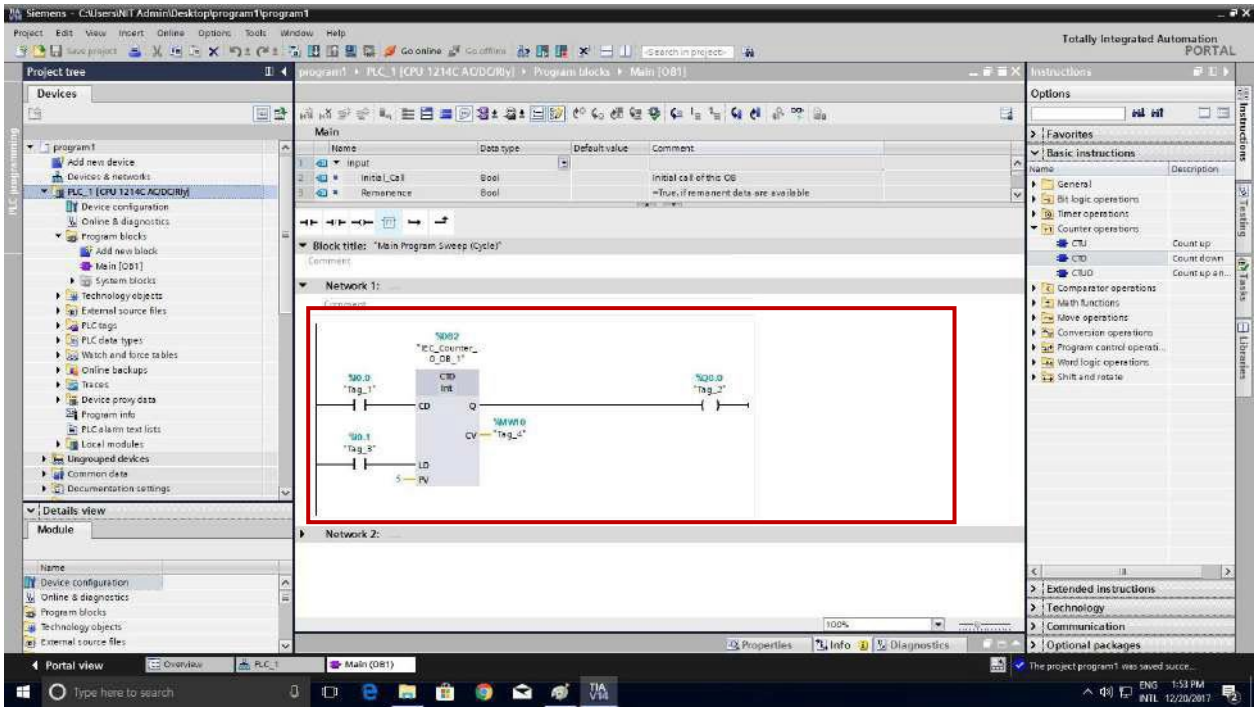
3.


Siemens - C:\Users\NIT Admin\Desktop\program1\program1
Totally Integrated Automation
PORTAL
Project tree
Devices
program1 ▸ PLC_1 [CPU 1214C AC/DC/Rly] ▸ Program blocks ▸ Main [OB1]
Main
Call options
Data block
Single instance
Manual
Automatic
If you call the function block as a single instance, the function block saves its data in its own instance data block.
more...
OK
Cancel
Network 1:
Network 2:
Instructions
Options
Favorites
Basic instructions
Extended instructions
Technology
Communication
Optional packages
Details view
Portal view
Overview
Main (OB1)
Properties
Info
Diagnostics
Connection to PLC_1 terminated.

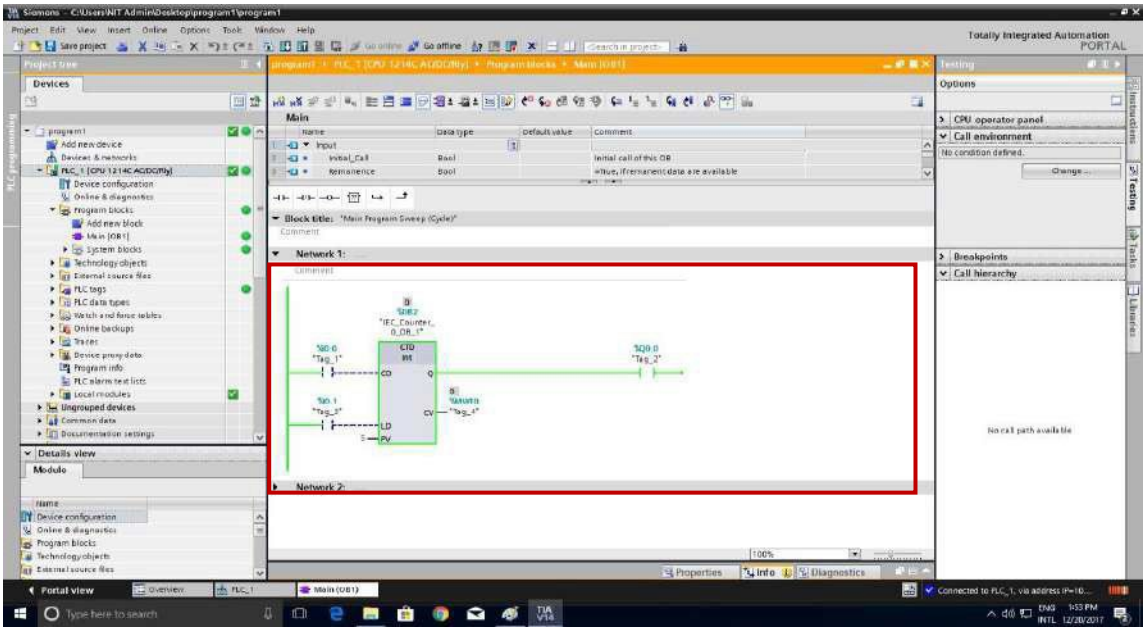
4.


Siemens - C:\Users\NIT Admin\Desktop\program1\program1
Totally Integrated Automation
PORTAL
Project tree
Devices
program1 ▸ PLC_1 [CPU 1214C AC/DC/Rly] ▸ Program blocks ▸ Main [OB1]
Main
Block title: "Main Program Sweep (Cycle)"
Network 1:
Network 2:
Instructions
Options
Favorites
Basic instructions
Extended instructions
Technology
Communication
Optional packages
Details view
Module
Portal view
Overview
Main (OB1)
Properties
Info
Diagnostics

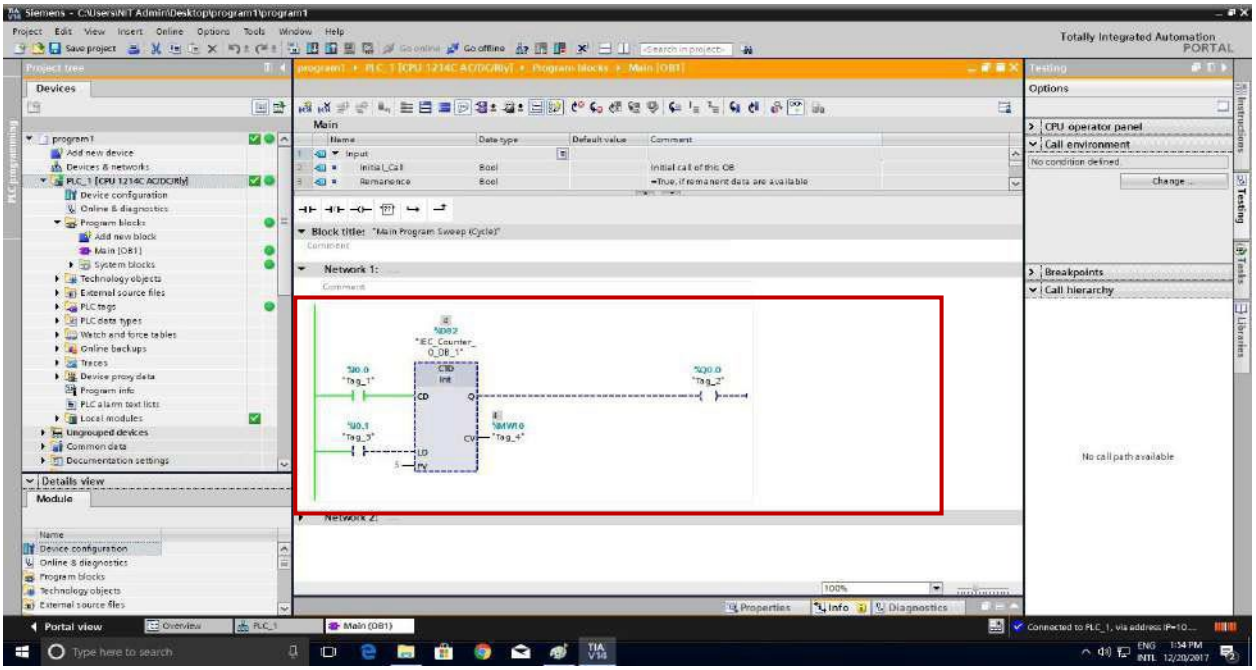
5. Before Enabling the Counter Block.



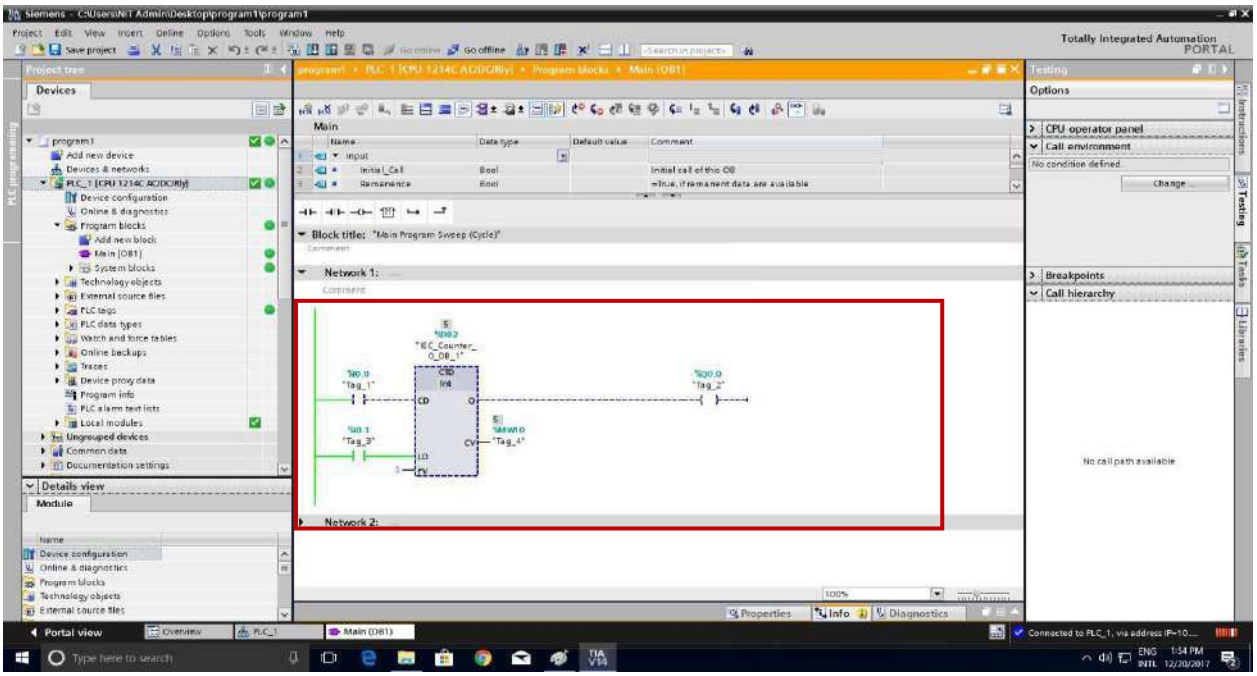
6. After Enabling the Counter Block



7. After Load Presetting the Counter Block



8.



CONCLUSION:

Thus, the DOWN Counter operation was performed successfully using PLC software**.**

Experiment No: 05

Write program on MOVE, Control statement, math function, data manipulation technique on PLC

MOVE

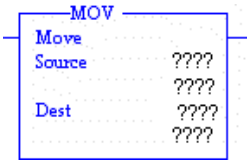
AIM OF THE EXPERIMENT:

Write a simple ladder logic program using MOVE instruction.

EQUIPMENTS REQUIRED:

- PLC software
- Desktop Computer

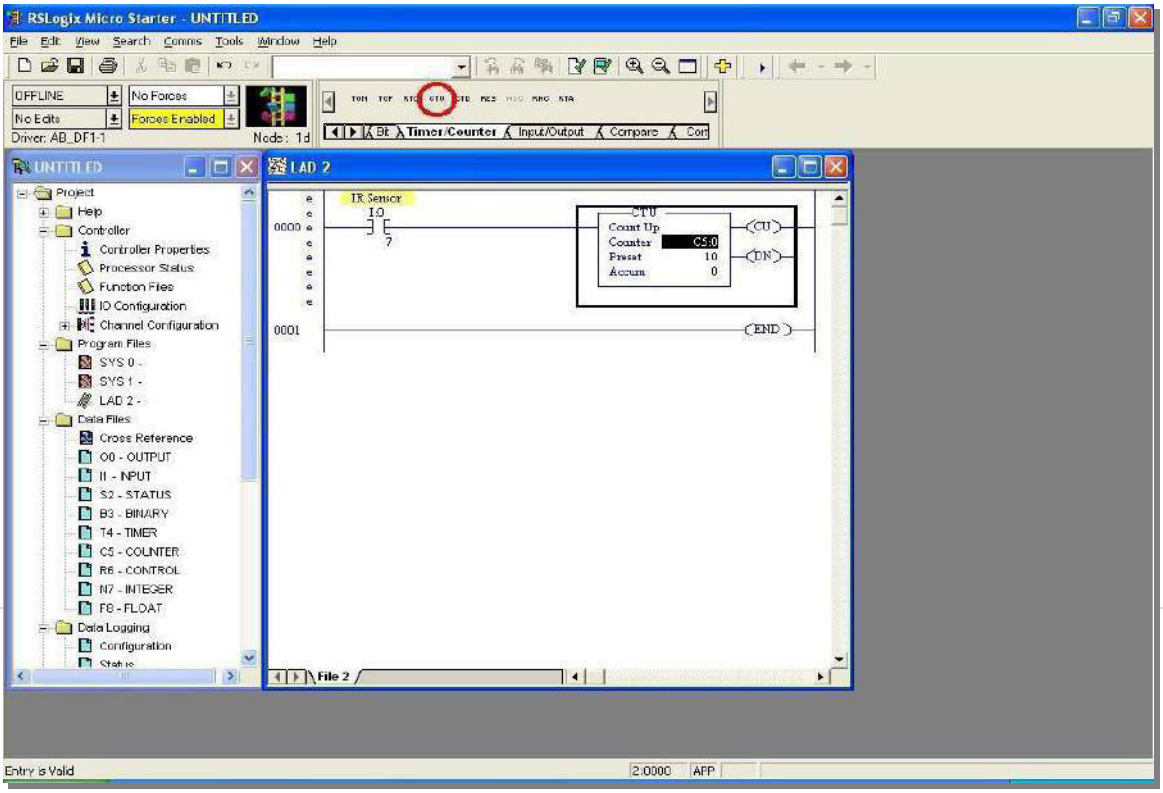
THEORY:



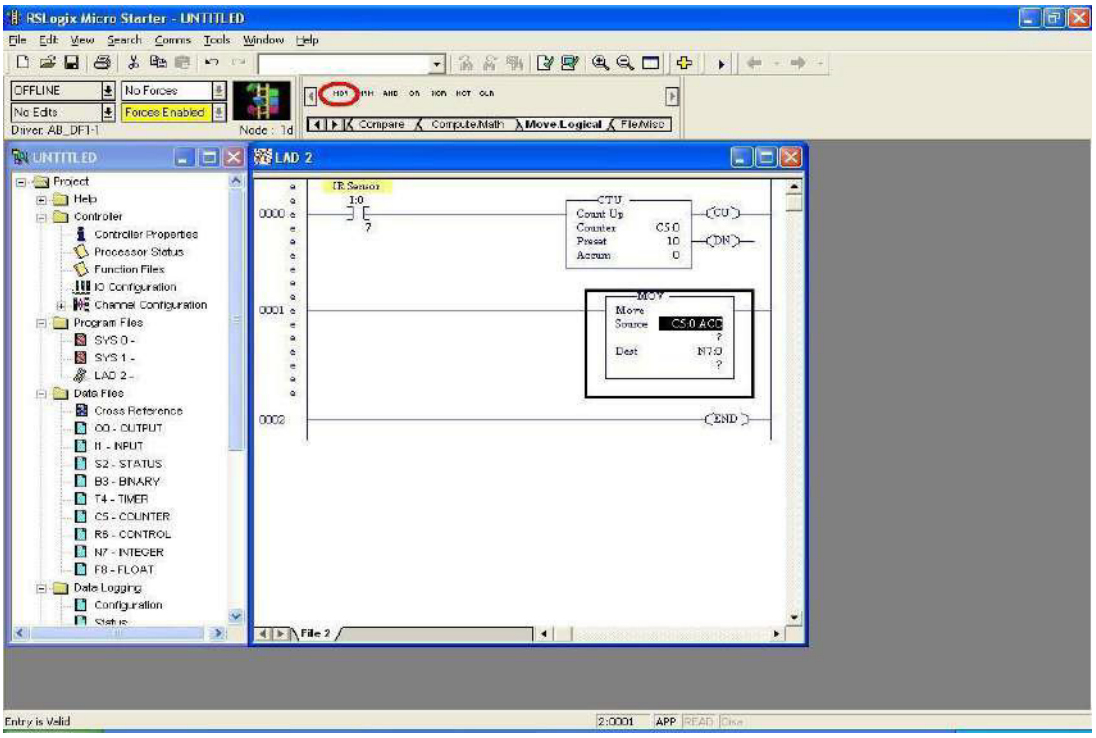
When rung conditions preceding this instruction are true, the MOV instruction moves a copy of the source to the destination each scan. The original value remains intact and unchanged in its source location.

PROCEDURE AND LADDER LOGIC PROGRAM:

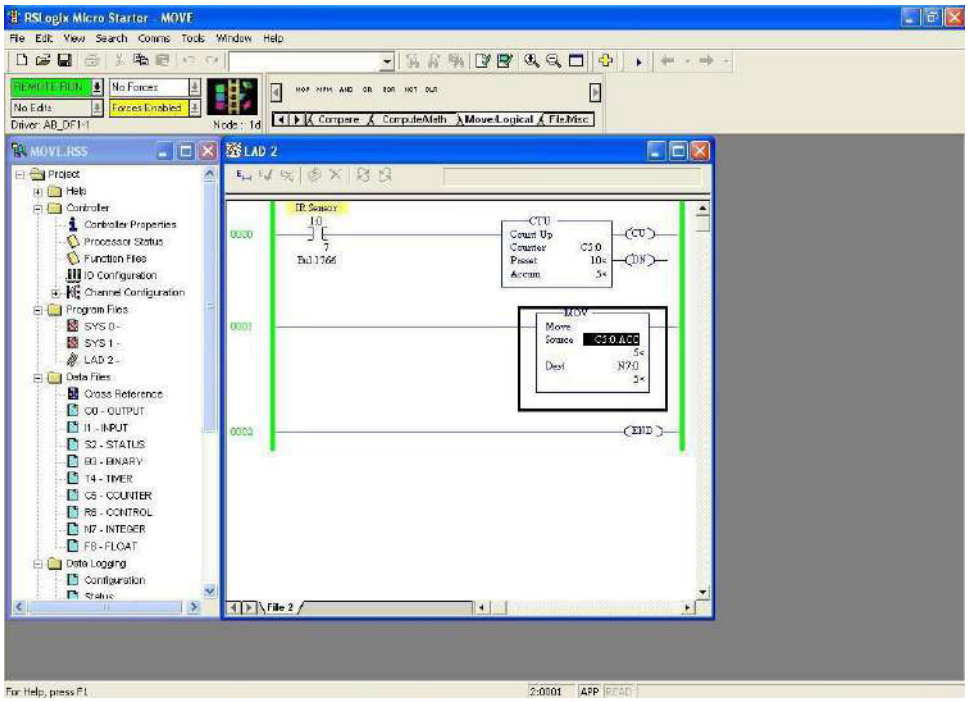
Take one NO bit and assign address I:0/7and comment to IR Sensor and take Counter and assign Counter to C5:0 and Preset to 10 is shown below.



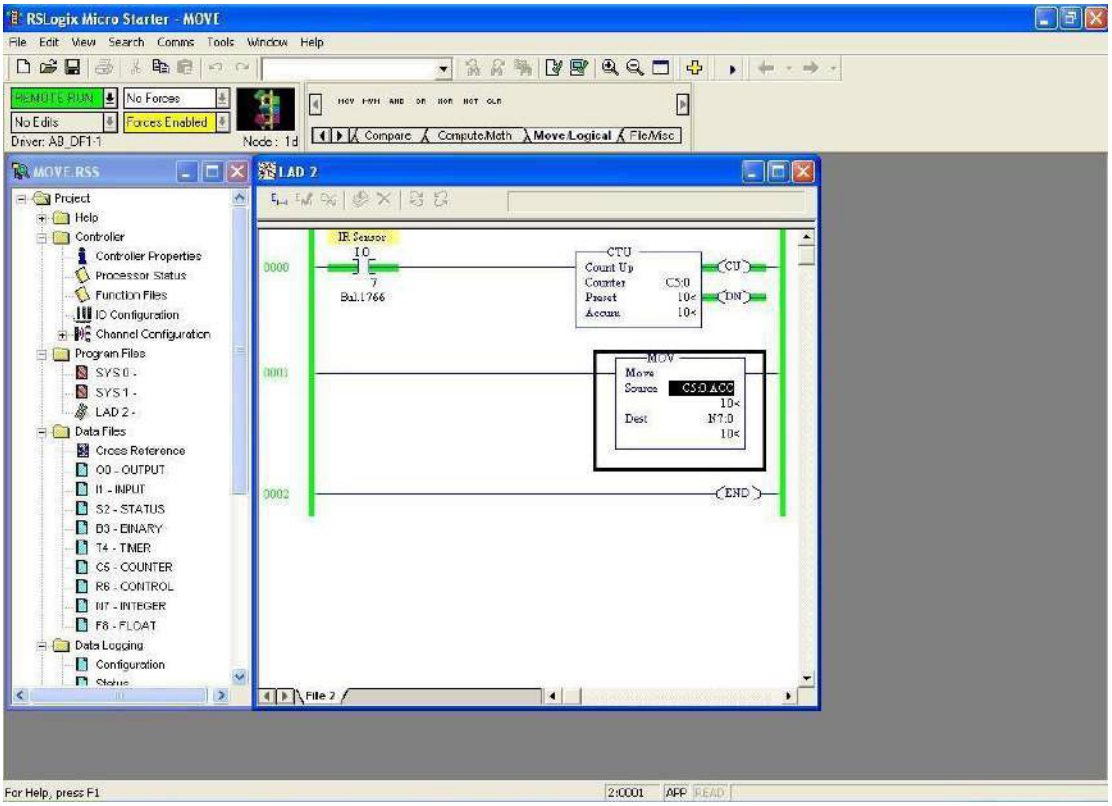
- Take Move instruction form Move/Logical Instruction and assign Source to C5:0.ACC and assign Dest. to N7:0 is shown below.



- Verify and download and run the program.
- When any obstacleinfrontofI:0/7(IR Sensor) then counter C5:0 will start counting. When counter Accumulator value is increase and move instruction Move the Counter accumulator value in Destination N7:0 is shown below.



- When any obstacleinfrontofI:0/7(IR Sensor) then counterC5:0will start counting. When counter Accumulator value is increase and move instruction Move the Counter accumulator value in DestinationN7:0 is shown.



CONCLUSION:

Thus, the MOVE operation was studied and performed successfully using PLC software**.**

CONTROL STATEMENT

AIM OF THE EXPERIMENT:

Write a simple ladder logic program using Control statement (Jump & Label instruction).

EQUIPMENTS REQUIRED:

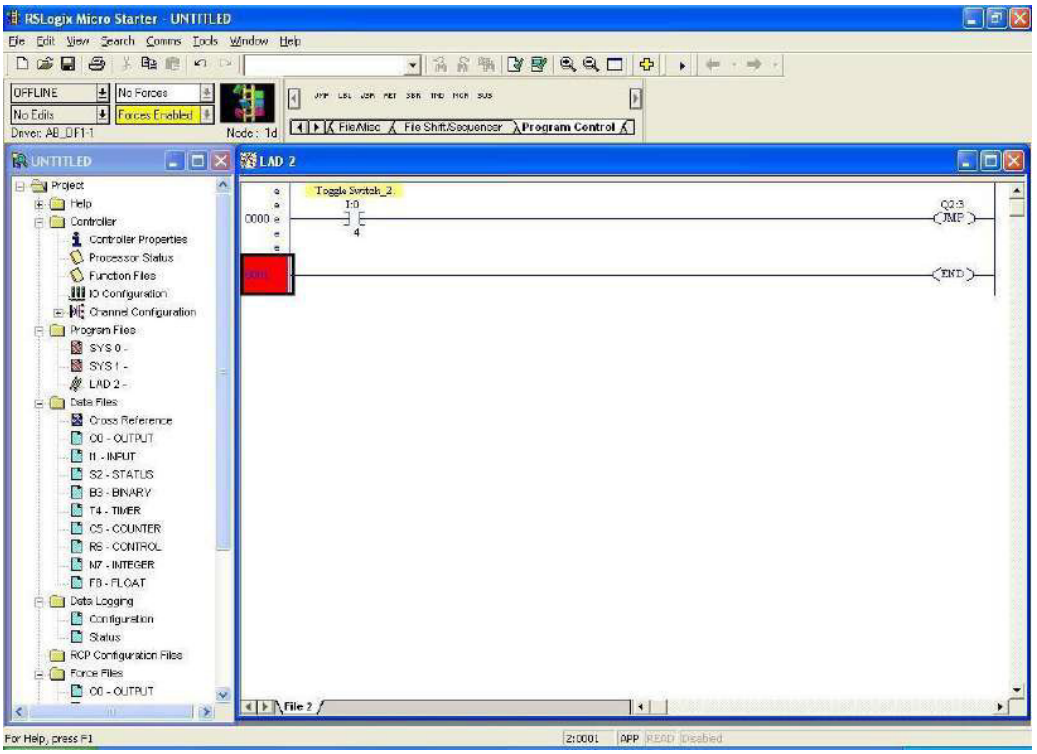
- PLC software
- Desktop Computer

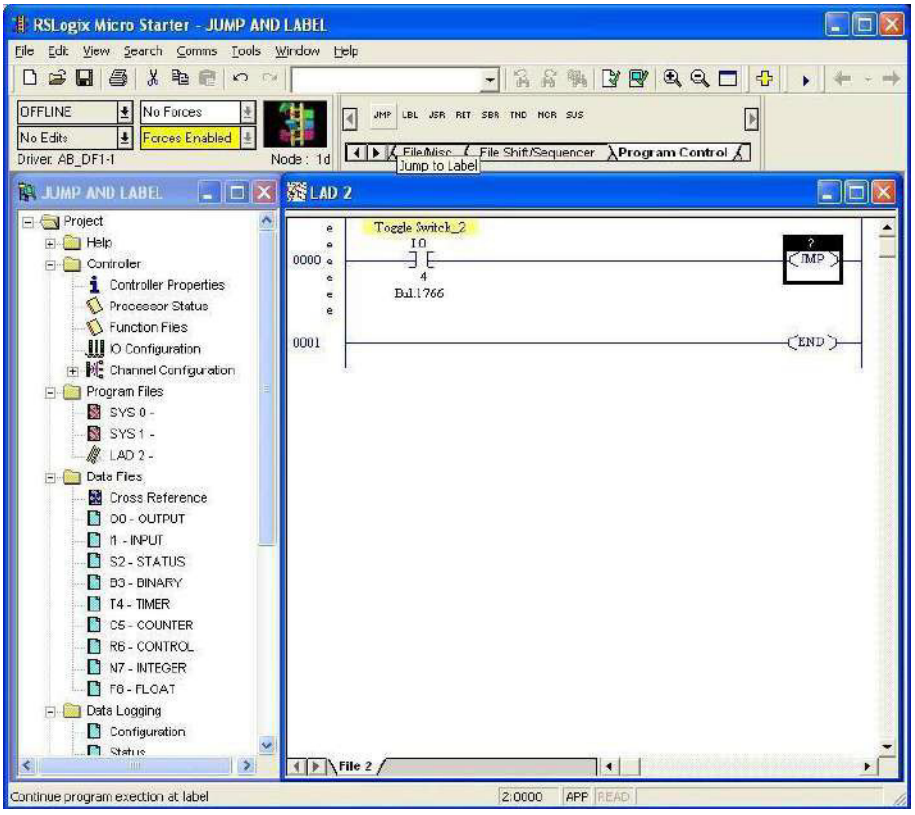
THEORY:

When the rung condition for this output instruction is true, the processor jumps forward or backward to the corresponding label instruction (LBL) and resumes program execution at the label. More than one JMP instruction can jump to the same label. Jumping forward to a label saves program scan time by omitting a program segment until needed. Jumping backward lets the controller execute program segments repeatedly.

PROCEDURE AND LADDER LOGIC PROGRAM:

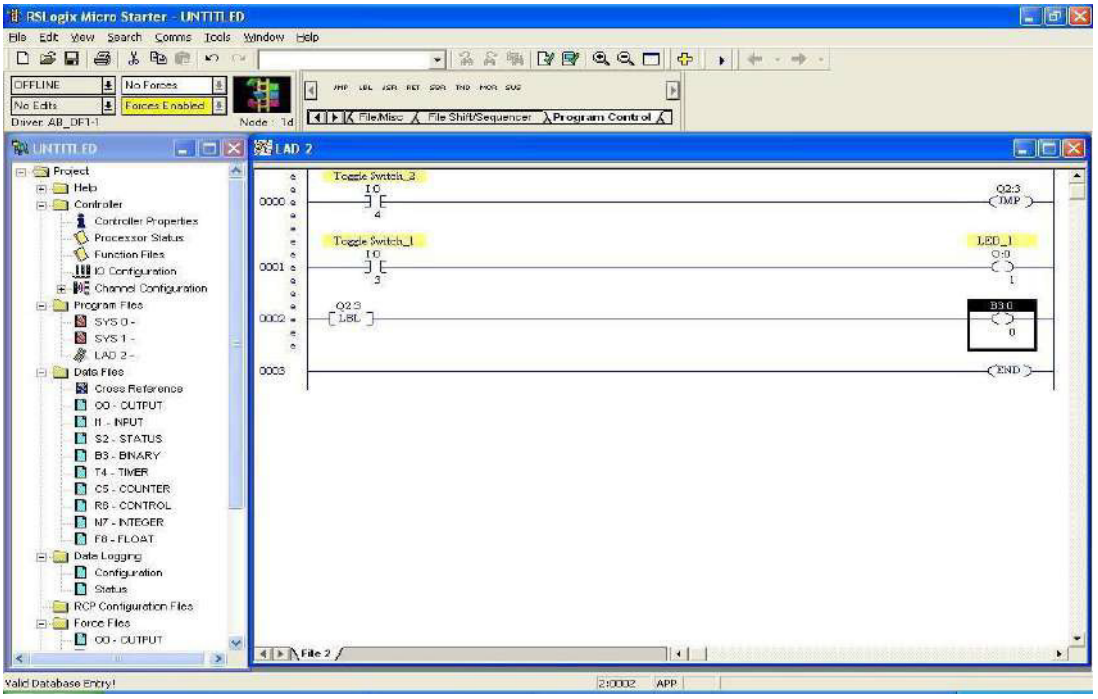
Take NO Bit and assign address I0:0/4 and comment to Toggle Switch_2 and take JMP instruction from Program control instruction is shown below.

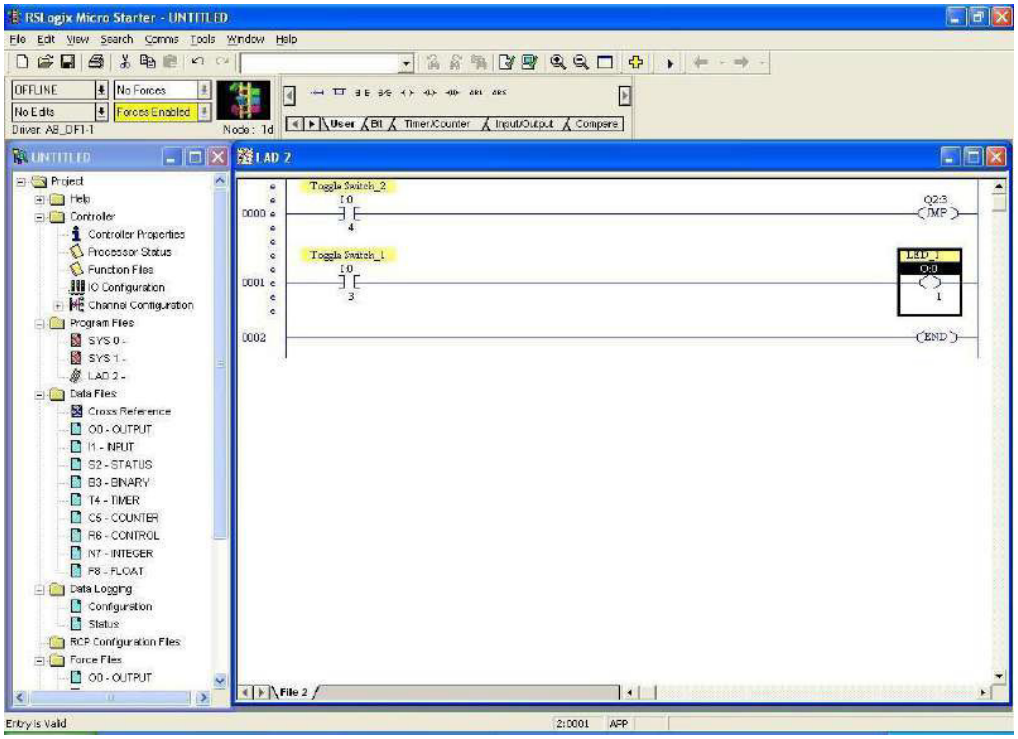




- Assign address to JMP instruction to 3 is shown below.

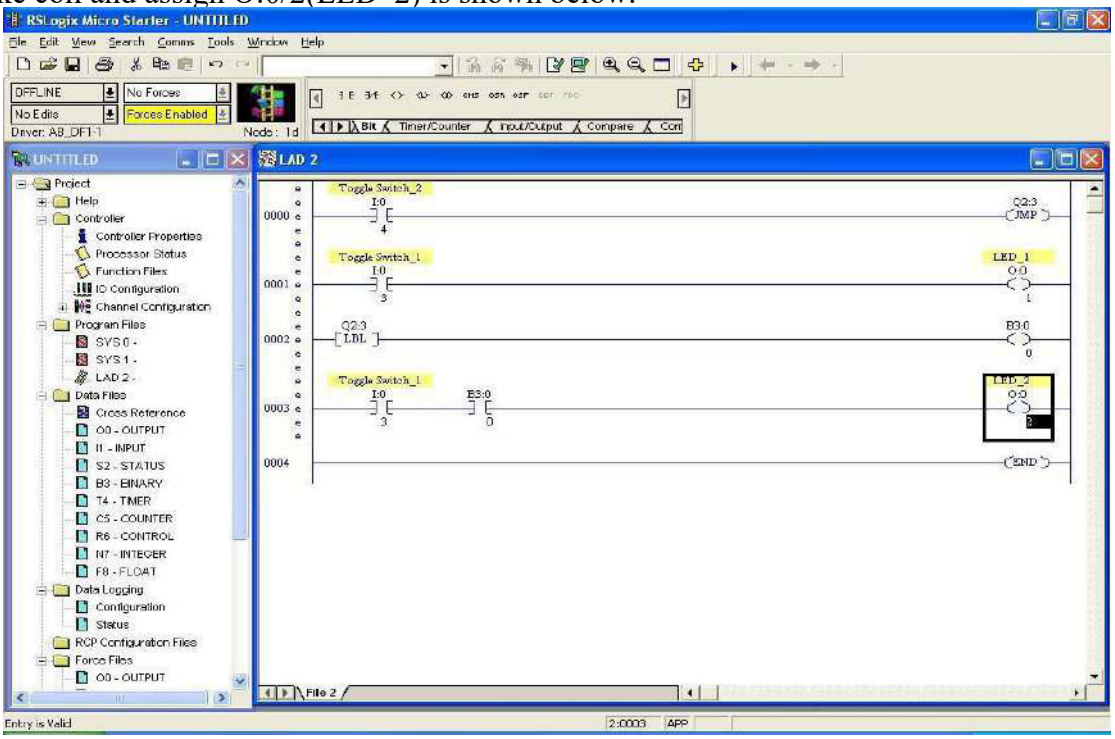
- Take NO bit and assign address I:0/3(ToggleSwitch_1) and take coil and assign address O0:0/1 (LED_1) is shown below.





- Take LBL instruction from Program Control instruction and assign address Q2:3and take Coil and assign addressB3:0/0 is shown below.

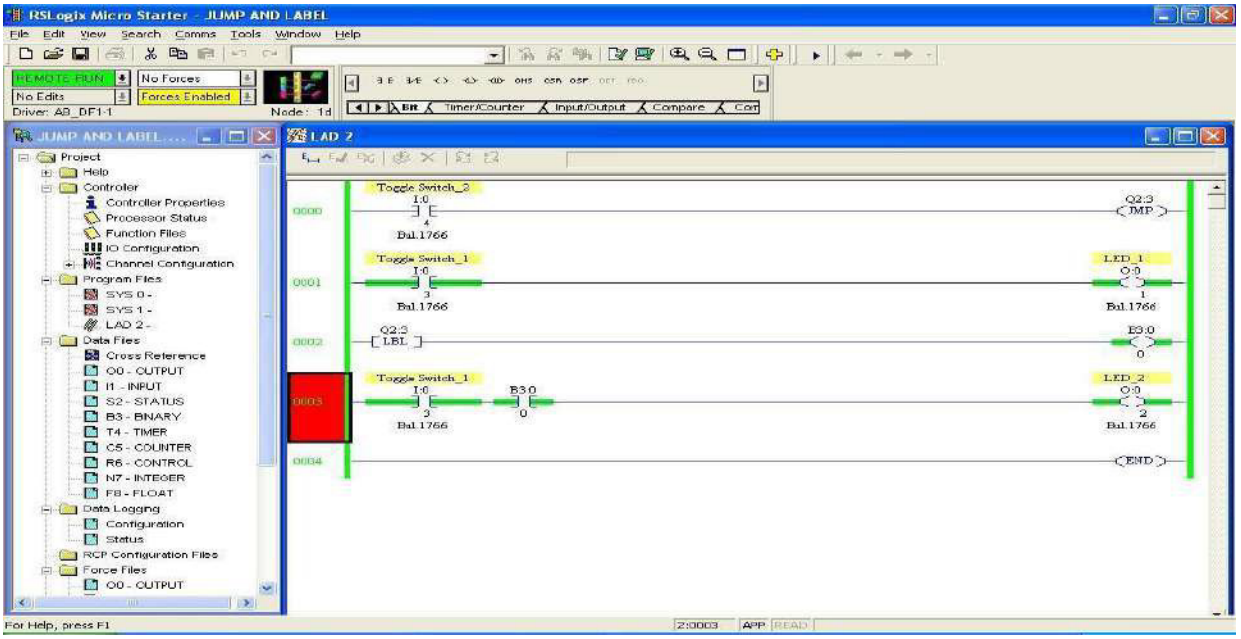
- Take NO bit and assign address I:0/3 (ToggleSwitch_1) and take another NO bit and assign B3:0/0 and take coil and assign O:0/2(LED_2) is shown below.



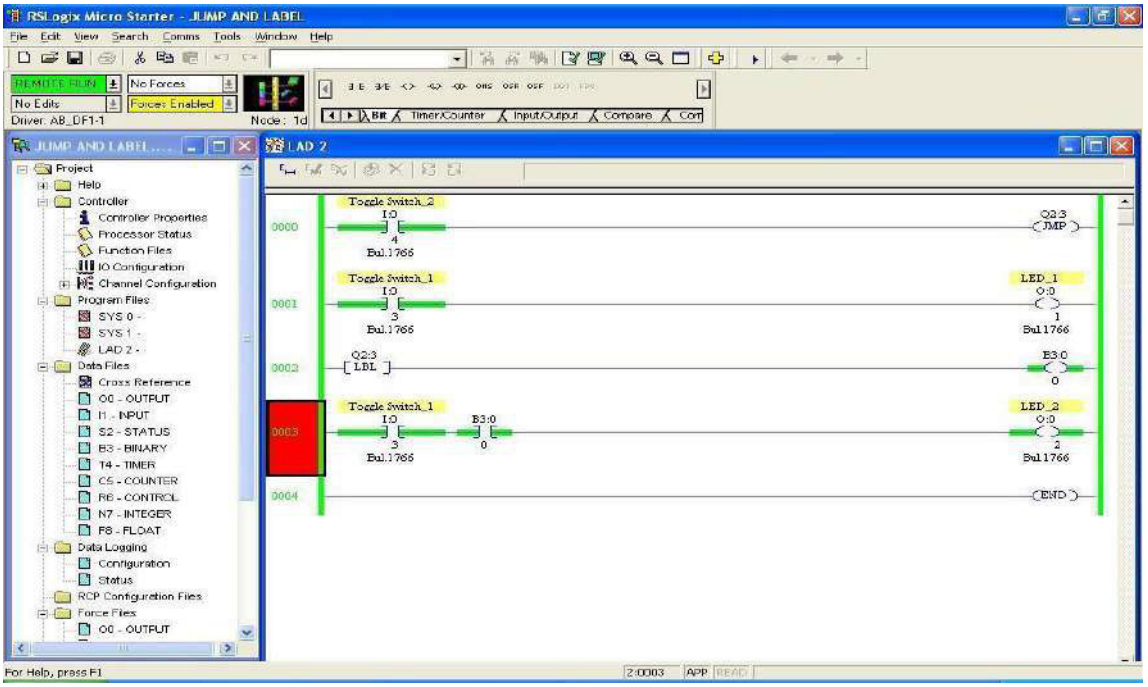
- Verify, download and run the program of Jump and Label instruction.
  -
- When you enable the I:0/3(ToggleSwitch_1) then output O:0/1(LED_1) and

  O:0/2(LED_2) will be ON is shown below.

- When you enable I:0/4(ToggleSwitch_2) then JUMP instruction is ON.



- When you enable the I:0/3 (Toggle Switch_1) then output O:0/1 is off because execution of instruction is Jump on network 3(0003) means outputO:0/2(LED_2) will be ON is shown below.



CONCLUSION:

Thus, the Control operation was performed and verified successfully using PLC software.

MATH FUNCTION

AIM OF THE EXPERIMENT:

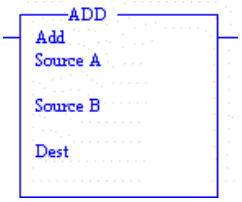
Write a simple ladder logic program using Math instruction (Addition).

EQUIPMENTS REQUIRED:

- PLC software
- Desktop Computer

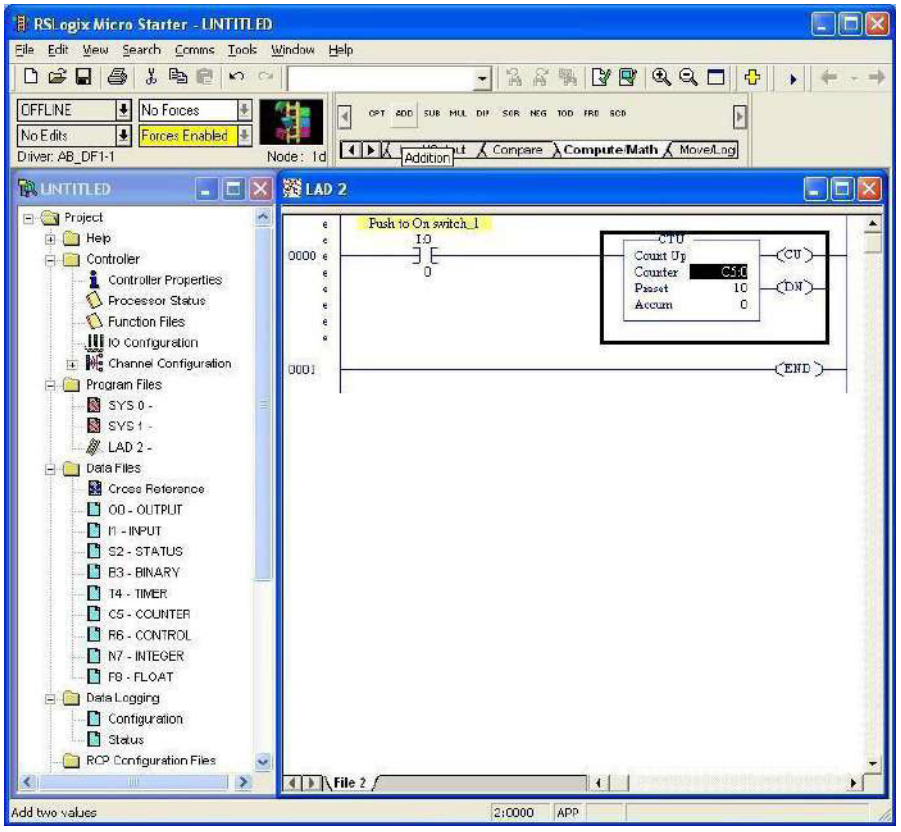
THEORY:

Use the ADD instruction to add one value to another value (Source A to Source B) and place the SUM in destination.

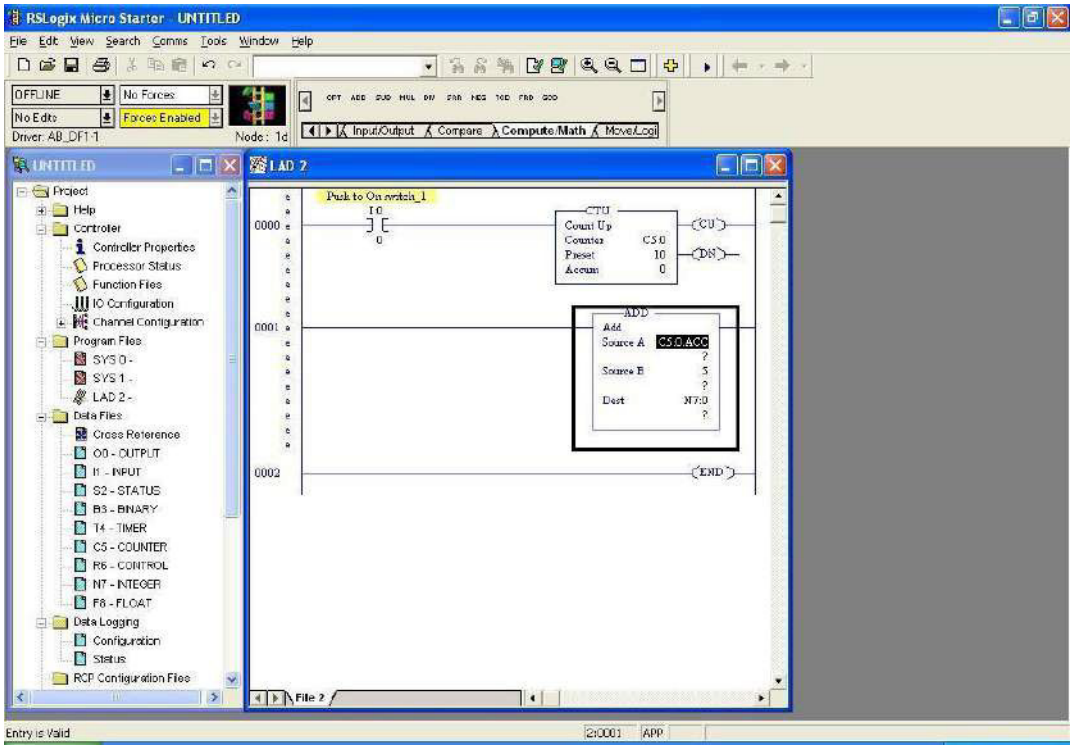


PROCEDURE AND LADDER LOGIC PROGRAM:

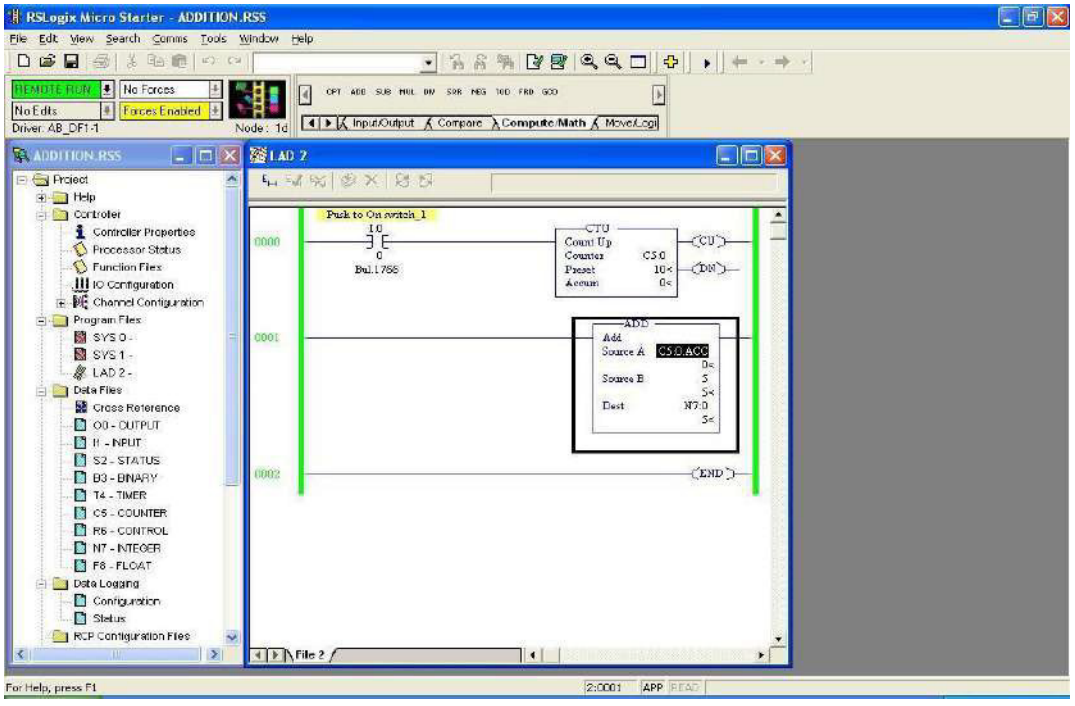
- Take one Examine if closed bit and assign it I: 0/0 (Pushbutton Switch_1) and Take one up counter and assign it Counter to C5:0 and Preset to 10 and Accumto0 is shown below.



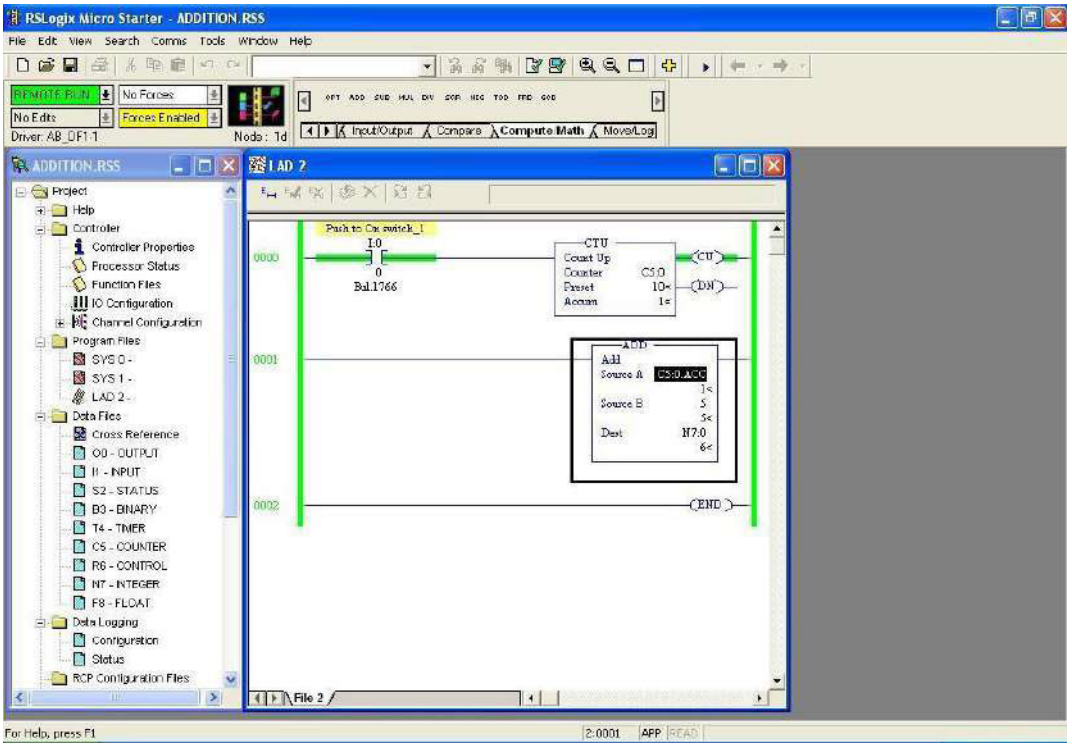
- Take Addition instruction from Compute Math (in instruction Bar) and assign it Source A to C5:0.ACC and Source B to 5 and Destination to N7:0 is shown below.



- Compile and Download and Run the Program.



- When you press I:0/0 (Pushbutton Switch_1) counter C5:0 start counting and addition is shown in N7:0.



Use of Addition Instruction

CONCLUSION:

Thus the ADDITION operation was performed and verified successfully using PLC software**.**

DATA MANIPULATION

AIM OF THE EXPERIMENT:

Write a simple ladder logic program using data manipulation technique (Bit shift left).

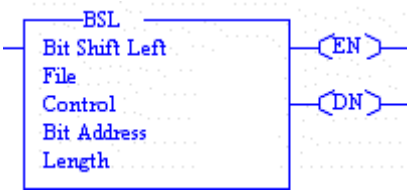
EQUIPMENTSREQUIRED:

- PLC software
- Desktop Computer

THEORY:

BitshiftinstructionisusedtoshiftthebitinanydirectionasuserwantsuchasBitshiftLeft (BSL),Bit Shift Right(BSR).

Bit shift left (BSL): When the rug goes from false to true, the controller set the enable bits (Enable Bit 15) and the data block is shifted to the left one bit position. The specified bit at the bit address is shifted into the first bit position. The last bit is shifted out of array and stored in unloaded bit. The shift is completed immediately



The following general information applies to bit shift instruction

Entering parameter:

Enter the following parameter when programming these instruction

- File is the address of the bit array you want to manipulate. You must use the file indicator (#) in the bit array address.
- Control is the address of control element that store the status byte of instruction, the size of array (in number of bits).

The control element is shown in below

	15	13	11	10	00
Word 0	EN	DN	ER	UL	Not used
Word 1	Size of bit array (number of bits)				
Word2	Reserved				

Status bits of control element may be addressed by

mnemonics. They included

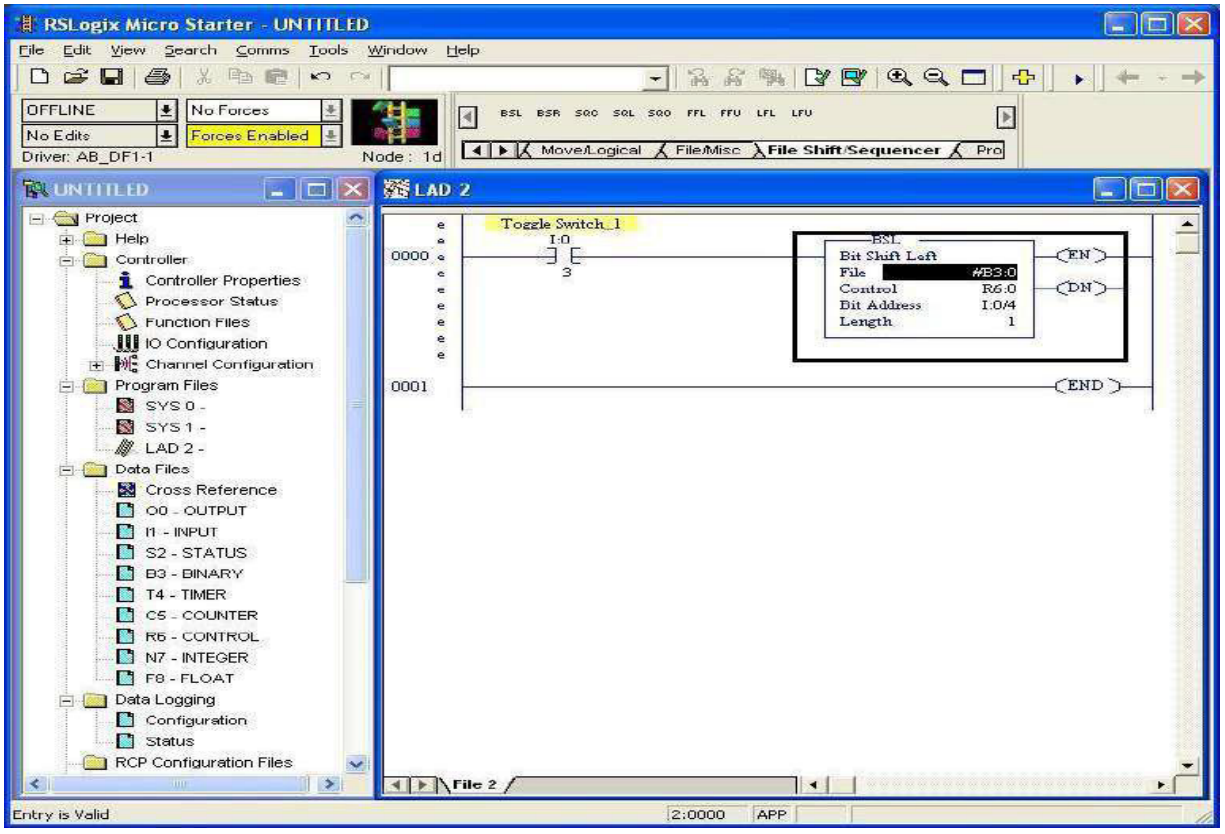
- Unloaded bit UL(Bit 0)is

The instructions output.

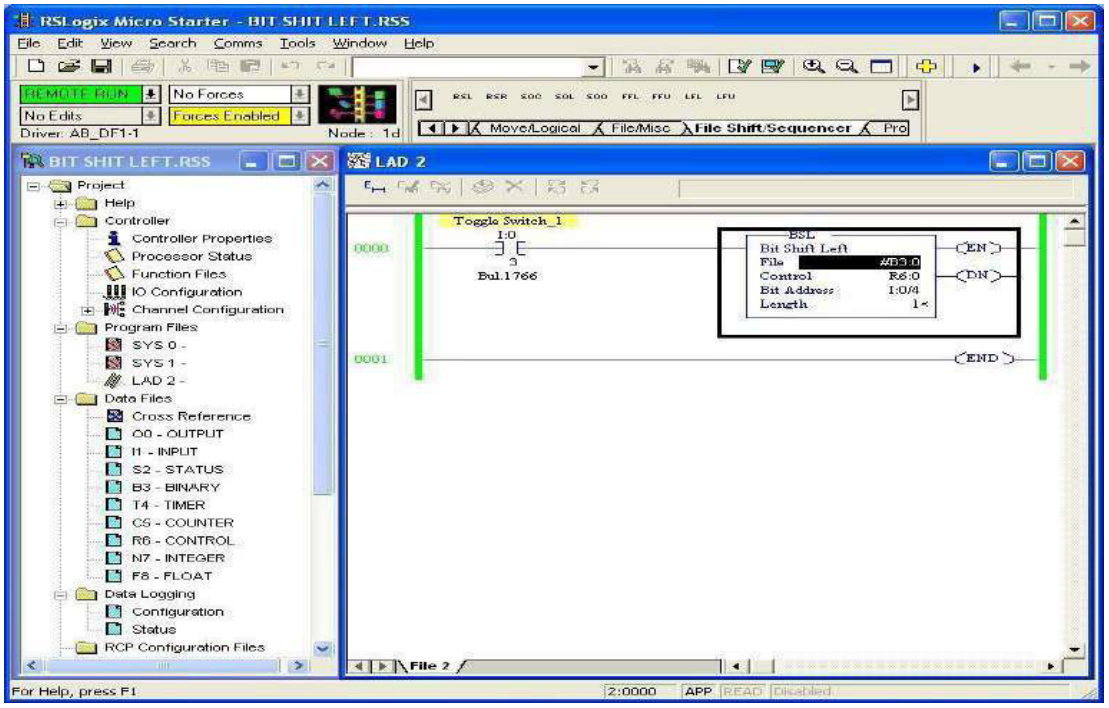
- Error bit ER (Bit11) when set, indicates the instruction detected an error such entering a negative number for length or position.
- Done Bit DN (Bit 13) when set indicate the bit array shifted one position.
- Enable bit EN (Bit 15) is set on a false to true transition of the rung and indicate the instruction is enabled**.**
- Bit address is the address of the source bit. The status of this bit is inserted in either the first (lowest) bit position (BSL) or last (highest)bit position (BSR).
- Length (size of bit array) is the number of bits in bit array up to 1680 bits**.**

PROCEDURE AND LADDER LOGIC PROGRAM:

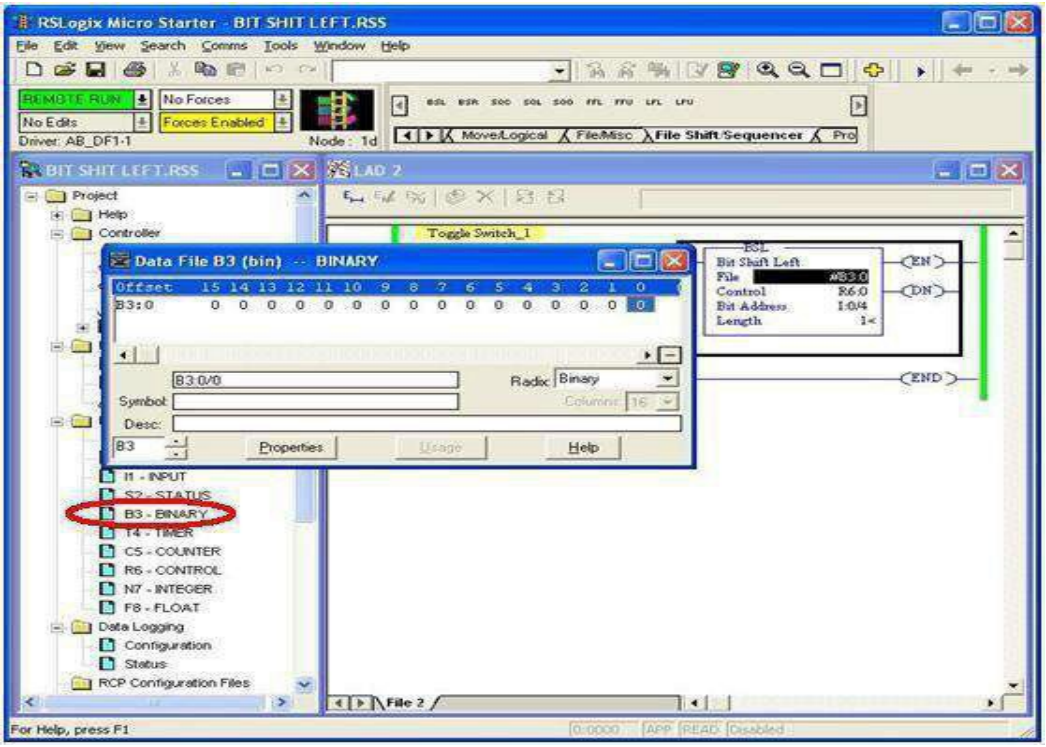
- Take No bit assign I:0/3 (Toggle Switch_1) and select a File shift / sequencer instruction in Tool bar then take a BSL (Bit Shift Instruction) and assign it Fileto #B3:0 and Control to R6:0 and File Address I:0/4 and Length 1is shown below.



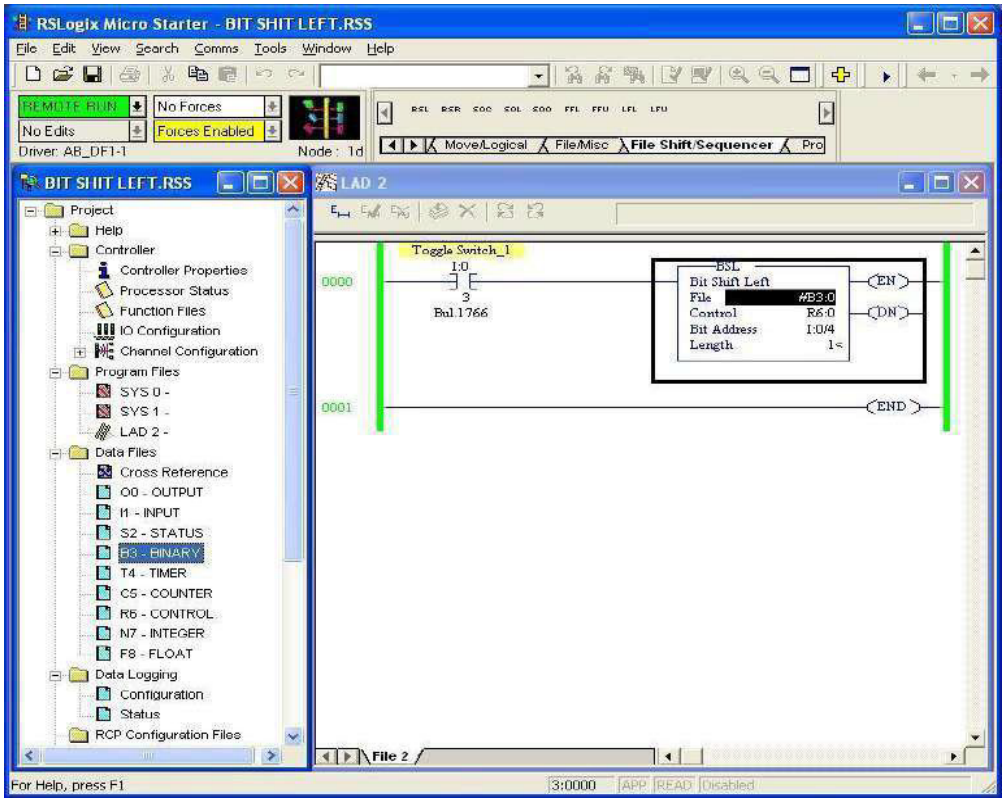
- Compile and download and run the program.



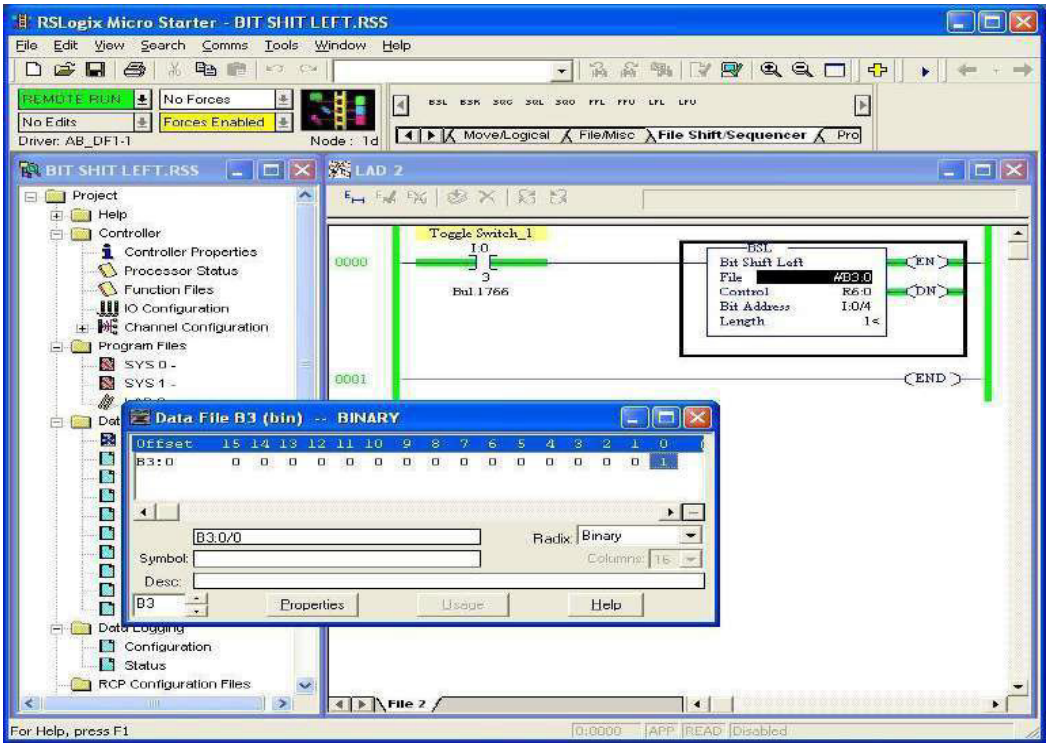
- Click on B3:0 Binary Option is shown below. After click on B3:0 Binary this window will open.



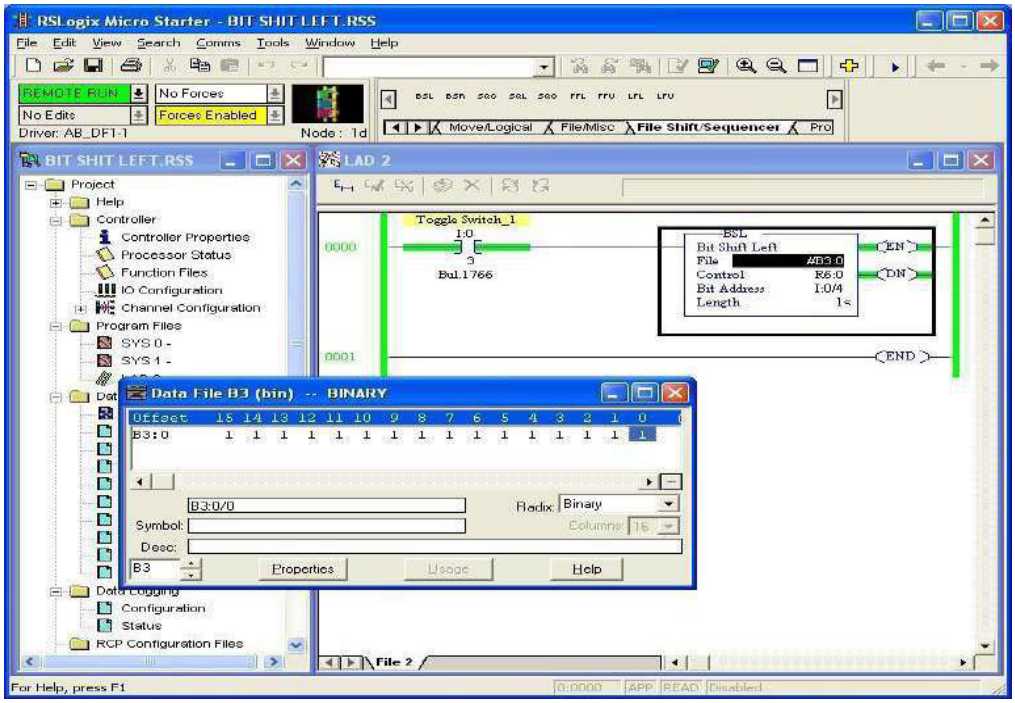
- You enable the I:0/4(ToggleSwitch_2) is shown below.



- When you enable the Toggle switch, B3:0 bit position is shown below.



- When you Enable the I:0/5many times, B3:0 Bit position Shift Left is shown below.
- When You Enable and Disable the I:0/3 (Toggle Switch_1)bit will be shift left



CONCLUSION:

Thus the data manipulation operation was performed and verified successfully using PLC software**.**

Experiment No: 06

CONVEYOR CONTROL SYSTEM

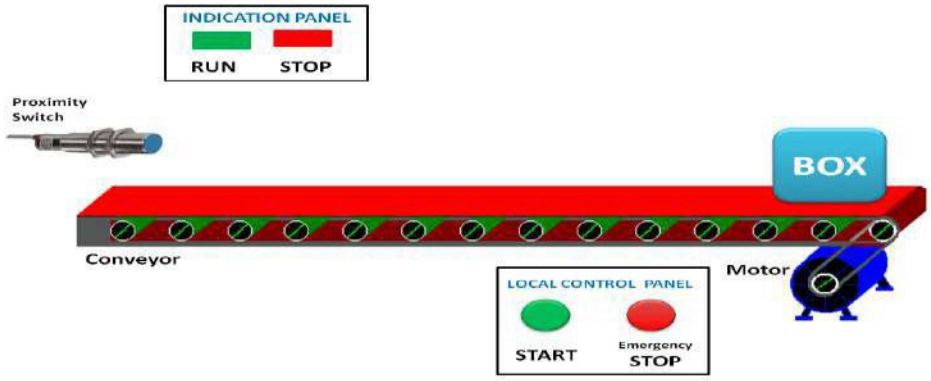
AIM OF THE EXPERIMENT:

To study about conveyor control system using PLC

APPARATUS REQUIRED:

Sl. no.	Apparatus Name
1.	VPAT-24 kit
2.	VPLCT-03 kit
3.	PLC software installed PC
4.	STEPPER MOTOR
5.	Patch chords
6.	Power chord

THEORY:



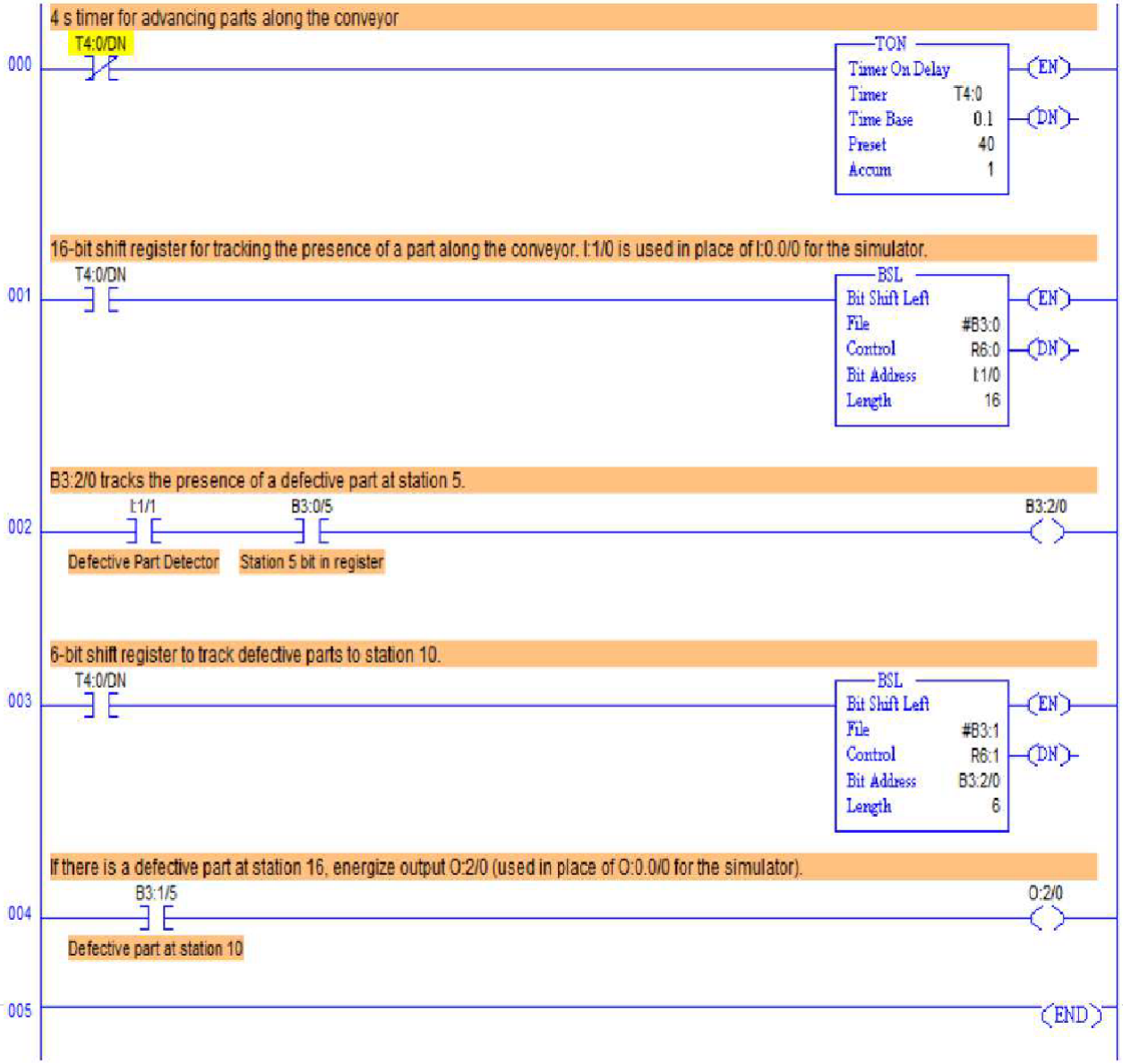
Conveyor System:

- ✓ A conveyor system is a common piece of mechanical handling equipment that moves materials from one location to another.
- ✓ Conveyors are especially useful in applications involving the transportation of heavy or bulky materials.
- ✓ Conveyor system allows quick and efficient transportation for a wide variety of materials, which makes then very popular in the material handling and packaging industries.

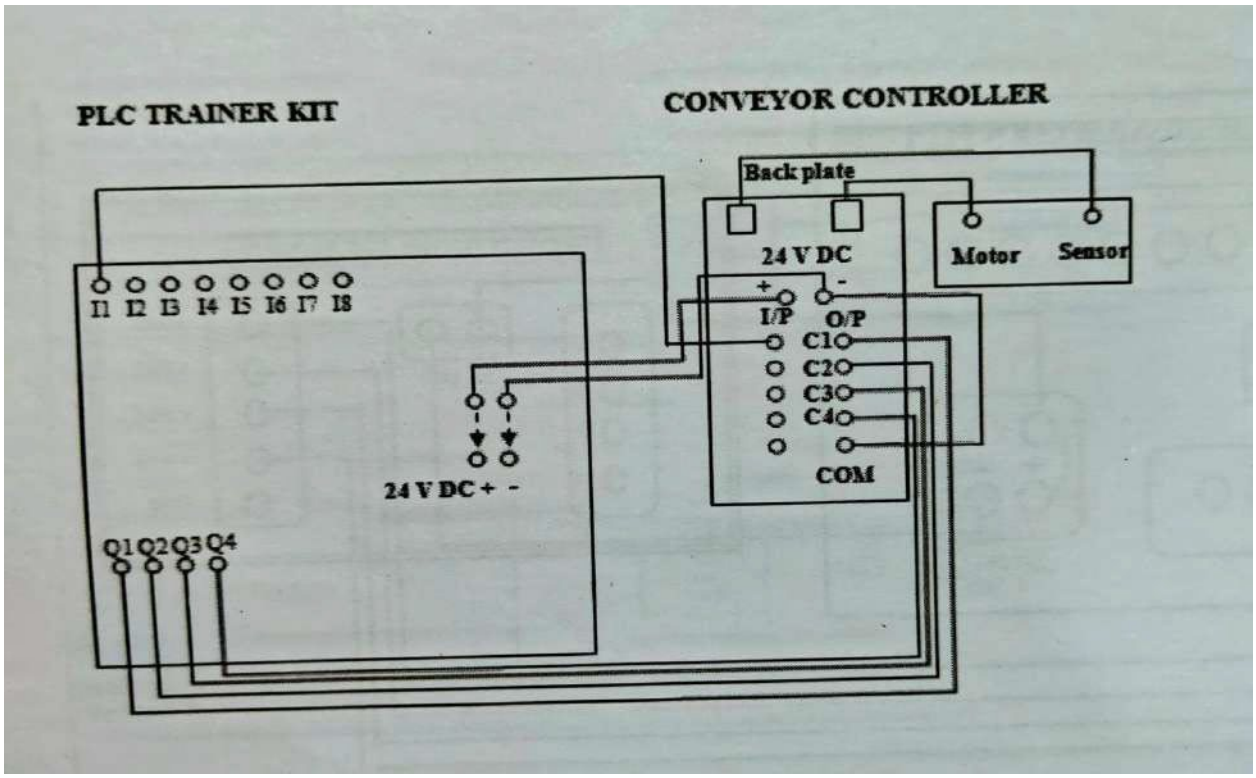
Stages in the System:

1. **Comparator**: - Two inputs (Threshold & Feedback); compares the output of detecting network and a preset.
2. **Converter/Switching circuit**: - Controller; takes decision based on the comparator output.
3. **Steeper Motor**: - A stepper motor is a brushless, synchronous electric motor that converts digital pulses into mechanical movement.
4. **Conveyor belt**: - Object translational mechanism.
5. **Object presence detector**: - To give feedback based on presence of object at the end position.

LADDER LOGIC:-



WIRING DIAGRAM:



PROCEDURE:

1. Open the PLC Software and design the ladder diagram.
2. Interface the PLC with the system using Ethernet cable.
3. Make electrical connections as per wiring diagram.
4. Download the program and run it.
5. Place the object on the conveyor. If the sensor the object, conveyor stops for 2 seconds and again starts automatically.

CONCLUSION:

Thus, the study of conveyor control system had been studied.

Experiment No: 07

ON-OFF THE DC MOTOR

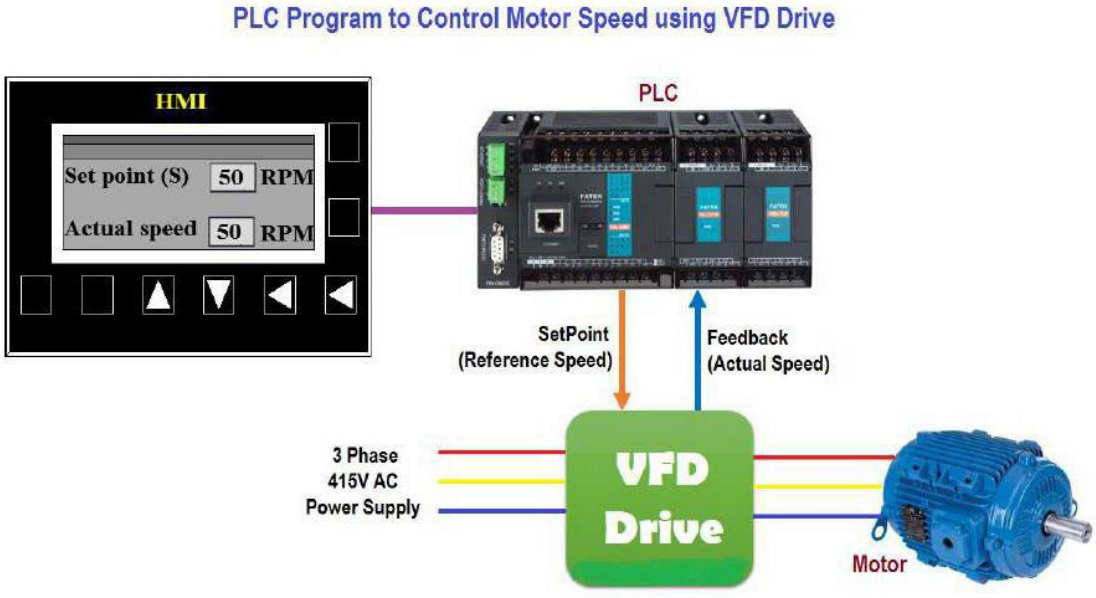
AIM OF THE EXPERIMENT:

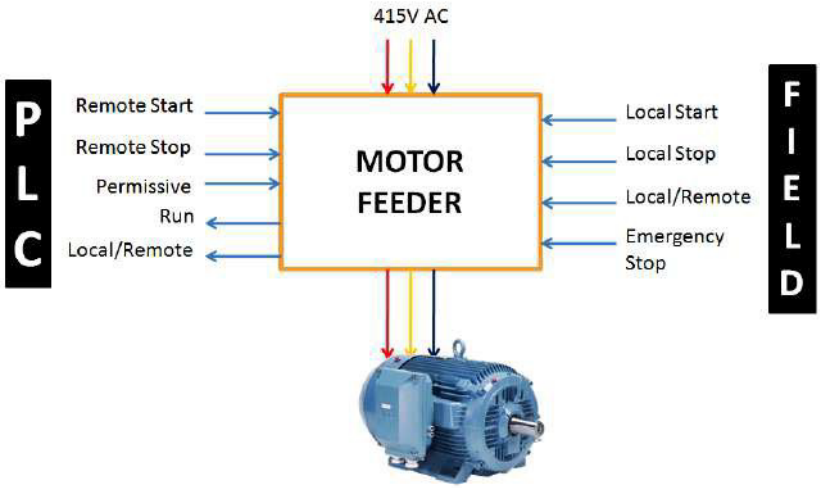
Write and implement ladder logic program to on-off the DC motor using PLC.

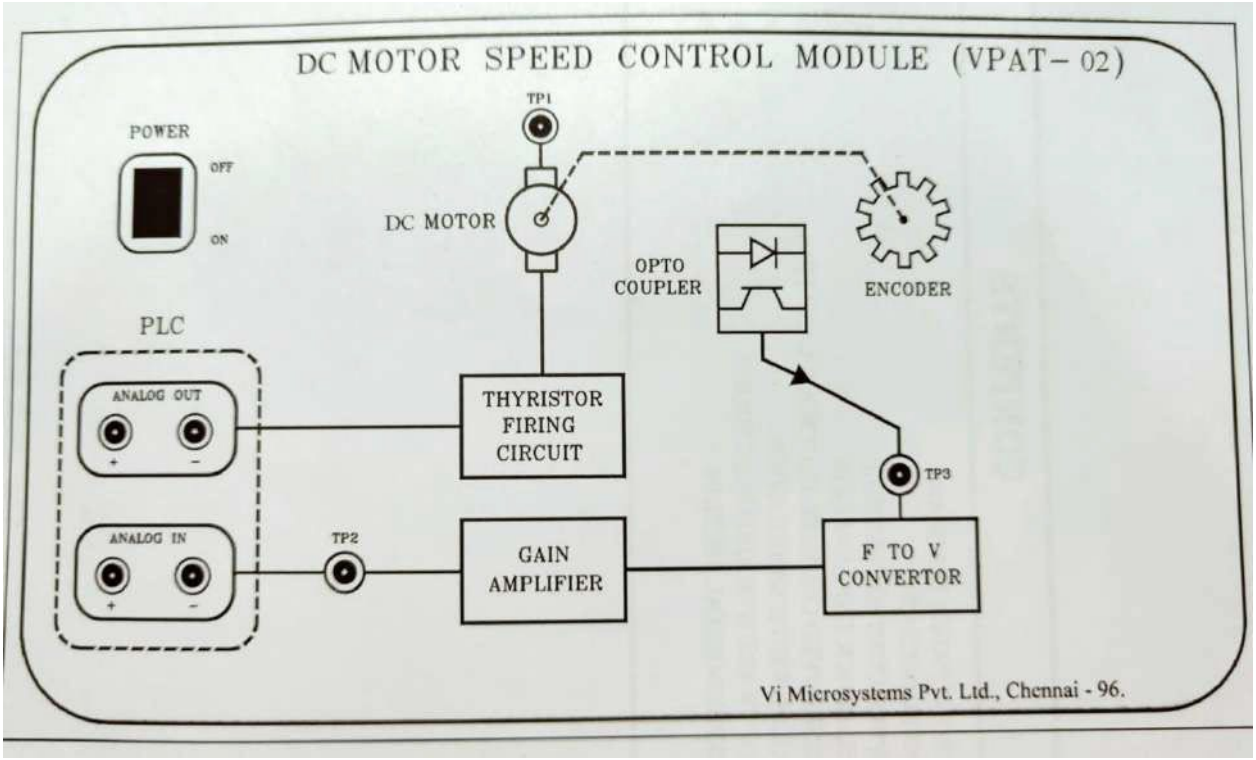
APPARATUS REQUIRED:

Sl. no.	Apparatus Name
1.	Speed control module trainer (VPAT-02)
2.	PLC trainer kit
3.	Personal computer installed with TIA portal software
4.	DC Motor
5.	Patch chords

THEORY:



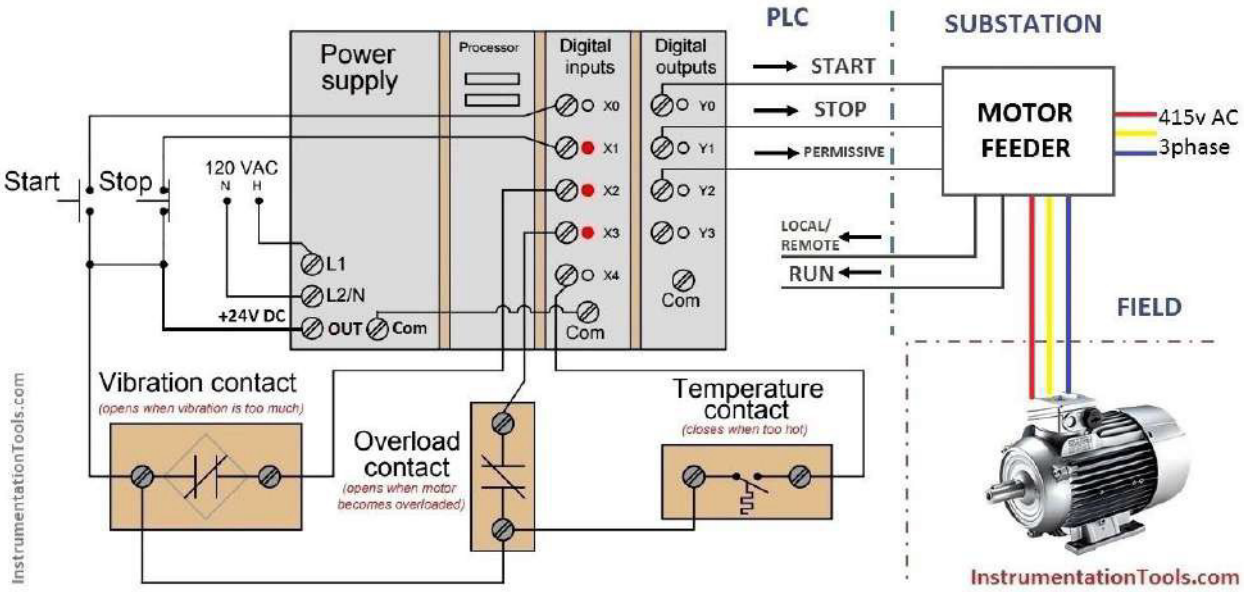




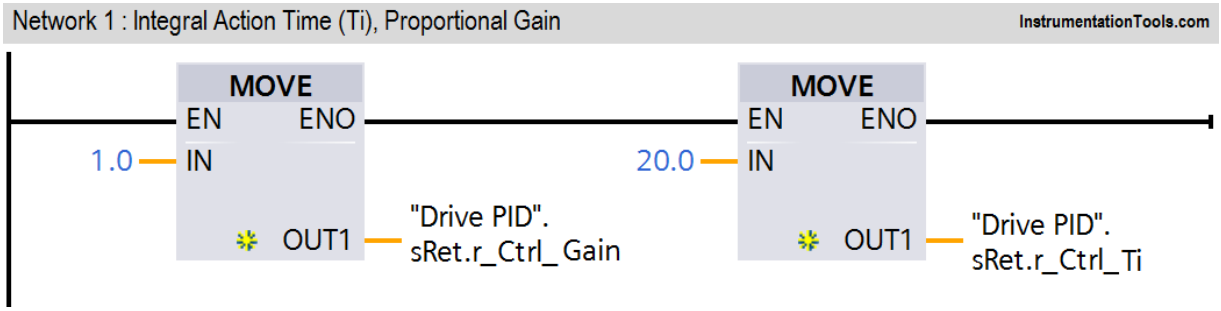
Speed Control Module:

- This module maintains the speed of DC motor to the set point using PLC.
- After giving the set point (SP) the PLC produces the analog output, this will be given to motor.
- So initially the motor starts rotating at some speed.
- The speed of the motor is measured using Opto coupler sensor.

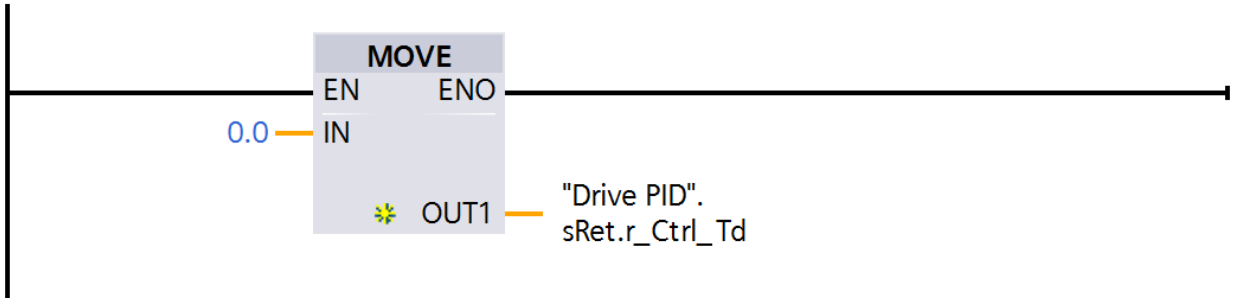
- The output of Opto coupler will be a series of pulses are converted into voltage using frequency to voltage converter.
- This voltage is process variable (PV) and is applied to the analog input of PLC.
- In PLC program PID block reads this PV and compares the both set point (SP) and process variable (PV), it creates error value and produces the control variable (CV) to the motor unit through analog output.
- This controlled output will maintain the speed of AC motor at its set point (SP).
- Likewise it forms the closed loop control using PLC.



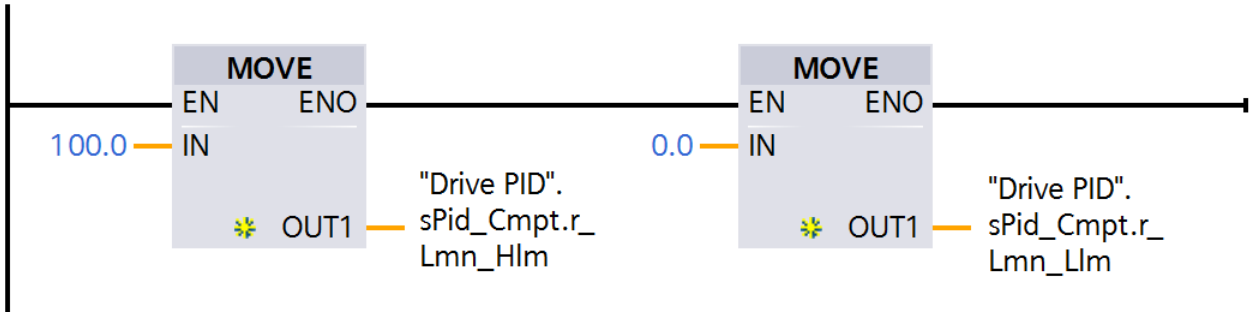
PLC Ladder diagram to control Motor Speed



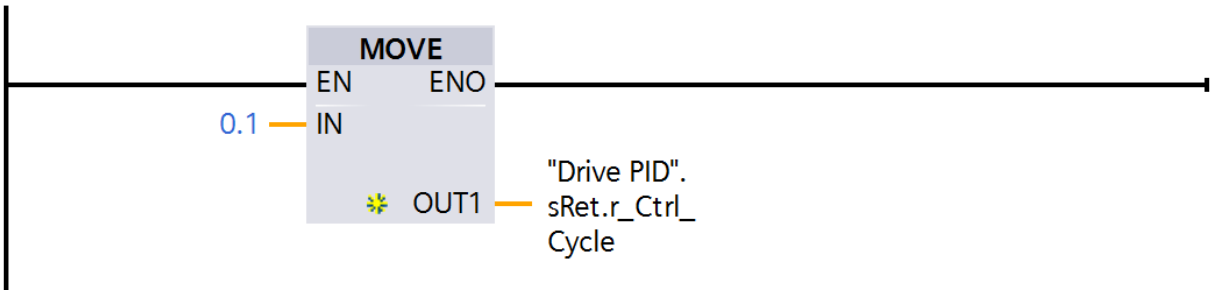
Network 2 : Derivative Action Time (Td) InstrumentationTools.com



Network 3 : Max and Min Output Limit InstrumentationTools.com

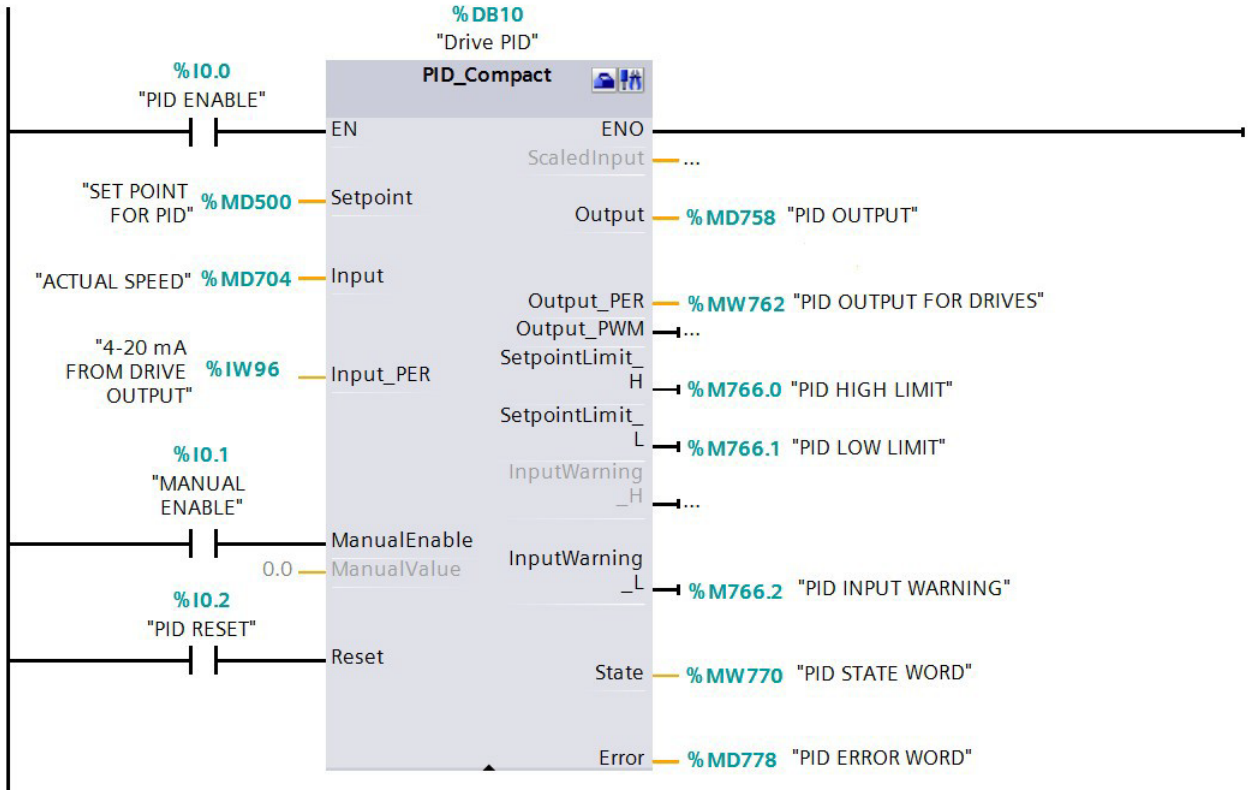


Network 4 : PID Sampling Time InstrumentationTools.com



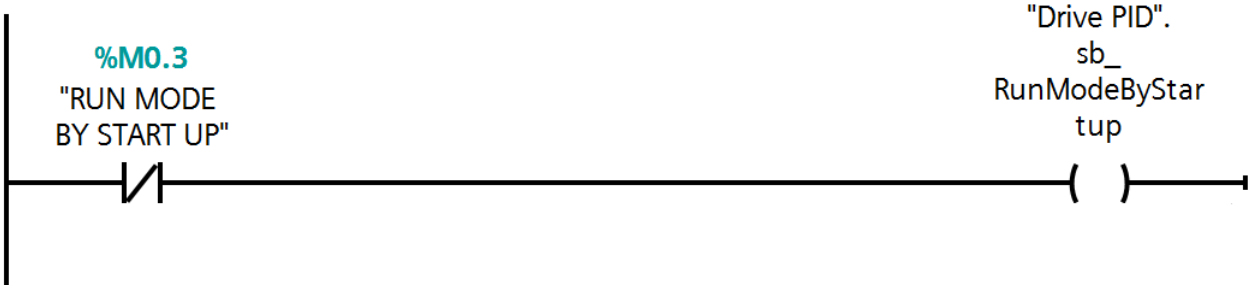
Network 5 : PID Block (Linear PID for Drives Output

InstrumentationTools.com



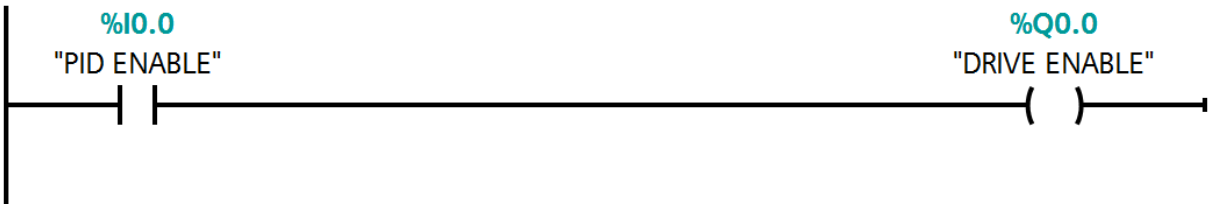
Network 6 : Enable Running Last State Re-Start or Start-up

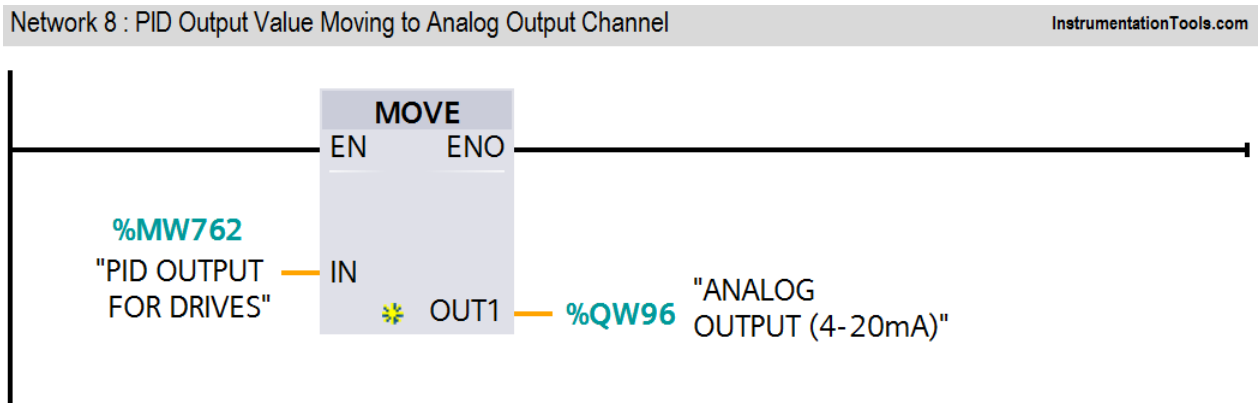
InstrumentationTools.com



Network 7 : PID Output for VFD Drive

InstrumentationTools.com

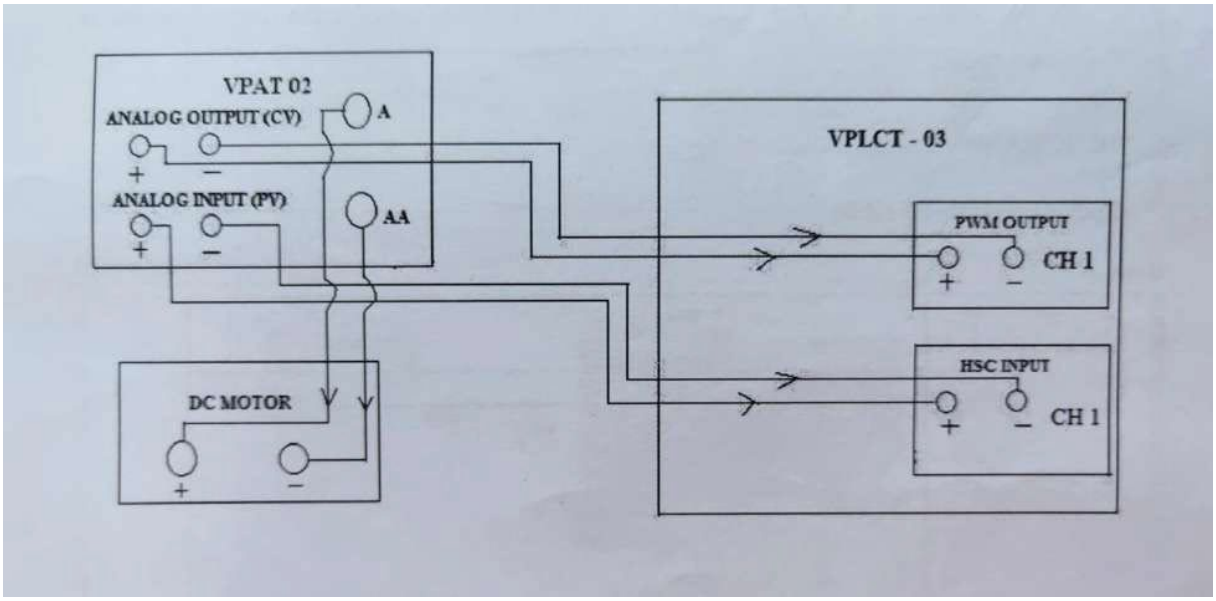




Runtime Test Cases:-

Inputs	Outputs	Physical Elements
I0.0=1	Q0.0=1	Drive ON
MD500=50RPM	MW762=13838 (approx.)	Drive speed =50RPM

WIRING DIAGRAM:



PROCEDURE:

1. Made the connections based on wiring diagram.
2. Switch ON the PLC kit.
3. Interface the PC to PLC through Ethernet Communication Cable.
4. For configuration of software, programming follows the procedure.
5. Switch ON the DC MOTOR SPEED CONTROL kit.
6. Download and run the program to PLC.
7. Set the set point of motor speed (0-1500 rpm) i.e. 0.0 to 100.0 in PID set point block.
8. Based on set point the PID controller enables the PWM output pulses.
9. By this pulse the DC Motor starts rotating at some speed.
10. Opto coupler sensor senses the feedback of high speed ON pulses.
11. That feedback we can read from High speed counter input channel.
12. Now the PID controller compares the set point (SP) and Process (PV) (SPEED).
13. By varying the Control Output (CV) (PWM pulse) it makes the motor running at set speed.

RESULT:

Thus, the ON-OFF operation of DC motor using PLC was studied successfully.

Experiment No: 08

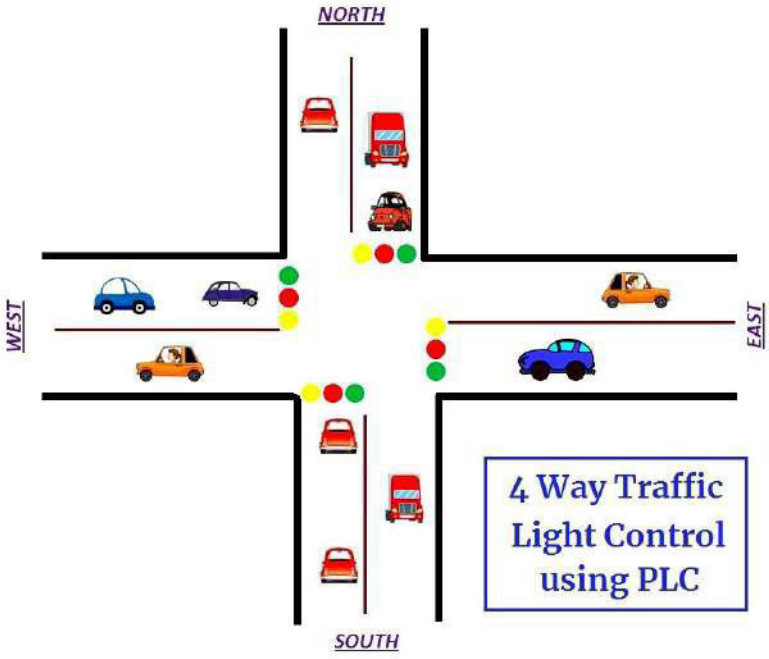
TRAFFIC LIGHT CONTROLLER

AIM OF THE EXPERIMENT:

To study the Traffic light controller system by using PLC.

APPARATUS REQUIRED:

Sl. no.	Apparatus Name
1.	VPAT-03 kit
2.	PLC
3.	PLC software installed PC
4.	ETHERNET cable
5.	Patch chords
6.	Power chord



THEORY

Density Based Traffic Light Controller System:

- Traffic control system is used to control automatically the vehicle and human beings by using simple program logic.

- Microcontroller system is very complex to programming and wiring. Hence PLC is used to control the traffic.
- For the complex road maps the PLC provides simpler solution by means of ladder programs.
- It is four directions based traffic light controller system, which is depends upon the density of vehicle.
- If an one direction's density of vehicle is HIGH, but another one direction's density of vehicle is very LOW, then we will give more time duration to high density direction half of time duration to middle density direction, the one third (1/3) of time duration to very low density direction as respective as our decision.
- If one direction is in glow green LED [Run-mode], other directions are in red- LED [Stop-mode] .

List of Inputs and Outputs

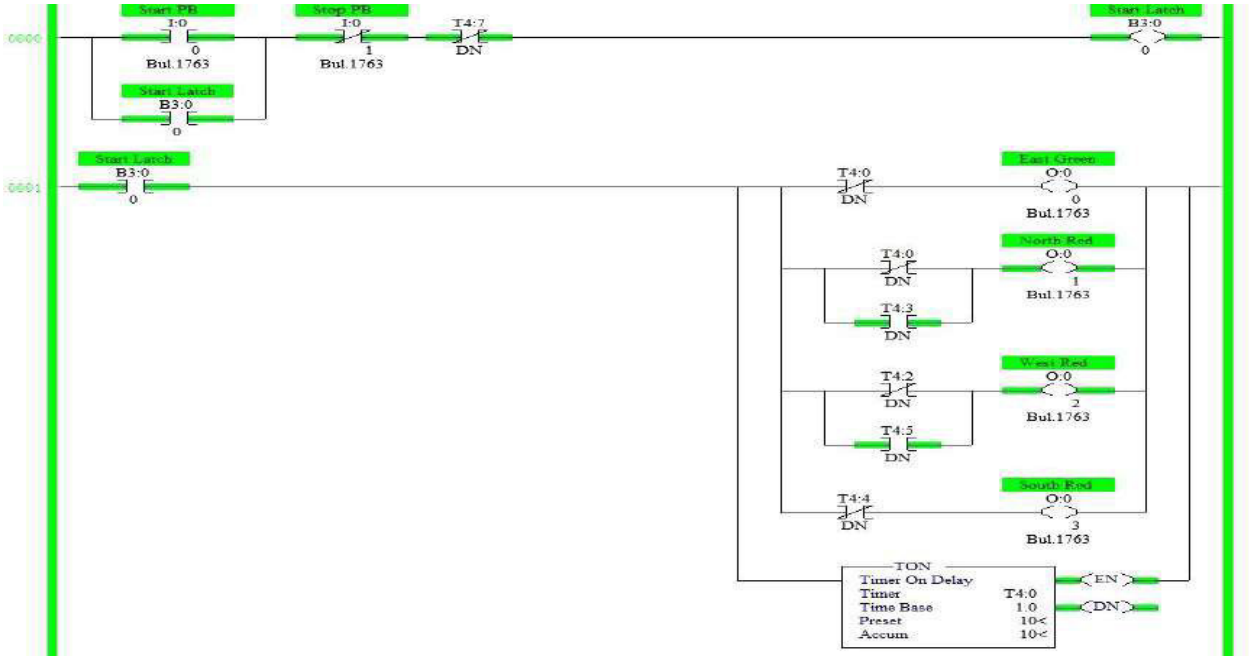
Sl. no	Address	Name	Input/ Output
1	I:0/0	Start	Input
2	I:0/1	Stop	Input
3	B3.0	Memory	Memory
4	O:0/0	East Green	Output
5	O:0/1	North Red	Output
6	O:0/2	West Red	Output
7	O:0/3	South Yellow	Output
8	O:0/4	East Yellow	Output
9	O:0/5	North Yellow	Output
10	O:0/6	North Green	Output
11	O:0/7	East Red	Output
12	O:0/8	West Yellow	Output
13	O:0/9	West Green	Output
14	O:0/10	South Yellow	Output
15	O:0/11	South Green	Output

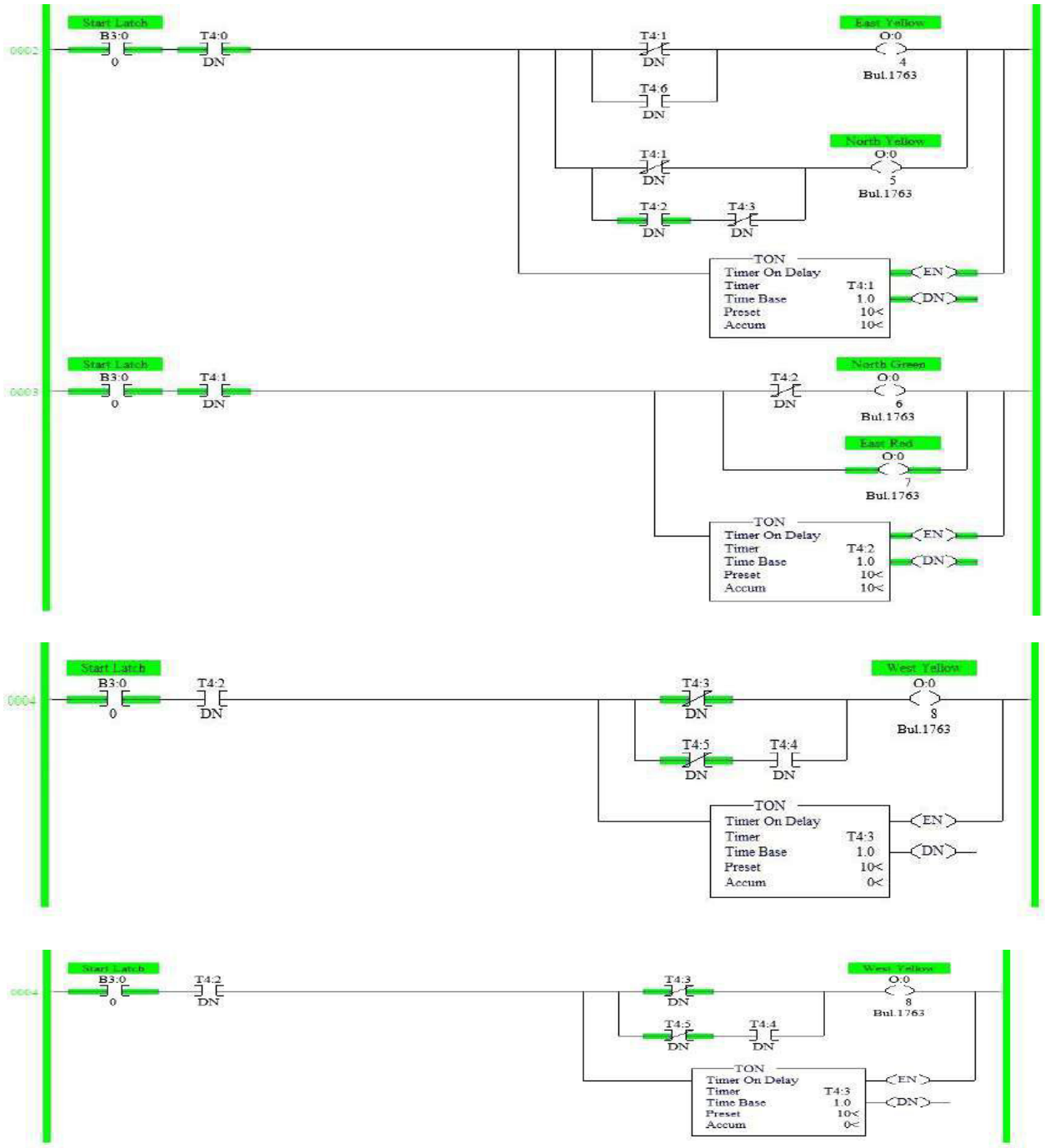
Sequence of Operation

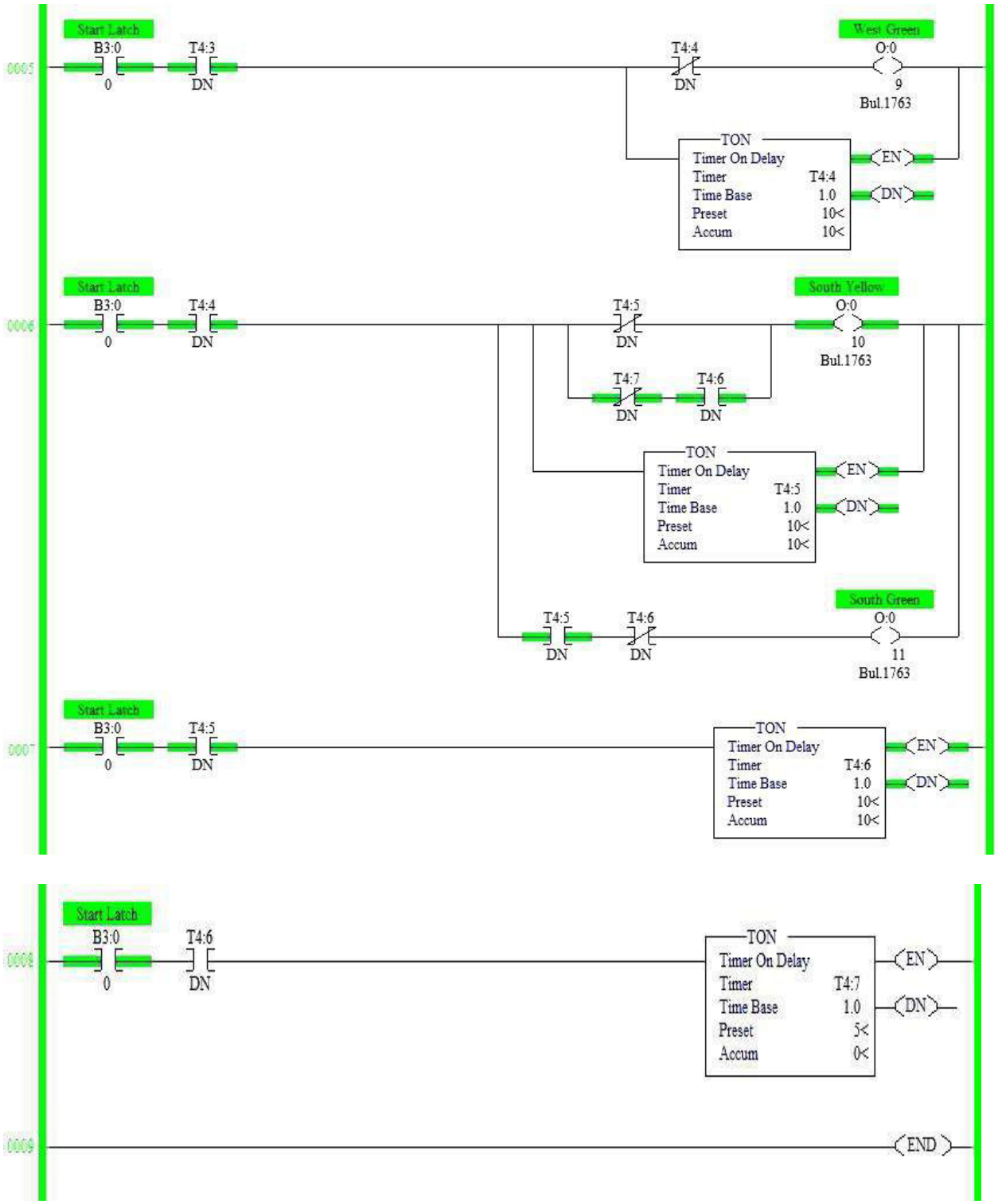
Below tabular column gives the Steps or sequence of outputs to turn ON the traffic system lamps (RED, GREEN, YELLOW)

S.NO	EAST	WEST	NORTH	SOUTH
1	G	R	R	R
2	Y	R	Y	R
3	R	R	G	R
4	R	Y	Y	R
5	R	G	R	R
6	R	Y	R	Y
7	R	R	R	G
8	Y	R	R	Y

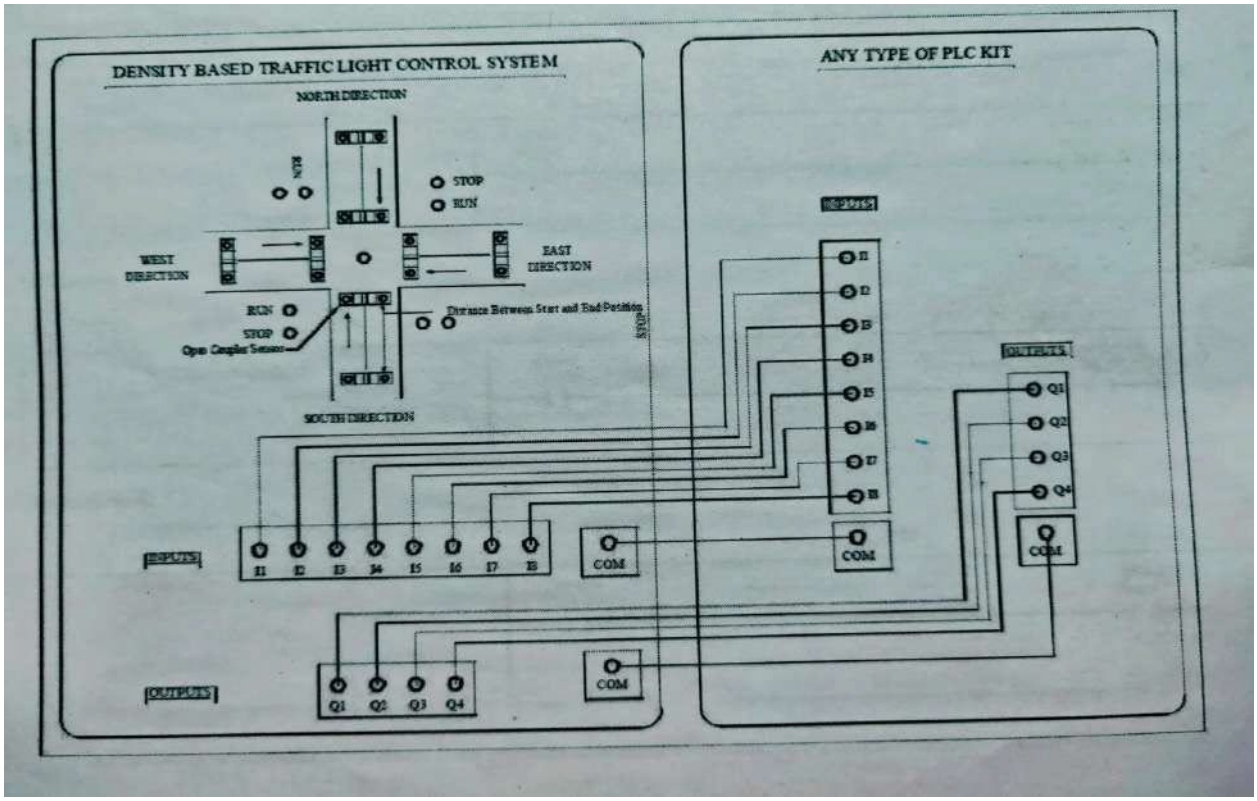
LADDER LOGIC PROGRAM:-




Start Latch
B3:0
0
T4:0
DN
T4:1
DN
T4:6
DN
East Yellow
O:0
4
Bul.1763
T4:1
DN
North Yellow
O:0
5
Bul.1763
T4:2
DN
T4:3
DN
TON
Timer On Delay
Timer T4:1
Time Base 1.0
Preset 10<
Accum 10<
EN
DN
Start Latch
B3:0
0
T4:1
DN
T4:2
DN
North Green
O:0
6
Bul.1763
East Red
O:0
7
Bul.1763
TON
Timer On Delay
Timer T4:2
Time Base 1.0
Preset 10<
Accum 10<
EN
DN
Start Latch
B3:0
0
T4:2
DN
T4:3
DN
T4:5
DN
T4:4
DN
West Yellow
O:0
8
Bul.1763
TON
Timer On Delay
Timer T4:3
Time Base 1.0
Preset 10<
Accum 0<
EN
DN
Start Latch
B3:0
0
T4:2
DN
T4:3
DN
T4:5
DN
T4:4
DN
West Yellow
O:0
8
Bul.1763
TON
Timer On Delay
Timer T4:3
Time Base 1.0
Preset 10<
Accum 0<
EN
DN


Start Latch
B3:0
0
T4:3
DN
T4:4
DN
West Green
O:0
9
Bul.1763
TON
Timer On Delay
Timer T4:4
Time Base 1.0
Preset 10<
Accum 10<
EN
DN
Start Latch
B3:0
0
T4:4
DN
T4:5
DN
T4:7
DN
T4:6
DN
South Yellow
O:0
10
Bul.1763
TON
Timer On Delay
Timer T4:5
Time Base 1.0
Preset 10<
Accum 10<
EN
DN
T4:5
DN
T4:6
DN
South Green
O:0
11
Bul.1763
Start Latch
B3:0
0
T4:5
DN
TON
Timer On Delay
Timer T4:6
Time Base 1.0
Preset 10<
Accum 10<
EN
DN
Start Latch
B3:0
0
T4:6
DN
TON
Timer On Delay
Timer T4:7
Time Base 1.0
Preset 5<
Accum 0<
EN
DN
END

WIRING DIAGRAM:



PROCEDURE:

1. By applying start one triggering pulse to input I1.0(I9), which causes for open coilM0.0 energized in Network 1, Due to energization of open coil M0.0 initiates the operation.
2. It causes the open coil Q0.0 [Q1 coil] energized in Network 5 on delay timer is also energized. After completing the time delay [5sec], which is depending upon preset Time coil M0.0 will be energized. Due to energization of M0.1 it cuts the Q0.0 coil output and enables the Q0.1 west Direction output.
3. Due to Q0.1 (Q2 coil) will energized in Network 7, then Q1 coil Q0.0 de-energized and timer is reset in Network 8. This same process was repeated again in west direction, North Direction, and East direction.
4. In west direction, that means Network 8 ON delay timer will be energized depends upon Q2coil [Q0.1] then M0.2 [open coil] is energized, after completed programmed time delay, which is depends upon preset time, then rest the timer. At that time, Q2 coil [Q0.1] de- energized, timer was reset, Q3 coil [Q0.2] was energized.
5. In network I0, latch connection Q3 coil [Q0.2] is used to energized for continuously. Hence on delay timer are starts ON due to Q3 contact [Q0.2] energization in run- 8, then M0.3 [Q0.2] will energize. After completion of time delay, which is depends upon preset time, then reset the timer. Then Q4 coil [Q0.3] energized and Q3 coil [Q0.2] are de-energized at that same time.

6. In Network I3 ON delay timer is starts ON, due to contact [Q0.3] energization, then M0.4 coil will be energized, after completion of time delay, rest the timer this time delay depends upon the [M0.4] preset time.

CONCLUSION:

Thus the study of traffic control system was done by using PLC.