



University of Global Village (UGV), Barishal

Dept. of Electrical and Electronic Engineering (EEE)

Lab Manual PLC & Microcontroller

Noor Md Shahriar BSc in EEE, <u>RUET</u> Senior Lecturer Co-chairman, Dept. of EEE University of Global Village (UGV) 874/322, C&B Road, Barishal, Bangladesh. Socontact: <u>+8801743500587</u> Facebook | Im 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 Rationale	3
Course Objectives	3
Assessment Pattern	3
Course Learning Outcome (CLO)	4
Course Outline	4
Course Schedule	4
LIST OF EXPERIMENTS	6
Lab Precaution	7
Experiment no: 01	8
Experiment No: 02	12
Experiment No: 03	20
Experiment No: 04	24
Experiment No: 05	34
Experiment No: 06	50
Experiment No: 07	53
Experiment No: 08	60

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

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.	Tania & Dataila	Class	CLO
No.	Topic & Details	Hours	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	Δ	CLO1,
2	digital inputs and outputs.	4	CLO2
2	Timers in PLC: On-delay, off-delay, and retentive timers, their configuration, and	6	CLO2,
5	implementation.	0	CLO3
4	Counters in PLC: UP and DOWN counters, programming methods, and practical	1	CLO2,
4	applications.	4	CLO3
5	Advanced Functions in PLC: MOVE, control statements, math functions, and data	1	CLO3,
5	manipulation techniques.	4	CLO4
6	Applications: Conveyor system control, traffic light controller, and motor control	6	CLO3,
0	using PLC.	0	CLO4
7	SCADA Integration with PLC: Interface, tag association, and real-time operation	1	CLO4,
/	using graphical symbols.	4	CLO5

Course Schedule

Class No.	Торіс	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

Exp No. NAME OF EXPERIMENTS 1 Introduction to PLC trainer & its installation with PC Write and implement a simple ladder logic program using digital inputs and outputs 2 for PLC. Write and implementation a simple ladder logic program using timer. 1) On delay timer 3 2) Off delay timer 3) Retentive timer Write and implementation a simple ladder logic program using counter. 1) UP counter 4 2) Down counter Write program on MOVE, control statement, math function, data manipulation 5 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. Interface SCADA with PLC and associate tags with memory and I/ O and operate 9 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.

LIST OF EXPERIMENTS

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".
- 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.
- 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.
- 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:

- 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 $(\dashv \vdash)$.
- 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-LOW ; 1-HIGH



PROCEDURE:

- 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
2.	software
3.	Ethernet cable
4.	Patch chords

<u>THEORY</u>:

- 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:

Α	В	С
0	0	0
0	1	1



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.
- 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

<u>THEORY</u>:

- 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:

Α	В	С
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

<u>AIM OF EXPERIMENT</u>:

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

<u>THEORY</u>:

- 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.

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

Programming:-



TRUTH TABLE & SYMBOL OF EXNOR GATE:

Α	В	С
0	0	1
0	1	0

	1	0	0	
	1	1	1	
L	0-	LOW 1-HIG	H	
	_			
1	Z_₽	~ \	$r = \overline{\Delta G}$) R
	в)))¢-		
	- – Ž	~		

XNOR

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.
- 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-NOR_GATE operation was studied successfully through PLC Software.

Experiment No: 03

Write and implementation of simple ladder logic program using timer

<mark>ON DELAY TIMER</mark>

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.

<mark>OFF DELAY TIMER</mark>

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.

🕒 🖬 Seve project 🚔 🗶 🚈 🗔 🗙 🍤 1 (**1	t. 田田里間 🌶 Goodine & Gruther 战 西语 🛪 🖃 🔟 (Starthurp)	iette 🖌	rouny meganes	PORTA
	program1 + PLC_1 [CPU 1214C AC/DC/RUy] + Program blocks + Main [OU1]	_#=×	Inducations	
Devices			Options	
9	공장공은 씨는 E 프 프 U 원 : 월 : 드 딸 안 당 방 왕 및 날 날	1988 879 8 3	251 257	 Image: Image: Image
	Main		> Fauntites	
1 program 1	Name Data type Default value Comment		an Bart to the strength	
Add new device	1 d T hout		· Basic instructions	and a start of the
Devices & retworks	7 - 0 * mittel Cell Bool mittel cell at	the Off	tiame	Description -
PLC_1 [CPU 1214C AC/DORb]	3 4 Remanence Bool -True, if mm	anent data are available	Contention of the second secon	
Device configuration			· · ··································	Windows print the
3 Online & diagnostics	<u>+⊢</u> +⊩-+⊢		100 °10'	theorem and your
 Trogism blocks 	and the state of t		Normally open	n contact (Shi
Add new block	 Block BBe: "Mein mogram Sweep (Cjcle)" 		di -proste	Mich No.
- Main (083)	compart		BICK.	Nameta ann
I = Technology objects	 Network 1: 		10 - on	Balant av The
• og tidemal source files			1 6 5	Cat auteur
+ 🕞 FLC tags	editories).		AD SET OF	Sethefeld
Eg PLC data types	1		HI LESST AS	Berlay hit fa
 Watch and force tables 			WP cp	Satisfactat B
A Online Beckups			and the second s	Second and S
• 📷 Traces			Gel and	Senningen
 B, Device proxy data 			-0	Scan opera
III fregraminie	 Network 2: 		100-00-	Set operan
PLC alarm text lists	Comment		ME -01)-	Setoperan
• 🔄 Local modules			ST F TRO	StanBOR
• Tr Lingrouped devices			BT N THE	Scention
• al Common deta			PT & THE	Detect pot
Documentation settings			APP + TTAC	Detert ned
 Dingueges & resources 			Timer one rations	
Details view			• Sti Counter operations	
1.000			+ (2) Comparator operations	
			+ TE Meth functions	
			Nove operations	
Name Address			C 11	
			> Extended instructions	
			Contract of the second	
		Look (2)	7 Technology	
		Discontine (5) Into (1) Discontine	Communication	
		Stropenses Stince D & Diagnostics	 Optional packages 	
Portal view El container	- Man (more)		the biolocs biodium (was sovid	succe

2.

	minet tree	programt + R.C. 1 (CRI 1214C	AC/DCIRINE > 1	Program blocks + M	ata 10811		a a x	last contines	
Uptors Uptors Uptors Image: Image		program v Prc_1 jevo 1214c	Adduniy) F	rogiam olocits in m				A CONTRACTOR OF CONTRACTOR	11.11
Comments	Devices					112		Options	
Wein > Execution > Execution Program > Execution > Execution Peription Program > Execution > Execution > Execution Program > Execution > Execution > Execution Program > Execution > Execution > Execution > Execution Program > Execution > Execution > Execution > Execution > Execution Program > Execution > Execu	S 📃 🖬	N N 2 2 2 1 E E E	· · · · · · · · · · · · · · · · · · ·	目間 (* 6) 住 8	A to P P P O O S	i ula	8	64 64	
Import Inters Data syste Default subs Comment Import		Main						> Favorites 2	
Market Implified Implified <tdimplified< td=""> <tdimplified< td=""></tdimplified<></tdimplified<>	program1	Nome	Data type	Default value	Comment			✓ Basic instructions	
Bourde & Ardexon H B	Add new device	1 🔄 🕶 Input		1	1000000		^	Name	Description
Impresentations Impresentations Impresentations Impresentations Impresentations Impresentations Impresentations Impresentations Impresen	Bevices & networks	2 40 • Initia Call	Bool		Initial call of this OB			🖡 🛅 General	
Image: Design State of Sta	PLC_1 [CPU 1214C ADDORN]	a 🗨 · Remanence	500		-True, if remanent data are availab	pe	×	🕨 🥃 Bit logic operations	
Book tribes of generation Add ere block Main (931) System Blocks Main (931) System Blocks Main (931) Motivark 1: Comments Motivark 1: Comments Math and bloc tables Software of the software Math and bloc tables Motivark 2: Metwork 2: Metw	A Online & diagnostics							Timer operation:	
Add men block Weide Program Subsets Comment Program Subsets Comment Subsets Comment Subsets Comment Subsets Comment Subsets South and block tables Device provides Program info Program info Program info Program info Proces minification subtrags Commend Subsets Commend S	* Prostam blocks	ar ar ar 🖂 🗉 –						 Counter operations 	
Advisor Addiess Addies Addiess Addies Addiess Addies A	Add new block	 Block title: "Main Program Sweet 	ep (Cycle)"					5 CU	Countup
System blocks Tachnology Abjects Tachnology Tachnology	🐠 Main (081)	Comment						* CD 2	Count down
Technology-objects Formard Fo	System blocks	Notwork 1							count up at
Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files Image: Section all source files I	🖡 🌆 Technology objects							Comparator operations	
NetCodes types SetCodes types	 External source files 	Corninera						Move operations	
PLC deta type: Control to be tables Control tables Control to be tables Control tables Contables Control tables Contables Control tables Co	PLC tags	T						Conversion operations	
Watch and totes tables > Watch and totes tables Io Onive Bookups > Site and rots re Provide provide for total > Site and rots re Provide provide for total > Site and rots re Provide provide for total > Site and rots re Provide provide for total > Site and rots re Provide provide for total > Site and rots re Provide provide for total > Site and rots re Provide for total > Site and rots re <td>PLC data type:</td> <td></td> <td></td> <td></td> <td>12</td> <td></td> <td></td> <td>Program control operati</td> <td></td>	PLC data type:				12			Program control operati	
Commendation settings Commendation Commendat	Watch and force tables							Word logic operations	
Image: Program info Image: Program info If Colume rest lots Image: Program info If Colume rest lots Image: Program info If Colume rest lots Image: Program info Image: Program info Image: Program info <td>Record</td> <td>le le</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>) 🙀 Shift and rotate</td> <td></td>	Record	le le) 🙀 Shift and rotate	
Image: Program info Image: Program info Image: Program info Image: Program info <td>Device providata</td> <td>Natural 2</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td>	Device providata	Natural 2							
E PLC a larm text lot:) E Local module: Die Unsport divide:) December divide: Die Commendate) December divide: D December divide: >	Program info	P ACTION 2.							
Image: Second module: Image: Second module: <td< td=""><td>E PLC alerm text lists</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></td<>	E PLC alerm text lists								
Name Address Intermediation setting: v Details view v Intermediation setting: v Intermediation setting: v Details view v Intermediation setting: v Intermediation setting: </td <td>🕨 🌆 Local modules</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td>	🕨 🌆 Local modules								
Address Address Address C Communicate C In C Communicate C In	+ 🔚 Ungrouped devices								
Image: Comparision settings Image: Comparision settings Image: Comparision settings Image: Comparision settings Name Address Image: Comparision settings Image: Comparision settings Image: Comparision settings Image: Comparision s	F 🙀 Commen data								
Details view Name Address 100% > Extended instructions > Technology > Communication @Properties * Units @ Properties * Units * Optional packages	F Documentation settings								
Name Address III Name Address > Extended instructions Image: Image	Details view	12							
Name Address	***************************************	1							
Name Address		1							
Name Address <td< td=""><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></td<>									
Extended Instructions Technology Techn	Name Address	1						< 11	
Communication Communication Communication Communication Communication								> Extended instructions	
Communication Properties Uning & Diagnostics Properties								> Technology	
Q Properties Unifo () Q Diagnostics = > > Optional packages						100%		> Communication	
		-			Q Properties	Linfo Diagnostics	1 1	> Optional packages	

3.



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

	III (program1 + PLC_1 [CPU 121	14C AC/DC/Rly] > Program blocks >	Main [0B1]	_ 7 = X	Instructions	# E
Devices					Options	
<u> </u>	이라 백백 동소 (1988)	= 🗩 🛿 : 🚔 : 🖃 😥 🥙 📞 d	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	8	tes tes	
	Main				> Favorites	
r 🔄 program t	nome Name	Data type Default valu	ue Comment		✓ Basic instructions	ALCONO.
Add new device	1 📶 🕶 input			^	Name	Description
Devices & networks	2 📲 🖲 Initial_Cal	800	initial call of this OB		Seperal	Termines.
FLC_1 (CPU 1214C ACIDCIRIN)	3 🔄 🕷 Remomence	Bool	-True, if remanent data are available	~	Bit logic operation	
Device configuration			active manual		• Tel Timer operations	
Coline & diagnostics		*			- T Counter operations	
 grogram blocks 	- Riock title: "Lais Program S	weer (Cole)*			# CTU	Countep
Add new block	Company Contractions	and falses			- CTD	Count dow
A Main [OB1]						Countupa
 System blocks 	ONetwork 1:				FC Comparator operations	
Figure Technology objects	Comment				Stath functions	
 (a) External source files 					Move operations	
PLC tags		081			• 😽 Conversion operations	
Contraction of the second s	7et_0	Counter_			+ at Program control operati.	49
gy watch and torce tables	0.	DB"			• Word logic operations	
Contrie backups	8	ERU			Shift and rotate	
• an inters		int				
Site Device proxy data	- cu	Q-				
The second secon	faise - g	CV				
Fictore in the second	STT2 - PV					
Lineman ad dayle ar						
Common data	the second se					
Commentation settions	h Notund 2:					
Contraction and the second sec	O P Network 2:					
Details view						
tiama galdiare						-
introe lineares					< 1	
					Extended instructions	Sales and
					> Technology	
			100%	I IIII	> Communication	A A A A A A A A A A A A A A A A A A A
			The second secon	second and the second sec	N Output and an always	

5. Before Enabling the Counter Block

of or at training and the second s	program1 + PLC 1 (CPU 1214C AC/DC/Riv) + Program blocks + Main (081)	< instruction	1.11
Devices		Optimes	Chartibeally
(m) et	(A X	the last	
			1. 25.5
	Mein	> Favorites	101120000000
_ program i	Name Datatype Datativitie Commont	✓ Basic instructions	
Add new Device	a v npot	Name	Description
I Deves a leavest		🔜 🕨 🛅 General	
	s with a transmitter and the source of the s	🖆 🕨 💽 Bit logic operations	
2) Online & disconstine		• (@ Timer operations	
Program birde		• 💽 Counter operations	
P std new block	▼ Block title: "Mein Program Sveep (Cpcle)"	🔹 CTU	Countup
- Main (071)	Constant	📽 CTD	Count down
 mi Szczemblocks 		S CUD	Count up an
Technology objects	 NetWork 1; 	Comparator operations	
• an External source files	Denner	1 Meth functions	
I Cartotep	1	Move operations	
E FLC data types	Sp01	Conversion operations	
Wetch and force tables	"EC Counter	M Program control operation	10
Goline backups		Viero logic operations	
e 🔄 Taces	1000 CR0 1000	The shirt and ratate	
Device proxy date			
20 Program info			
🖹 FLC # lerm text litts	MANNO		
Local modules			
Hungrouped devices			
Gammon dete			
Socumentation settings			
D. S. B. LEWIS			
Details view			
		-	
name Address		21 10	-
		C 1	_
		Extended instructions	and the
		> Technology	
	100% 💌 🦛	> Communication	
		and the second se	

6. After Enabling the Counter Block

		areasonant a much triting to		Marces + Maile (CHP1]			Tasking	
and a second	11.202	Collected and a service of the backwellow					Onthony	hida
evices	[100] mil.				Fr. 48 (1) (1971) (1)	1.1	options	
	1771 272	NA NO 21 CO NO DO		to be so the de parts in	A CI 67 [12] 101			
	The second second	Main					> CPU operator pa	anel
program 1	G 🔍 🔿	Name	Data type 1	pefault value comment			✓ Call environment	et.
a Add new device		i al • nput	121			2	No condition defined.	
Devices & networks		1 - initial_call	500	initial cas of this i	28			
1 PLC_1 [CPU 1214C ACIDC/My]		E -413 • Remanance	Boel	=7us, dismans:	t data are available	2		Change
IT Device configuration		second reaction of the second second		10401040				
Se Online & diagnostics	1.00		-					
The Program blocks	•	· Block title: "Main Program	n Sweep (Cycle) ⁴					
Add new block		Comment						
Term Tog 1								
The system blocks		 Network 1: 					> Breakpoints	
 Technology objects 		C.D. BURGER					M Call biography	
• He External source ster		har built for					A local and the tar story	
FLC tags	•		1993					
• In FLC data types			D.					
 Us watch and force tables 		TIEC	Country					
Collee backupt			0.08*					
• Carnet		100.0	CRU	300.0				
Device proxy data		7 Tett. 1"	leat	Ten J'				
Mg Pingram Info		1 hourselow	0					
FLC alarm test litts			4					
Local modules			0					
La Ungrouped devices	117412	740.1	- Meron					
Common data		192×	EV Had at				1104104917	
E Documentation settings		i processi					No call pr	eth avoidable
and a second sec	×	2-112					1000000000	
etails view	and the second strains							
		1.1						
		and the second second						
		THE CHIEFE BIT						
ime Address								
					100%	Let an		
		-			dia la	alala and an and a second	1	
					A R. Description of the second s	The Contraction dataset		

7. After Resetting the Counter Block



CONCLUSION:

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

<mark>DOWN COUNTER</mark>

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.

	program1 + PLC_1 [CPU 121	AC ACIDCIRUY Pro	gram blocks + M	ain [OB1]		TEX In		
Devices						Or	otions	
	* 332234 553		10 60 6. 12 52	9615688	- G.	31	962 ef	a 🗆
1.	Main					1	Emerilian	
1 program t	e lane	Data type	Default volue	Comment		Ľ.	P devolutions	
Add new device	- input	and the		Barrier and the second second		A	Basic instructions	Louise a
Devices & networks	2 - miliar call	Bool		mitial call of this Off		7484	me	Descripti
· B PLC 1 [ORU 1214C ACKORN]	3 -0 + Remanence	Rool		Thue, if remanent date are available	de .		General	
Device configuration							- Bitlogic operations	
S Online & disgnestics							MI -1-	Horniell
• S Program blocks	=						Normally ope	en contect
add new block	 Block title: "Main Program S 	weep (Cycle)"					H -hole	mann
A Main 10811	Cammetti						10-()-	Assigne
· · · · · · · · · · · · · · · · · · ·						_	0 -0-	Negati
Suit External Lourse Ries	· Network 1:					_	(0) -(7)	Reseto
PLC tags	Comment						0 -(\$)	Set out
P PLC data types	10 0						O TET_BF	Setbill
Si yatch and frees tables							KU NESET_BP	Resetb
Online berkum	-						图 58	Setirese
1 The Decent							肥売	Repetion
It. Desare most data							HI -19-	Scan op
Bit Program into	Watnessk 2					_	H -m-	Sceniop
Br. PLC alarm text lists	- HELMON Z.						P (P)	Set ope
Finit Local modules:	Comment						P(1)-(10)-	Set ope
Lingraperi dedres	1						ET P_THG	Scan RL
Common data							图 11.104	ScanRL
TO Decementation satisfast							图 R_196	Delect
The Local Part & second as							图 L TRIG	Detect
A conference .	*					•	e Imer operatum	
Details view						1	• Counter operations	
						1 K	Comparator operations	
						2	1 Neth functions	
1						1	Move operations	
lame Address						\$	н	
						5	Extended instructions	
						2	Technology	
					100%		Communication	
				Properties	Division (1) 2) Diamostics	1	Ontional nackages	

2.

	II 4 program1 + PLC_1 [CPU 121	4C AC/DC/Rly] > Program	blocks + Main [OB1]		■ ■ X Instructions	
Devices					Options	
	विको अ <i>अव क</i> े ∎ व ा		60 C. 28 Ce 35 Ce 1. 1. Ce 21	0.00	13	est est
	Main			- 91 (17)	A Envirter	2
T program1	Name	Data type	Default value Comment		in avointes	and a state of the
Add new device	1 🔄 🕶 input				· i basic instructio	2005
着 Devices & networks	2 📲 * Initial_Cal	800	initial call of this OB		Name	Detchpbo
FLC_1 (CPU 1214C AC/DC/RIy)	3 🔄 * Remonence	8cel	=True, if remanent data are a	ava ila ble	General	100
Device configuration			(care proved)		1 B Bit logic operat	2010
W Online & diagnostics		r:			 tal limer operation 	15
Frogram blocks	=				 Counter operat 	3605
Add new block	 Block title: "Main Program S 	weep (Cycle)"			a cu	Count up
Mein [Op1]	Comment				- C1D	Count dow
System blocks	and Manager A					Count down
Figure Technology objects	Network 1:				Comparator op	CERCURS.
• Sei External source files	Comment				 Math functions 	
PLC tags					 Move operation 	NS
Ger PLC data times					Conversion operation	erations
Mutch and farce tables					Program contro	d operati .
t 🔽 Coline backups					 Word logic operation 	retions.
Bares	R.				 Shift and rotate 	¢
Bevice orow data	Maharak 2					
St Program info	I GOWER Z.					
D Pl C alarmy taxt lints						
International modules						
Lineman ad dealerse						
Common data						
 U) Decementation counigs 	v					
Details view						
	10352201					
Name Address					<	
					> Extended instru	uctions
					A 17 start	
					> Technology	
				100%	Communication	n
			Proper	ties 🔄 Info 🗿 💆 Diagnostics	Detional packa	iges
						and the second se

	program1 + PLC_1 [CPU 1214C AC/DC/Rly] + Program blocks + Main [OB1]	L I I X	Instructions	部門
Devices			Options	
1 I I I I I I I I I I I I I I I I I I I	····································	13	101 Int	
- (m.)		75	A I Providence	
1 propram1	Name Detanue Defaulturiue Comment		> (Favorites	
Add new device	I Thout	0	✓ Basic instructions	
A Devices & networks	all a local Cal Root Initial cal of this CB		Nama	Description
PLC 1 (CPU 1214C ACIDCIRIA	Call actuals IX		• General	
Dr Device configuration		1.	Bit logic operations	
W Coline & disgnostics			Imer operations	
Y 📮 Program blocks	Name 166 Counter (S.D.)		 Fi Counter operations 	
Bi Idd ana black	Block title: "Main Program BB Number		🛎 CTU	Count up
ADD NOV DIDLE	Comment Single O Manual		🛎 cro	Count down
The In [Col]	instance		🖀 CTUD	Countuper
• 30 system blocks	* ONetwork 1:		• Comparator operations	
Hechnology objects	Example 1 Hydrocell the function block as a single instance, the function		En Math functions	
(g) External source files	Dide sides in the own material cards of the		+ Two ve operations	
PLC tags			Conversion operations	
PLC data types			• Set Program control operati	
 Watch and force tables 			Mari Ionic constations	
🕨 🙀 Online backups			Chilf and service	
🕨 🥁 Traces				
Device provy data				
Erogrem info	······································			
PLC a larm text lists				
Locel modules	more.			
Europouped devices	Network 2:			
Common data	OF Sentel			
Decumentation settings				
Details view				
Name Address			<	100
			> Extended instructions	
			and the second of the second of the	
		1452	> ¡Technology	
	100%	· · · · · · · · · · · · · · · · · · ·	> Communication	
	2 Properties 7 Info 1 V D	aunostics	> Optional packages	117.14.14.4

4.

Decket Options Program block Progra	roject tree	L 4 program1 + PEC_1 [CPU 1214C AC/DC/Rly] + Program (Instructions	
Image:	Devices				Options	
Main Main Program Program P	4	[] 말 A A # # 4 = 5 = 5 월 # 월 # [] ()	ିରେ ଅଟେ କେଳ୍ଲ କେଳେ କେ ଲେ ଲେଲେ ଲେଲେ ଲେଲେ ଲେଲେ ଲେଲେ ଲେଲେ ଲେ	8	tes las	
program Image: Default value		Main		100	> Favorites	
Model Add new blocks Image and the set with a constraint of the constraint of	program1	Name Data type D	efault value Comment		national data in the national sector	****
Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves a network: Improves network: Impr	Add new device	1 dl 🔻 input		0	· Dasie Instructions	1
Image: Inclusion: Im	A Devices & networks	2 a * Inital_Cal Bool	Initial call of this C6	1	Name	Description
Image: Second particle I	* THE PLC_1 (CPU 1214C AC/DCIRIN)	Bool Rememberce Bool	=True, if remanent data-are available		General	
Colvice & dispositions Court operation Court operatio	Device configuration		(man pres)	1.	Bit logic operations	
Counter question Coun	W Online & diagnostics				Imer operations	
Add we black Weini (Ost) Meini (Ost) Syntam Back Weini (Ost) Comment Syntam Back Weini (Ost) Comment Syntam Back Comment Syntam Back Comment	 R Program blocks 	E CONTRACTOR CONT			• (+1) Counter operations	
Wein (021) Image: Comment Comm	Add new block	 Block title: "Main Program Sweep (Cycle)" 			- CTU	Countup
System block: System	Tesin 10011	Comment			- CTD	Count down
Technology objects Technology object	System blocks				CIUD	Countupen
Extend part of the set of th	Fechnology objects	 Network 1: 			Comparator operations	
• Plc Gag • More appropriate • More appropriate • More appropriate • More appropriate • More appropriate • More appropriate • More appropriate • More appropriate • More appropriot <td>1 Se Esternal source files</td> <td>Comment</td> <td></td> <td></td> <td> Aleth functions </td> <td></td>	1 Se Esternal source files	Comment			 Aleth functions 	
Image: Second	Pictage				Move operations	
Sign Watch and Bloce tables Solos Sign Program info Tisg_1** Sign Program info Tisg_2** Sign Program info Sign Program info Device Configuration Sign Program info Device Configuration Sign P	A TO PLC data tenat	5082			Conversion operations	
Online backapp Social Socia	Watch and home tables	"IEC_Counter_			Program control operati.	
Stock Stock <td< td=""><td>Colina backup</td><td>0_08[1.</td><td></td><td></td><td> Word logic operations </td><td></td></td<>	Colina backup	0_08[1.			 Word logic operations 	
in truck introduction introduction introduction introducti	P 3 Contre Deckaps	340.0 CTD	500.0		 Shift and rotate 	
Device provide as Program inde Proceeding and the set of the set	Rectory data	"Tag_1" Int	"Tag_2"			
Program into Program into Program into Program into Program into Program into Program Proproperting Program Program Proproperting Program	 Device provy data 		()			
Interview	24 Program into	NO MIC				
Ist used modules Image: Second devices Decisits view Image: Second devices Name Image: Second devices Device configuration Image: Second devices Program bodies Image: Second devices Technology Image: Second devices	FLC diarm text lists	10.1 CV "TAL_4"				
> In transported detects Image: point of the second detects Name Image: point of the second detects Notice of degretors Image: point of the second detects Image: point of the second detec	Lacel modules	'Tag_3'				
A grounnen deta Commendeta	Engrouped devices					
Concentration setting: V Details view Module Potential setting: Potential setting: Potential setting: Potential setting: Communication Comm	Gommon dete	5 — PV				
Details view Module Notwork 2: Image: Communication of the structure	Decumentation settings	*				
Module Notwork 2: Notwork 2: C I C C I C C I C C I C C C I C	Details view	CONSIST				
Name Deske configuration	Module	N N				
Name Device conjuston Device dispatcins Program Bocks Exempl source files Unitors Device dispatcins De		NOTWORK Z:				
Anne	12/22				1.11	
Decktrongiguetori > Program Bocks > Program Bocks > Exchalogy objects 105% Improvements Exchalogy objects Improvements Standal source files >	naroe .				<	11.1
Online & biognotics Image: biognotics Image: biognotics Image: biognotics Program Block Image: biognotics Image: biognotics Image: biognotics Technology Communication Image: biognotics Image: biognotics Technology Communication Image: biognotics Image: biognotics	Device computation	8			> Extended instructions	
Program Books Technology objects	Online & diagnostics				> Technolom	
Technology objects Technology objects Technology o	Program blocks		10 AK		- Technology	
Exernal source lifes 😒 Diagnostics 💿 💊 Optional packages	Technology objects		100%	101 million	Communication	
	External source files	×	🖳 Properties 🔄 Info 追 💆 Dia	postics	> Optional packages	

5. Before Enabling the Counter Block.

🔁 🗔 taos project 🍙 💥 🗷 🕞 🗙 🖓 🗄 🥵	1 🔃 🕼 🖳 🛱 💋 Go online 🔒	Saattini 者 🖪 🖪 🗶 😑	Search in project		Totally Integrated /	PORTA
roject tree 🛛 🛛 🕯	program1 + PEC_1 (CPU 1214)	CACDORIy] + Program blocks	• Maini [081]	_##X	Instructions	# E)
Devices					Options	
4 🛛 🗃	高品が会に 10月日	B St 21 E 8 0 6 6 8	1 62 93 62 L L G 61 62 77 24	9	est es	
	Main				> Eavorites	
program t	Name	Data type Default val	ue Comment		Basis instructions	****
💕 Add new device	1 🛋 🕶 input			0	· Dasic instructions	Description
👬 Devices & networks	2 - Initial_Cal	Bool	Initial call of this CG	1	Name .	Derchipport
* _ FLC_1 [CPU 1214C AC/DC/Rby]	3 🕢 # Remanence	Bool	=True, if remanent data are available	1×	General	
If Device configuration	I I I I I I I I I I I I I I I I I I I		Total Total	100	The second second	
🖳 Online & diagnostics					Counter operations	
🔻 😹 Program blocks 🛛 🗮					Counter operations	Provent to an
Add nevi block	 Block title: "Main Program Swe 	seb (clicie)			a cm	Count dram
🚭 Main [051]	(and the second s				Citing Citing	Countuo an
 System blocks 	 Network 1: 					countrap an
Technology objects	Comment				Atath functions	
External source files					Move operations	
PLC tags	E				Conversion operations	
Egi PLC data types	500 COL	erter.			+ Set Propram control operati	
Watch and force tables	O DB	1*			Word logic operations	
Online backups	100 D CR0	•	500.0		Shift and rotate	
 Traces 	"Tag_1" Int		"Tag_2"			
Device provy data		0	()			
22 Program info		NATIVE O	- 1. A. 20			
PLC alarm text lists	140.1	CV-"Tag_d"				
Lagel modules	"Tag_3"					
Engrouped devices						
Gommon deta	5 — PV					
Decumentation settings						
Details view						
Module						
	Notwork 2:					
Name					<	100
Device configuration					> Extended Instructions	
Online & diagnostics					N IT should be be be be	
Program blocks					> Technology	
Echnology objects			102%	in interest	> Communication	
Coternal source files 🗸 🗸 🗸			🖸 Properties 🛛 🗓 Info 🤰 💆 🛛	lagnostics	> Optional packages	
and the second se				10 M M		

6. After Enabling the Counter Block

	🖉 program 🗉 PLC, TTCPD 1214C AC40CHUy) + Program blocks + Mann (2011)	lesting 🖉
levices		Options
	1. 144 박 후 바 특별 프 관광 4. 24 14 16 19 19 19 19 19 19 19 19 19 19 19 19 19	1.100
1 processor 1	name Data per offact when comment	3 CPU operator panes
Add new device	1 - 0 × trut (1)	Call environment
A Devices & networks	T - wool Call Bool Initial call of two OP	No condition defined.
- 🔓 RLC_1 (CPU 1214C AC(DC(RUy)	The Remanence Bool =Tue, if remanents data are available	Change
IT Device configuration		
😼 Online & diagnostics		
* 🙀 trogram blocks		
Add new block	- BROCK UPEL Institution street in Cont.	
😅 Muin (OB1)	Luminem.	
🕨 🕁 System blocks 🖉 🌑	Network 3:	> Renakpoints
 Mechnology objects 	C LINAVAL	A Component
 External source flat 		Call nierarchy
PLa PLC togs		
 III PLC data type: 	8	
 We take a not farser to below 	1950 Fourier	
Dig Online backups	0,08,1*	
s in teres	Since CD TOP TOP T	
• 🕍 Device provy data	"Tag 1" Bt "Tag 2"	
Eg Program info		
FLC alarm test lists		
 Im Local modules 	50.1 TANATA	
🛏 lingrouped devices	""""""""""""""""""""""""""""""""""""""	
Common data	i i	and the state of the later
In Documentation settings	S-PV	NO CAL DELY AVAILABLE
Details view		
Details view		
Deta≋s vlew Iodulo		
Details view	▶ Network 2	
Petalls view oduło amz	Mehandr 2:	
Petalis vlow bdulo ume evice configuration	Noturri 2:	
Petalis vlow odulo wice configuration nine & diagnastica	Metwork 2:	
Antalis view odulo mice exice configuration nice & diagona Kos organi blocks	Network 2:	
Detailitis vitew bodulo witer configuration witer configuration configurations configurations configurations	Metwork 2:	

7. After Load Presetting the Counter Block

Provide Options Provide Notice Notice Notice <th>Write Options program Win prop <</th> <th></th> <th>II A program1 + PLC_1TOPU 1214C AC/06/Rig</th> <th></th> <th></th> <th>(Testing 👘)</th>	Write Options program Win prop <		II A program1 + PLC_1TOPU 1214C AC/06/Rig			(Testing 👘)
Image: Section Section Image: Section	Image: Second and the sec	evices	second in the second second second second			Options
Wein 2 CPL Operation game Wein Device S device Provide de device Image: Second and the second and t	Main 2 CPL dependence CD dependence			2 :	5	
program 1 Image: Data type Data typ	Program 1 Image Outer State type Outer State type Outer State type Outer State type Image State type<		Main			> CPU operator panel
And mere device Project Project <td>And mere device Bervice Revolds IP Device Servolds IP Device Ser</td> <td>program 1</td> <td>🖾 🔍 🔿 Hame Date ty</td> <td>pe Default value Comment</td> <td></td> <td> Call environment </td>	And mere device Bervice Revolds IP Device Servolds IP Device Ser	program 1	🖾 🔍 🔿 Hame Date ty	pe Default value Comment		 Call environment
Ab Device S metrods If Device configuration If Device confi	AD Device S metworks Initial Call Seel Initial Call for Coll Preview Device conjunction Initial Call Seel Initial Call for Coll In Device conjunction Initial Call Seel Initial Call for Coll In Device conjunction Initial Call Seel Initial Call for Coll Initial Call for Coll In Device conjunction Initial Call Seel Initial Call for Coll Initial Call for Coll Initial Call for Coll In Device Conjunction Initial Call Seel Initial Call for Coll	Add new device	1 📲 🕶 input		*	Monorphise defined
Br C: Boxie Stringston Compare Decise Stringston System Biocks Image Stringston Decise Stringston System Stringston System Stringston System Stringston Decise Stringston System Stringston <td>Bit Control Control<td>A Devices & networks</td><td>2 刘 🔹 initial_Cal Bool</td><td>initial call of this OB</td><td></td><td>(No contrade destred</td></td>	Bit Control Control <td>A Devices & networks</td> <td>2 刘 🔹 initial_Cal Bool</td> <td>initial call of this OB</td> <td></td> <td>(No contrade destred</td>	A Devices & networks	2 刘 🔹 initial_Cal Bool	initial call of this OB		(No contrade destred
IP Decise configuration IP + iP + -0 - IP + -1 IP + iP + -0 - IP + -1 IP + iP + -0 - IP + -1 IP + iP + -0 - IP + -1 IP + iP + -0 - IP + -1 IP + iP + -0 - IP + -1 IP + iP + -0 - IP + -1 IP + iP + -0 - IP + -1 IP + iP + -0 - IP + -1 IP + iP + -0 - IP + -1 IP + iP + -0 - IP + -1 IP + iP + -0 - IP + -1 IP + iP + -0 - IP + -1 IP + iP + -0 - IP + -1 IP + iP + -0 - IP + -1 IP + iP + -0 - IP + -1 IP + iP + -0 - IP + -1 IP +	Control de de la seconda	G PLC_1 [CPU 1214C ACIDCRIM	🖬 🔍 a Remanence Ecel	 True, if remanent data are available 	~	Change .
• Decime fields • Program Machs	• Orders & diagnestics • Program Hocks • Add we black • Motion & Diagnestics • Statem Board (Del) • Technology objects • Elevation Secure files • Elevation file • Control of a • Order & Diagnestics • Call bioarchy • Ca	III Device configuration	Date in the second second second			
Book Utile: * Kain Regram Savesp (Spith)* Superior Marks Main (DA1) System Marks Main (DA1)	Because files Metric and force tables Contraction Metric and force tables Metric andifference Metri and force tables Metri and force tables	& Online & diagnostics	⊣⊢⊣⊢⊸⊢ ഈ ⊶ –⁼			
Add move block Second space Y is the block of the block	Add move block System Books System Books	 Program blocks 	 Block titler "Lain Program Swaan (Orla") 			
Notice files N	A bin (Cell) System Bucks Metwork 1: Comment Metwork 1: Metwork 1: Comment Metwork 1:	Add new block	Leminter			
Stylem Backs S	Stylem Books S	Main [081]				
Control of the second sec	Contribution of the solution of the solut	 gji System blocks 	Network 1:			> Breakpoints
Claim back of the set of the	Cannot be a constrained of the second o	 Technology objects 	Comment			Call biorarchy
Control State State Control State Control COLOR State COLOR S	Contro Beckspann Info Contro Beckspan Contro Beckspan Contro Beckspan Contro Beckspann Contrectekspann Contrectekspann Cont	External source files				• Can meascriy
Witch and fore tables The call point in the fore tables The call point and table The call point and	Image: Section of the stability of the stab	PLC tags				
Weth and bries table Control tops Control top Contop Control top Control top Co	Contro belay bes Control by the solution Cont	PLC data types	a Ameri			
Control data C	Control backups Tress T	 Watch and force tables 	*EC Counter			
Image: Second	Stress work of the second sec	 Coline backups 	0_08_1*			
Control data Common d	Positive provides	Traces	540.0 CTD	500.0		
Image: Production statistic: Image: Production statisti: Image: Production statistic:	If Program initial Image: Constraint of the second sec	La Device proxy deta	"Tag_1" Int	"Tag_2"		
Contraction settings Contraction sett	Ple call mattice: Unclaim date: Unc	21 Program info	CD 01			
• Im translation divides • Im translation divides • Im translation • Commendation • Im translation • Decompetition bettings • Im translation • Decompetition bettings • Im translation • Im translation • Im translation	• Im transport derives • Im transpor	PLC a larm text lists				
Tring_3* Common data Tring_3* CV Tring_4* Common data Tring_5* CV Tring_4* Ho call path available	Trong S [*] Commondate Commond	Local modules	500.5 SMA	M.B.		
Commodus	Commendation settings Commendation settings Commendation settings Commendation settings Commendation settings Commendation Commendatino Commendation Commendation Commendatino Commend	H Ungrouped devices	"Teg_5" CVI "Teg	a.4*		
Decomprehation settings v Decomprehation settings v stalls view i duide NetWork 2:	TDecumentation settings v table view table v	😭 Common deta				1
table view dule	etalis view dule Metskojs 2:	Documentation settings	5 N			No can parts asonable
duie Metskots 2: me	dule me me sece configuration inter & disprosite gram Motios termel source files	otalls view	C. Makanan and			
Maria Super Series Control Con	me me configuration Configura					
met once configuration in the second	me vice configuration Afre & disposition Afre & di		Statistics and a			
me configuration	me configuration in the config		Network Zi			
Inter & depositor Inter & Inte	Inire & risk protects Spran Mocks chrology objects termal source files V Properties V Disposition V Disposition	me				
Inter & disposition	Jane & Generation State	vice configuration	0			
sprem blocks threelegy objects threelegy objects To Done To Do	sprem blocks chnology objects ermel source files Up Properties Linfo i & Disgnostices P = -	line 8 diagnostics				
checkego spieces	chralogy objects Properties Up for Up Statement Source Steeley Constraints And Steeley Steeley Statement Source Steeley Steele	ogram blocks			24.47	
remail source files Union and	emel source files 😌 Diagnostics 👘 👘	chnology objects		100%	-	
	Zuchanne Zuchanne	ternal source files		Di Drossetian 1 Linfo	1	
		Portal view	A PCC1 (B Main (DB1)			Connected to PLC_1, via address IP=10

8.

🔒 Save project 📑 💥 🏦 🖭 🗙	5 (11 = -	田田田県 Joconne Jocotine 計画用 × 日上 Searchinprojecte W		PORTA
			- # # X	Testing 🖉 🗊
Pevices				Options
		유,정 위 문 나, 臣 臣 프 이 원 # 원 # 그 위 한 66 전 원 우 64 년 두 64 만 유럽 유 Main	3	> CPU operator panel
_ program 1	A .	Name Date type Default value Comment		✓ Call environment
Add new device		eu * input	-	No condition defined
Devices & network:	-	Initial_Call Bool Initial call of this CE		
PLC_1 [CPU 1214C ACDORIN		Remanence Bool =Inue, if remanent data are sus table	12	Change
Coline & disgnostics	• =	46- 416Ce- 101 6		
System blocks		 Natural 1 		a low souther
Technology objects				Breakpoints
 External source files 		PDCH124428		✓ Call hierarchy
FLC tags	•	1		
LE data type:		5		
Watch and force tables		"EC Counter_		
Contine beckups		0_08_1*		
• a traces		140.0 CTD 1400.0		
Ma nevere provy cara		"Tag_1" Fet "Tag_2"		
The size of the sector of the				
Post a daren sever rest		5		
Local modules		SID 1 YANAYO		
Common data		"Tag_3" CV-"Tag_4"		
T Documentation settings				No call path available
Contraction action by				
Details view				
lodule				
		Network 2:		1
and a second s				1
laure configuration	14			1
Online & diagnostics	100			1
Program blocks				
Schnology objects		100%	Tel manut	
itemal source files	150	O Properties Stiller (1970)	annortics	1
and the state of t		a robernes a number of the second sec	addoadea	
Portal view	A RC 1	29- Main (081)		Connected to PLC 1, we address (P-30

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.



RSLogix Micro Starter - JUMP ANL	D LABEL
<u>File Edit ⊻iew Search Comms Tools</u>	Window Help
	- 3 6 % 3 8 8 8 8 4 + + + +
OFFLINE Image: Second	Node : 1d UPP LeL JSR RIT SBR THO HOR SUS Node : 1d UPP ClieMiss C File Shift/Sequencer AProgram Control A Jump to Label
🕅 JUMP AND LABEL 💶 🗖 🗙	疑 LAD 2 📰 🗖 🖸 🔀
	e IO 2 0000 a IE Imp e Imp Imp e Imp Imp 00001 Imp Imp
Configuration Status	
Continue program exection at label	2:0000 APP AEAD

- 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.

🕆 RSLogix Micro Starter - UNTITL	0	🔚 🗗 🔀
Ble Edit Mew Search Comms Icols	Window Help	
	◇ ◇ ◇ ◇ ○ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇	
OFFLINE		
No Edits		
Driver AB_DF1-1	Node 1d FileMisc & FileShift/Sequencer & Program Control	
💱 UNTITED 🔚 🗖	K 题LAD 2	
🕞 🔄 Project	s Toerie Svitch 2	
🕀 🧱 Help	• <u>10</u>	02:3
🖃 🤮 Controler		(MP)
1 Controller Properties		50
	= Doggle Switch_1	LED_1
	• 10 	2.2
E-19€ Channel Configuration		
E Program Fles	• Q23	BBO
		~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
SYS1-		
- 40 2-		Chan S
E Data Flee	003	(END)
Cross Heterence		
D CO. DHADA		
B TA THER		
NT NTEGER		20
D FR FLOAT		
Dete Logarce		
Configured		
Contract and a		
ECP Contiguistion Flee		
E Corce Fles		
CO - CUTPUT		-1
	4 b \ File 2 /	
Valid Database Entrul		
Yang Marapase Crimy:	23002 APP	

🛔 RSLogix Micro Starter - UNTITLED		
Ele Edit View Search Comms Loois	Window Help	
	· · · · · · · · · · · · · · · · · · ·	
OFFLINE 🛃 No Forces 🛃	🤹 👍 →→ 🎞 3 E 9/2 ()> 40> 40+ 681 685	
No Edits 🔮 Forces Enabled 🔮	Eduly lines (Bt / Tener Counter / Insult2015 t / Conners)	
Driver: A8_DF1-1 N	Node: 1d CEPTOSee Abit A Intervisioner A Inconstruct A compare	
🖹 UNTITUD 📃 🗖 🔀	蹬LAD 2	
😑 😋 Project 📃 🔼	e Togle Switch 2	
🗉 🧰 Help	e [0	Q23
E Controler	0000 •	
Controller Properties		
- 💫 Processor Status	e Toggla Szureh_1	TRD ¹
S Function Files		0.0
	e 3	
E Me Channel Contiguration	e	
Program Files	7002	('END')
SYS0-	0005	((((((((((((((((((((
SYS1-		
		8
- Leia ries		
		88
C2_STATUS		- 23
B3-BNARY		
14 - DIVER		
CS - COUNTER		2.2
R6 - CONTROL		
N7 - INTEGER		
- 📑 F8 - FLOAT		10
🚊 🧰 Data Logging		53
Configuration		
- 📑 Status		
BCP Configuration Files		12
E C Force Files		
- 🖸 OD - OUTPUT		
< I I I I I I I I I I I I I I I I I I I		<u>•</u>
Entry is Valid	2:0001 AFP	

- 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.

RSLogix Micro Starter - UNTITLI		
Ele Edit Mew Search Comms Tools	<u>Window</u> Help	
	· · · · · · · · · · · · · · · · · · ·	
DEE INE		
No Edito		
Driver AB DE1-1	Node: 1d Alb ABit & Timer/Counter & nout/Cutput & Compare & Con	
Provide state of the local state	2 Store a	
	a servar z	
🖃 🔄 Project 💦	e Toggle Sentch_2	
II II Help	e <u>I:0</u>	Q2:3
	e 4	
Controller Properties	a	
S Processor Status	e Toggle Smitch_1	LED_1
	0001	
NO Configuration	a 3	~î
Diservers Files	e	70.0
E even	e	0
# LINO	· · ·	
CAD 2.	e Toggle Series_1	LED_2
Cross Paterance	0003 e	
	e 3 0	82
	0004	('END `)
B3. EMARY		
T4 THER		
CS - COUNTER		
R6 CONTROL		
N7 - INTEGER		
E8 - FLOAT		
= 🧿 Data Looging		
Configuration		
Status		
RCP Configuration Files		
🖃 🥘 Force Files		
- 🖸 OD - OUTPUT	8	-
<		
Tables in Malid	2000 500	
and a solution of the solution	Zacca Ale	

- 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.

🖥 RSLogix Micro Starter - JLIMP Al	ND LABEL	
File Edit View Search Comms Tools	Window Help	
	- 「「「「「「「」」」 「「」」 「」 「」 「」 「」 「」 「」 「」 「」	
Driver AP DE1.1	Node 14 ABR & TimerCounter & Input/Output & Compare & Com	
Chiver Mb_D11-1		
🙀 JUMP AND LABEL	📶 🔀 LAD 2	
🔄 🔄 Project	<u>a</u> h w w w w k k k k	
🛱 🧱 Help		1.11.1
🖶 🧰 Controller	Togele Switch 2	01
Controller Properties		CIMP -
	4	S
	Bul.1766	
IO Configuration		
🗊 👰 Channel Configuration		1_431
📄 🧰 Program Files	1000	O
SYS 0 -	3	1
- 📓 SYS1 -	Bul.1765	Bul1766
ALAD 2 -	023	B3.0
😑 🛄 Data Files	0002 LEL]	
		0
TURTUO - DO	Torele Switch 1	LED 2
- 🛄 H - NPUT	I.0 B3:0	0:0
S2-STATUS		C>
B3 - BINARY	3 0	2
	Bu. 1700	D411/00
COUNTER		
RE - CONTROL	0064	CEND)-
N7 - INTEGER		
F8-FLOAT		
📋 🧰 Data Logging		
- Configuration		
🚽 🔤 Status		
RCP Configuration Files		
🖶 🧰 Force Files		
OC - OUTFUT	3	
a 10 10 10 10 10 10 10 10 10 10 10 10 10		
Help, proce El	20007 000 0500	
c right higgs int	2 30005 PPP 19291	

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.

Add	De Tresser	$(1,1)_{i\in I}$	
- rul	 A state of the sta		
Source	A spectrum		
Source	B · · · · ·		
Dest			

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

٠

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 o	of bit ar	ray (nu	umber of bi	ts)
Word2	Reser	ved			

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.

•

RSL ogix Micro Starter - BIT SHIT L	FT RSS	
Eile Edit View Search Comms Tools W	žindow Help	
	■ "då då"	₩ 2 9 9 9 9 9 9 9 4 - +
REMOTE HUN No Forces Image: Contract of the second	de: 1d	ur uru
R BIT SHIT LEFT.RSS 📃 🗖 🔀	醫 LAD 2	
	End Toggle Switch 1 10 10	ESL Bit Shift Left File K053 Control R6:0 Bit Address I:0/4 Leagth I
RCP Configuration Files	File 2 /	•
For Help, press F1	2:0000	APP READ Disabled

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

RSLogix Micro Starter BIT SHIT LEFT.RSS	
Elle Edit View Search Comms Iools Window Help	and the second sec
	ଓ 🖉 ସ୍ସ୍ 🗖 💠 🔸 ← - →
RUNGTE RUN No Forces Image: State	shift/Sequencer (Pro
🗱 BIT SHIT LEFT.RSS 🛄 🗖 🗙 🌾 LAO 2	
Project A LAN OX 28	
e 🔛 Help	
Controller Togga Switch_1	
🔀 Data File B3 (bin) - BINARY	Pile With Left - CEN >
	Control R60 (DN)
	Length 1«
B30/0 Bade Binary +	(END)
Symbol Columne 16 V	
B 🛄 Desc:	
B3 🕂 Properties Usage Help	
1 II - INPUT	
E S2-STATUS	
L 14-TIMER	
CS - COUNTER	
N7 - INTEGER	
F8-FLOAT	
E Date Logging	
Status	92
RCP Configuration Files	×
File 2	
For Help, press F1 0.0000 APP F	READ Disabled

• 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

📳 RSLogix Micro Starter - BIT SHIT	LEFT.RSS	
Eile Edit View Search Comms Iools	Window Help	
] 🗅 😅 🖬 (골) 🕺 🖷 📼 (🕫 🤊	·	▓[]❷[♥♥□]�] ▶] ♥ →
REMOTE RUN No Forces No Edits # Forces Enabled #	Node: 1d A Move/Logical & File/Misc	uru uru 🕞
🕄 BIT SHIT LEFT.RSS 💶 🗖 🗙	SLAD 2	
E- 🚔 Project	⊷ ₩₩ @× 88	
B- Help Controller Co	Toggle Switch, 1 0000	BSL Bit Shat Left File AG30 Control R6:0 Bit Adhees 100/4 Length 1s
Program Files SYS 0 - SYS 1 - ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓	BINARY	(END)
R Offset 15 14 13 D B3:0 1 1 1 D		
B3:0/0 B3:0/0 Symbol: Deec: B3::: Prop	erties	
Data cogging Configuration Status RCP Configuration Files	TE / File 2	
For Help, press F1	0:0000	APP READ Disabled

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

<u>ON-OFF THE DC MOTOR</u>

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:



PLC Program to Control Motor Speed using VFD Drive





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 4 : PID Sampling Time



InstrumentationTools.com

InstrumentationTools.com

Network 5 : PID Block (Linear PID for Drives Output





InstrumentationTools.com

Network 8 : PID Output Value Moving to Analog Output Channel

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
		South	
7	O:0/3	Yellow	Output
8	O:0/4	East Yellow	Output
		North	
9	O:0/5	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
		South	
14	O:0/10	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:-







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.