TEXAS A&M UNIVERSITY
Industrial Distribution Program

IDIS 400
INDUSTRIAL AUTOMATION

Student Version
LAB MANUAL
Revision B.2

Editors: Roger Lorenzo
Pat Wallace
Amrita Bal
Zhong Chen

TEXAS A&M UNIVERSITY
Industrial Distribution Program

IDIS 400
INDUSTRIAL AUTOMATION

Student Version
LAB MANUAL
Revision B.2

Editors: Roger Lorenzo
Pat Wallace
Amrita Bal
Zhong Chen

Table of Contents

IDIS400 Industrial Automation Lab Manual

0 – Introduction

A. LAB RULES/GUIDELINES

1. Split up into 7 teams of 2 or 3 students. No teams allowed of more than 3 students.
2. Submit 'One' lab report per team.
3. Submit 'One' quiz per person.
4. Every lab, one person works the problem, another person programs and downloads and the other takes notes and includes conclusions. This order switches in a cycle every lab.
5. Do not skip your respective sessions or attend other lab sections without prior TA approval.
6. **Labs are not optional**. MISSING LABS IS NOT ACCEPTABLE. Missed labs (After showing the TA a valid university excuse) must be made up as soon as possible scheduled with the TA; otherwise a zero will be assigned. Missed labs must be done during one of the regular lab sessions after getting the approval of the lab's TA.
7. **The student must have a grade of 70% or higher in the lab to pass the course. A lab grade below 70% will result in an automatic F for the course.**
8. Formula for the lab grade:

Lab work (team effort)	50%
Written report (team effort)	20%
Lab Quiz (individual effort)	20%
Safety (individual effort)	10%

Each Student must sign Lab Safety Agreement (LSA) on line (Howdy) to be allowed to perform lab work

Laboratory Safety: Basic Student Guidelines

Save this document for your own reference

It is a requisite to complete the LSA on line before working in the lab.

Safety is a priority at Texas A&M University!

While it may seem unlikely that an accident could happen to you, you should know the accident rate in universities is 10 to 100 times greater than in the chemical industry. To help prevent accidents, safety notes are included in the lab manual. In addition, any relevant Material Safety Data Sheets (MSDS) are in a laboratory binder and guidelines are posted.

Pay close attention to this information – our goals are:

1. To avoid accidents in the lab, and
2. To respond promptly and appropriately should an accident occur.

Safety depends on you!

It is your responsibility to follow the instructions in the lab manual and any additional guidelines provided by your instructor. It is also your responsibility to be familiar with the location and operation of safety equipment such as eyewash units, showers, fire extinguishers, chemical spill cleanup kits etc.

General Laboratory Safety Guidelines

This is part of the IDIS 300 and 400 Syllabus.

- **Wear appropriate protective clothing**. Do not wear open-toed shoes, sandals, or shirts with dangling sleeves. Tie back long hair and avoid dangling jewelry.
- **Clean** your workstation after each lab period, and return all equipment and materials to appropriate stations **before** leaving the lab.
- Always **turn off the power** before working on any electric circuit or electronic device.
- When operating with electric circuits and electronic devices other than just a computer, you must **work in pairs or teams**.
- When in doubt about the operation of any circuit or device in lab, always **have an instructor check your work** before connecting power to your system.
- **Report any safety issues or violations** that you are aware of as soon as possible to your course instructor and program director.
- Ensure that you have a **safe buffer area around you** and that you are working on an appropriate surface when using soldering irons in the lab.
- Always make sure that all lab equipment, soldering irons, project circuits are **powered down** before leaving your lab area.
- Ensure that your work environment is clear and free of debris before starting your work **AND** after finishing your project.
- **Never block** walkways in the laboratory with lab equipment, cables, and electrical power cords.
- **Do not eat, drink, smoke, or apply cosmetics** in the laboratory.
- **Avoid all horseplay** in the laboratory.
- **Dispose of sharps** waste properly — place broken glass in the glass discard container, metal in the metal waste container, and place other waste materials in the designated container(s). Secure all sharps, including needles, blades, probes, knives, etc.

LABORATORY FOR IDIS-400 INDUSTRIAL AUTOMATION

This course introduces the programmable logic controller (PLC) and its associated applications. Topics include power supplies, input/output modules, ladder logic diagrams, timers, counters, arithmetic & comparison instructions, photoelectric sensors, discrete and analog input/outputs, VFD drives and Panel View display.

Upon completion, students should be able to understand basic PLC systems and create and troubleshoot simple ladder programs using switches, pushbuttons, timers, counters, and arithmetic & comparison instructions.

Lab exercises:

Lab 1 – Logic Circuits
Lab 2 – Transition: Hardwiring Vs PLC Lab
Lab 3 – Ladder Schematics & Diagrams
Lab 4 – Introduction to RSLogix™ 5000 software and ControlLogix® 5550 Programmable Logic Controller
Lab 5 – Advanced Ladder Diagrams
Lab 6 – Timers
Lab 7 – Counters
Lab 8 – Arithmetic & Comparison Instructions
Lab 9 – Timers, Counters and Arithmetic Combinations
Lab 10 – Subroutines and Jump Instruction – (DEMO)
Lab 11 – Analog Inputs/Outputs
Lab 12 – Proximity and Photoelectric Sensors – (DEMO)
LAB 13 – Variable Frequency drives

B. Lab equipment:

Lab component is taught in the Rockwell Automation Laboratory. The Rockwell Automation Laboratory was established by a generous gift from Rockwell Automation, Inc. The lab consists of a main laboratory with 12 student workstations.

The main laboratory contains 12 student workstations and one instructor workstation, each equipped with ControlLogix® programmable logic controller, PanelView™ Operator terminal, PowerFlex®motor motion control unit and various analog/digital and input/output devices, such as push buttons and switches. Figure 0-1 and Figure 0-2 show front and back views of a workstation.

The workstations are networked using EtherNet and ControlNet networks, allowing data to be shared throughout the network. Figure 0-3 shows the current IDIS400 Lab network connections.

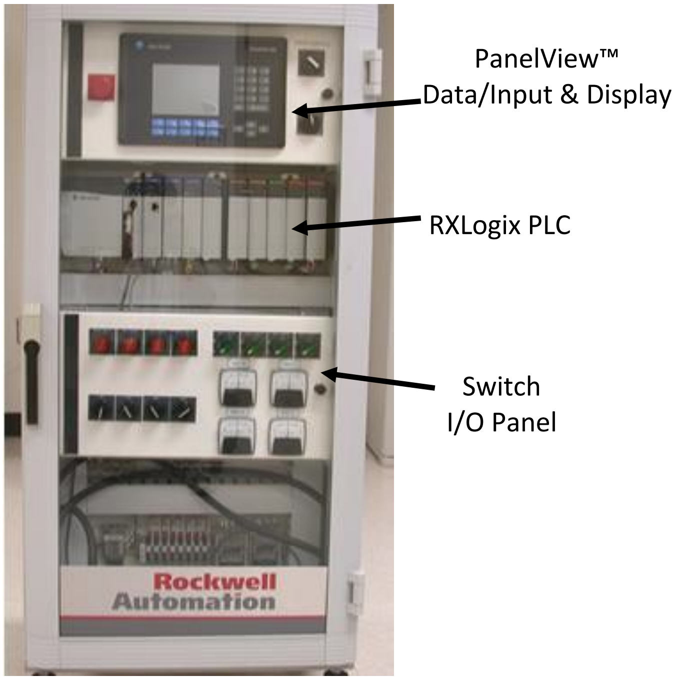


Figure 0-1 Front view of the workstation

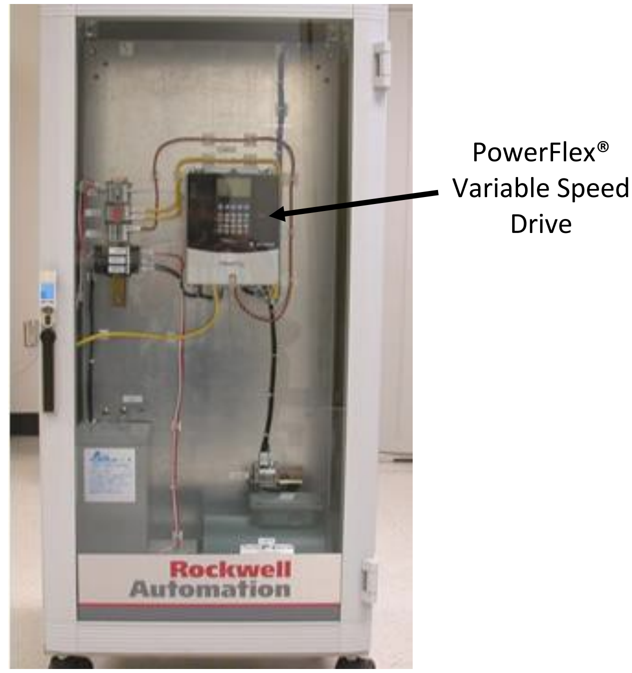


Figure 0-2 Back View of the Workstation

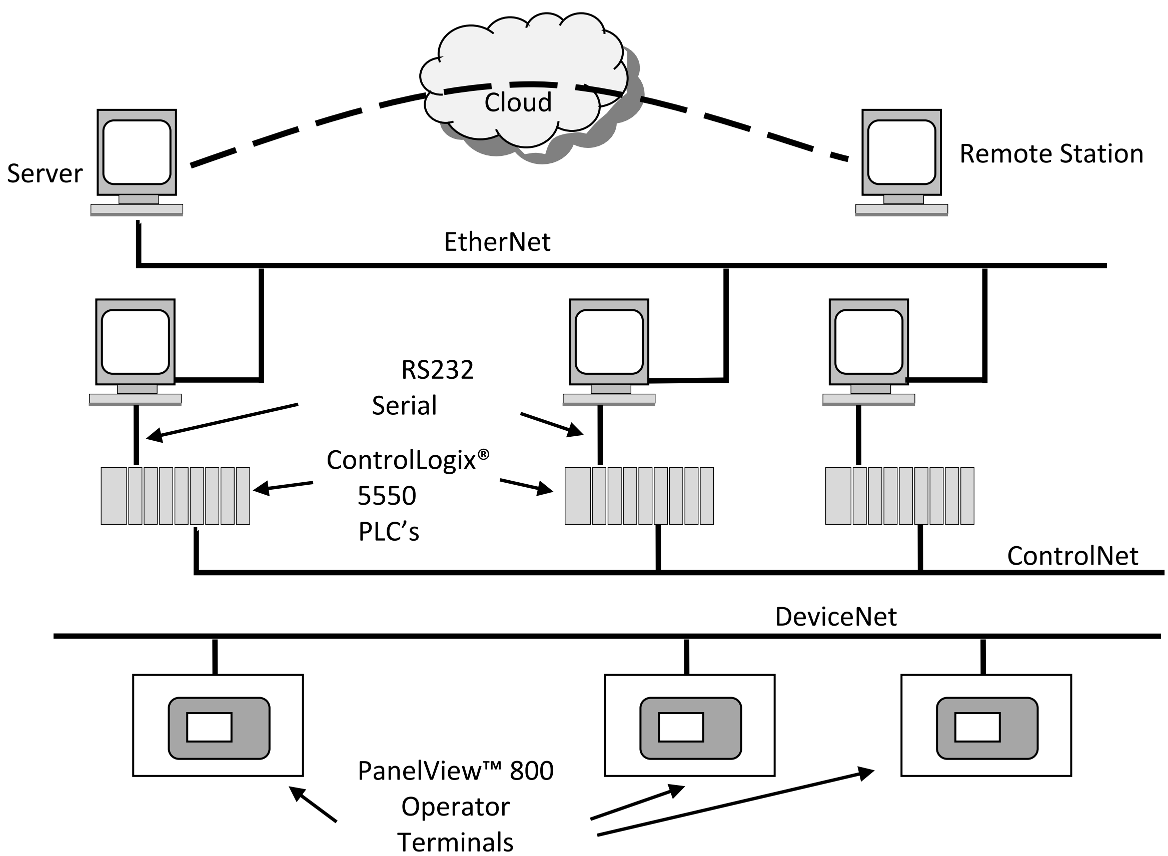


Figure 0-3 IDIS400 Lab Network

1 – Logic Circuits

Name (Last, First): ____________________, ____________________ ____________________, ____________________ ____________________, ____________________ Section Number: IDIS400 ___________	Grade Distribution: 1. Lab Work (0 – 40 pts.) _______ 2. Lab Report (0 – 40 pts.) _______ 3. Quiz (0 – 10 pts.) _______ 4. Safety Practice (0 – 10 pts.) _______ Total _________ Note: 50 pts. for lab work if no quiz is given.

Objectives: This lab is designed to develop an understanding of logic functions and how logic is applied to common problems.

Caution:

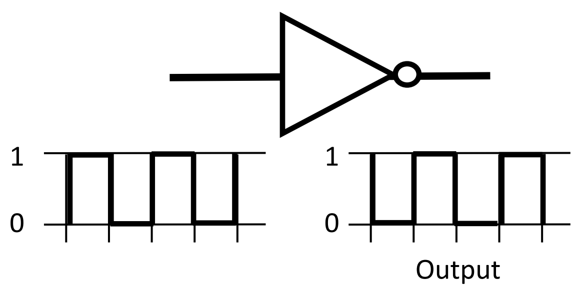
Each test bench has equipment stored on upper shelves. If you notice equipment near the edge, please move it so that it does not fall.

Materials:

Paper and pencil are all required for this lab because the student will try to determine the best logic to solve a real world problem.

Logic Functions:

Inverter

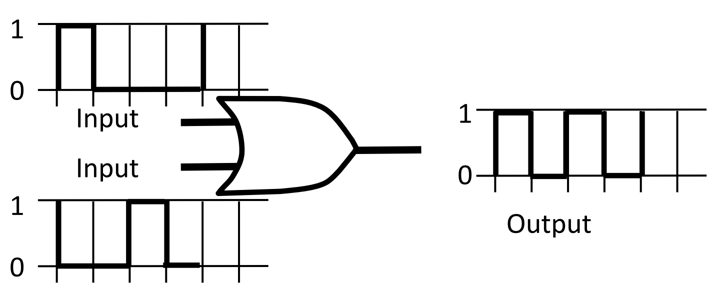


Truth Table

0	1
1	0

Figure 1-1 Digital Inverter

OR Function



Truth Table

Input	Input	Output
0	0	0
0	1	1
1	0	1
1	1	1

Figure 1-2 Digital OR Gate

NOR Function

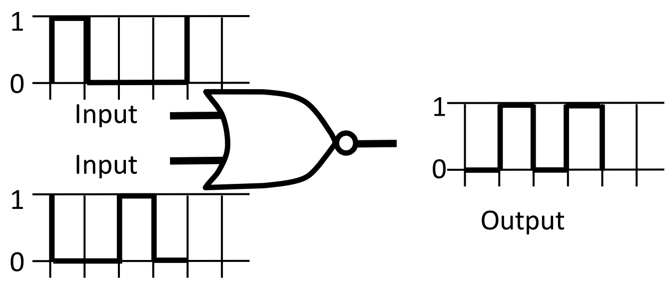


Figure 1-3 Digital NOR Gate

Truth Table

Input A	Input B	Output X
0	0	1
0	1	0
1	0	0
1	1	0

AND Function

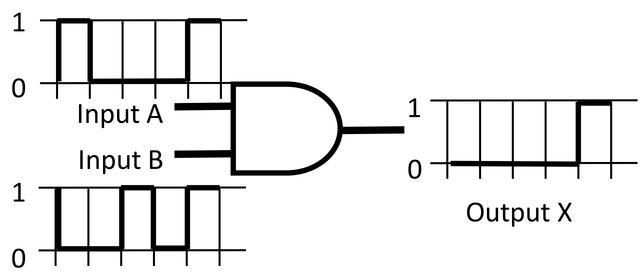


Figure 1-4 Digital AND Gate

Truth Table

Input A	Input B	Output X
0	0	0
0	1	0
1	0	0
1	1	1

NAND Function

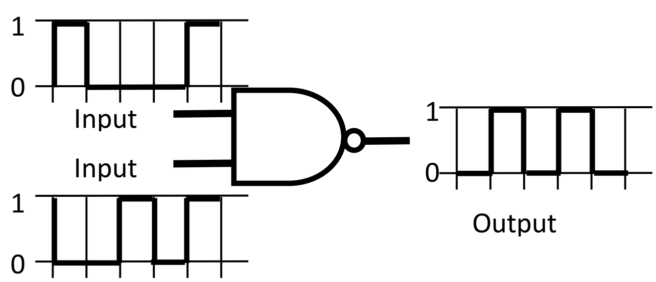


Figure 1-5 Digital NAND Gate

Truth Table

Input A	Input B	Output X
0	0	1
0	1	1
1	0	1
1	1	0

XOR Function

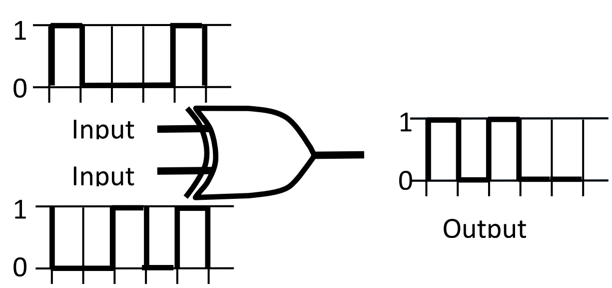
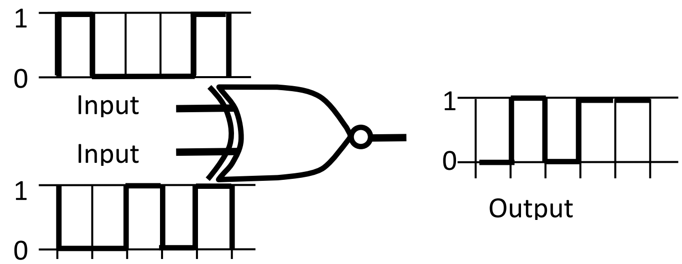


Figure 1-6 Digital Exclusive OR (XOR) Gate

Truth Table

Input A	Input B	Output X
0	0	0
0	1	1
1	0	1
1	1	0

XNOR Function



Truth Table

Input	Input	Output
0	0	1
0	1	0
1	0	0
1	1	1

Figure 1-7 Digital Exclusive NOR (XNOR) Gate

Note: for each problem to be solved, inputs to the logic are either a high (1) or a low (0). Outputs of the logic circuits will be either a high or low where a high is true and a low is false.

Problem 1

Your computer laser printer has a feature that prevents printing if paper is not in the paper tray and the ready light indicates that the printer has not warmed up. Develop a logic diagram that shows the logic of this feature. Note: input from the sensors produces a high (1) or low (0) and the output of the logic is either high (1) or low (0).

Let:

IN_0 = Power ON
IN_1 = Fuser ready (hot)
IN_2 = Paper Tray has paper
OUT_1 = Standby
OUT_2 = Ready to Print

Problem 2

Starting a car requires several conditions to be met. The car needs to be in Park or Neutral with the brake depressed before the ignition switch can engage the starter. Note: the ignitions switch applies power to the engine so the car can start. Oops, this car requires that the driver's safety belt be clasped before the car can be started. Draw a logic diagram that represents this logic. Note: input from the sensors produces a high (1) or low (0) and the output of the logic is either high (1) or low (0).

Let:
IN_0 = Brake switch
IN_1 = Transmission Park switch
IN_2 = Transmission Neutral switch
IN_3 = Seat Belt switch
IN_4 = Ignition/Start switch
OUT_1 = Starter relay

Problem 3

A garage door opener has a safety feature to prevent the door from closing if an object is in the path of the door. This safety feature is designed to prevent injury to pets or children should they stray under the door when it is closing. A wall or remote switch is used to activate the door motor which rotates one way to close the garage door and the other way to open the door. Develop a logic diagram to show how this might work. Note: input from the sensors produces a high (1) or low (0) and the output of the logic is either high (1) or low (0).

Let:
IN_0 = Door switch
IN_1 = Remote switch
IN_2 = Bottom Limit switch
IN_3 = Top limit switch
IN_4 = Photo threshold switch (low if activated)
OUT_1 = Motor up
OUT_2 = Motor down
OUT_3 = Motor On

Name: ___________________ Section: _________ Date: ___________

Print where we can read it!

This is an individual assessment.
Each student must turn in this page for lab credit.

Lab 1 Quiz

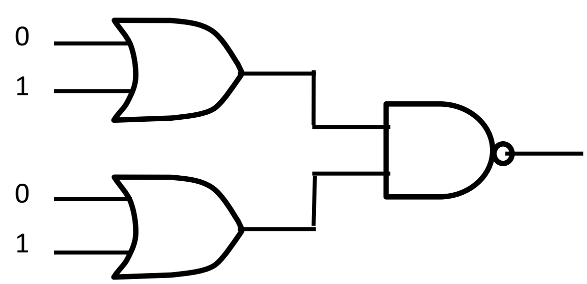
1. If two inputs must be true before an output can be true, the proper logic gate is the ___________.
   a. OR gate
   b. NOR gate
   c. AND gate
   d. NAND gate

2. Given that two inputs must be different before an output is true, the proper logic gate is the _________.
   a. XOR gate
   b. XNOR gate
   c. AND gate
   d. NAND gate

3. How many possible logic combinations can be made with a 4 input AND gate?
   a. 4
   b. 8
   c. 12
   d. 16

4. If a hall light can be controlled by either switch at each end of the hall, what logic function should be considered?
   a. OR gate
   b. NOR gate
   c. AND gate
   d. NAND gate

5. What would be the expected output from the following combinational logic circuit?
   a. high
   b. low



This page is intentionally left blank.

2 –Traditional Hardwiring vs. PLC

<table>
<tr>
<td>
Name (Last, First):
____________________ , ____________________
____________________ , ____________________
____________________ , ____________________
Section Number: IDIS400 ___________
</td>
<td>
Grade Distribution:
5. Lab Work (0 – 40 pts.) _______
6. Lab Report (0 – 40 pts.) _______
7. Quiz (0 – 10 pts.) _______
8. Safety Practice (0 – 10 pts.) _______
Total _________
Note: 50 pts. for lab work if no quiz is given.
</td>
</tr>
</table>

Objectives: At the end of this lab, you will learn how to:
Implement ladder diagrams using push buttons, control relays and lights.

Caution:

Voltage: 24 V
Current: 10 A
Each test bench has equipment stored on upper shelves. If you notice equipment near the edge, please move it so that it does not fall.

Materials:
To perform this lab, the following equipment is needed:
Allen-Bradley™ sensor station.
Few wires
Screw driver
Light Bulb
Push buttons

Lab Notes from TAs presentation:

Lab Notes from TAs presentation:

Procedure A:

This procedure is designed to become familiarized with the components on the Allen-Bradley™ experiment station. The pushbuttons and relays have both normally open (N.O.) and normally closed (N.C.) contacts.

Pushbutton:
A pushbutton has four terminals, two for normally open (N.O.) and two for normally closed (N.C.).

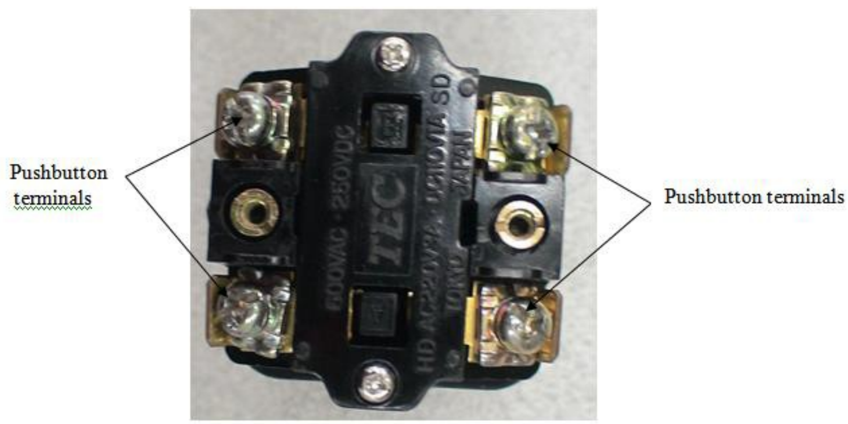


Figure 2-1 Pushbutton Switch Base

Pushbutton configuration:
The pushbutton has four terminals.

Normally Open terminals (N.O.): These terminals are represented by N.O. or by numbers 3 and 4 depending on the company which manufactures it.

Normally closed terminals (N.C.): These terminals are represented by N.C. or by numbers 1 and 2 depending on the company which manufactures it.

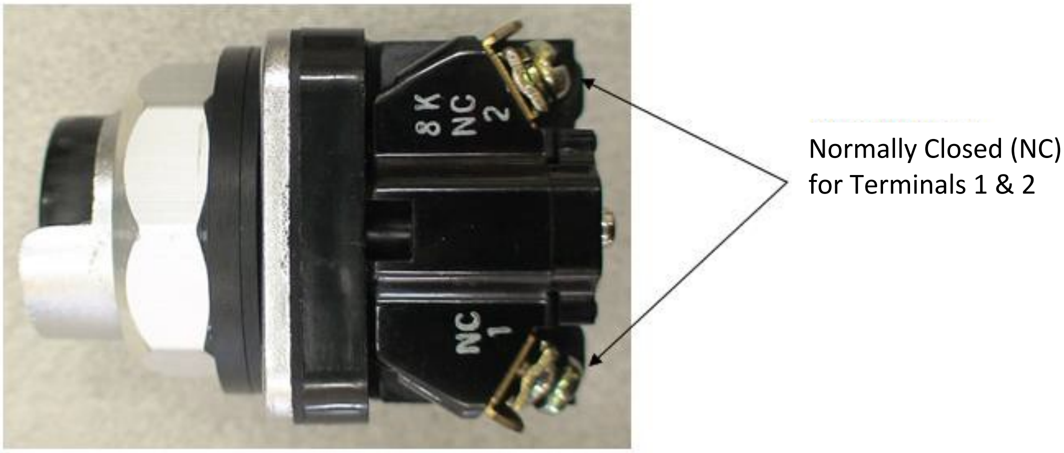
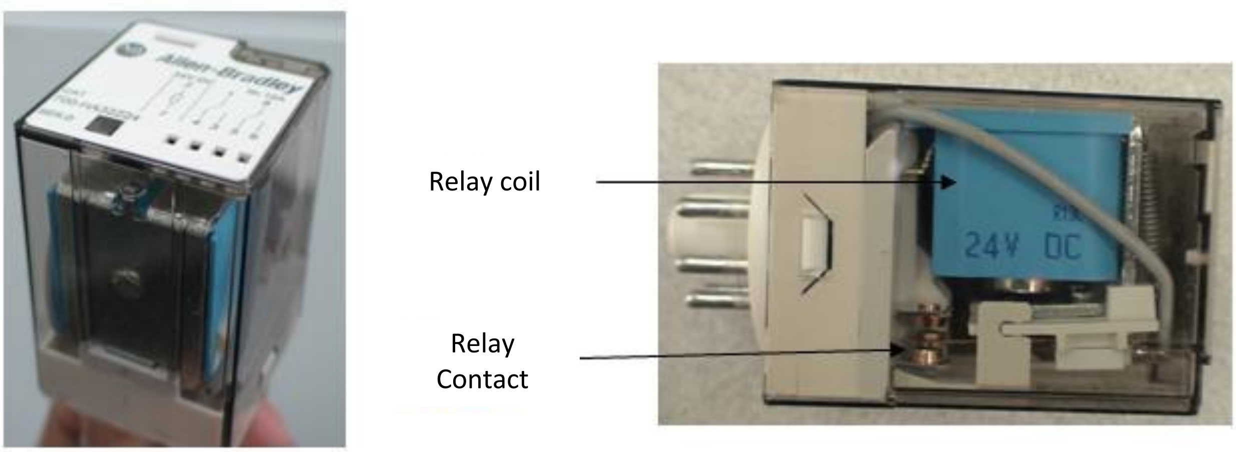


Figure 2-2 Side view of a pushbutton switch

Note: Rotate the pushbutton around to view the Normally Open (N.O.) contacts.

Control Relay (CR):
Take out the control relay from its socket on the Allen-Bradley™ sensor station and inspect the terminal numbers on the relay socket (numbers marked on the piece of paper, above and below the socket).

For coil energizing- use terminal 7, 2
For NO contacts - use terminals 1, 3 or 8, 6
For NC contacts - use terminals 1, 4 or 8, 5



Ice cube relay

Internal view of the Ice cube relay

Figure 2-3 Ice Cube Relay

Procedure B:

<u>Process:</u>
Turn on a light when the normally closed pushbutton (PB1) is actuated and the normally open pushbutton (PB2) is actuated.

A schematic of how the above process operates is shown below.

Assemble the circuitry for the above process utilizing the Allen-Bradley™ sensor station. The lab instructor will explain the steps for assembling the circuit.

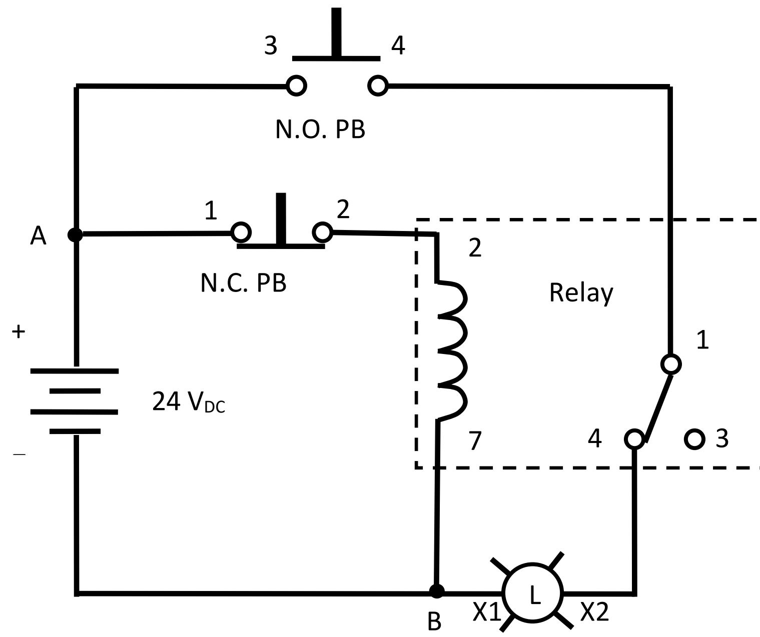


Figure 2-4 Wiring Diagram

The ladder diagram for the process is shown below.

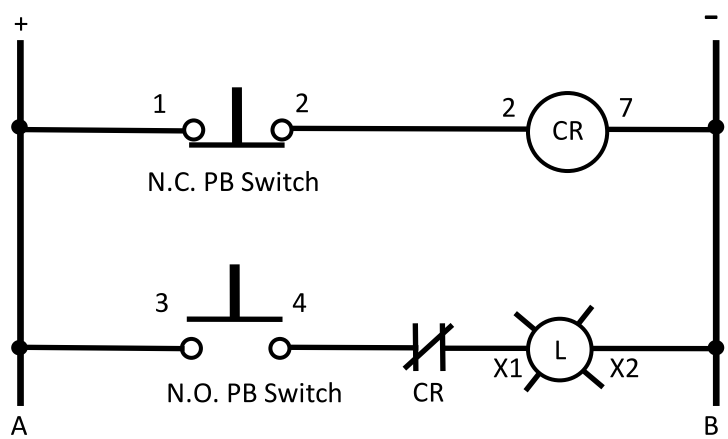


Figure 2-5 Process Ladder Diagram

1. Make the connections as shown in the circuit diagram. Make sure the power cord is unplugged while making connections.
2. Ask your lab instructor to verify the circuit before you power up the station.
3. Show the working circuit to your instructor. Once the circuit is verified, turn off the power supply and unplug the power cord.
4. Remove all the wires that connected to the relay and switches.

Lab Exercise:

Give examples that you observed working with relays (Procedure B).

1.

2.

3.

WRITE YOUR CONCLUSIONS

This page is intentionally left blank.

Name: ___________________ Section: _________ Date: ___________

Print where we can read it!

This is an individual assessment.
Each student must turn in this page for lab credit.

Lab 2 Quiz

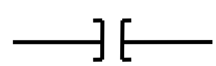
1. The ice cube relay is energized when the appropriate power is applied to terminals ___________.
    a. 1,3
    b. 5,8
    c. 7,2
    d. 6,8

2. A pushbutton switch that allows current flow through it before actuation is called ____________.
    a. Normally open
    b. Momentary
    c. Rotary
    d. Normally closed

3. Relays used in this lab have ___________.
    a. two separate circuits
    b. 10 pins
    c. Normally open contacts are on pins 1 & 4
    d. Normally closed contacts are on pins 1 & 3

4. In the following figure, the resultant of the output is as follows:
    a. 1 before actuation, 0 after actuation
    b. 0 before actuation, 1 after actuation
    c. 1 before actuation, 1 after actuation
    d. 0 before actuation, 0 after actuation

5. In the following figure, the resultant of the output is as follows:
    a. 1 before actuation, 0 after actuation
    b. 0 before actuation, 1 after actuation
    c. 1 before actuation, 1 after actuation
    d. 0 before actuation, 0 after actuation



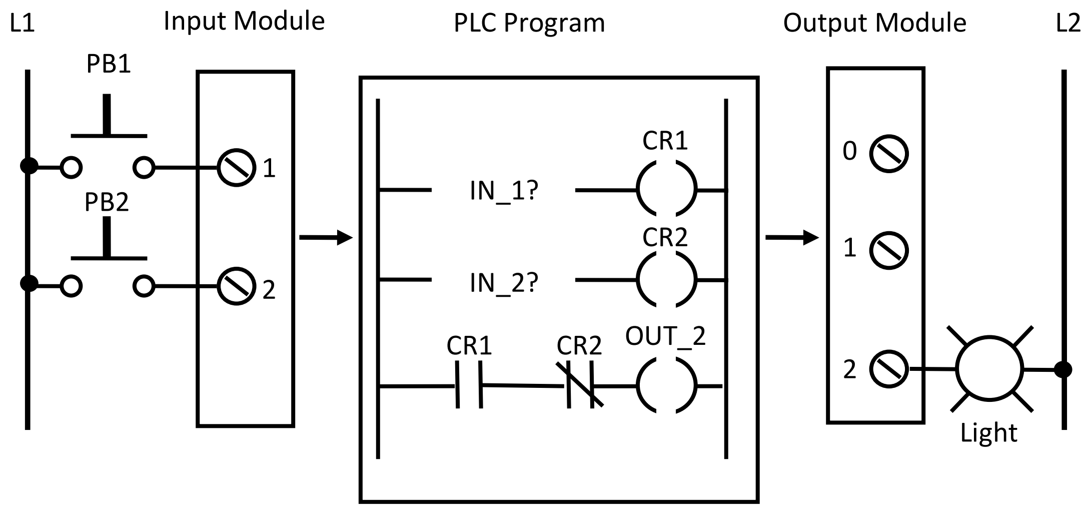
This page is intentionally left blank.

3 – Ladder Schematics & Diagrams

Name (Last, First): ____________________ , ____________________ ____________________ , ____________________ ____________________ , ____________________ Section Number: IDIS400 ___________	Grade Distribution: 9. Lab Work (0 – 40 pts.) _______ 10. Lab Report (0 – 40 pts.) _______ 11. Quiz (0 – 10 pts.) _______ 12. Safety Practice (0 – 10 pts.) _______ Total _________ Note: 50 pts. for lab work if no quiz is given.

Lab Notes from TAs presentation:

1. Complete the ladder diagram to turn on OUT_2 with PB1 and PB2. Assume that the pushbuttons PB1 and PB2 are in their non-actuated positions, and will energize OUT_2 when actuated.



Note: IN_1? and IN_2? are either -| |- or -|/|-

Figure 3-1 Program features

Complete the diagram below:

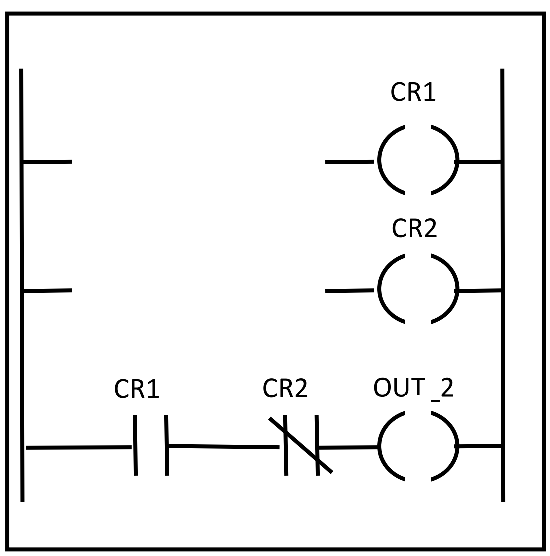


Figure 3-2 Program exercise

Design ladder diagrams for the following problem statements.

2. Energize a pilot lamp (OUT_1) when the normally closed limit switch (IN_5) is not actuated **or** when the normally open pushbutton (IN_1) is actuated.

3. Energize a motor starter (OUT_1) when the normally open pushbutton (IN_1) is actuated and the normally closed limit switch (IN_5) is not actuated.

4. Energize the buzzer (OUT_2) when either the normally open pushbutton (IN_1) is actuated or the normally open limit switch (IN_5) is actuated.

5. Design the ladder logic that will turn ON an output when a switch is closed and turn on a second output when the same switch is open. The two outputs cannot be ON simultaneously.

6. Activating a package conveyor requires a normally open (N.O.) switch IN_1 to be actuated and either the normally closed stop button (pushbutton) IN_3 not be actuated or the normally open limit switch IN_2, indicating there are packages to process is actuated.

WRITE YOUR CONCLUSIONS

Name: ___________________ Section: _________ Date: ___________

Print where we can read it!

This is an individual assessment.
Each student must turn in this page for lab credit.

Lab 3 Quiz

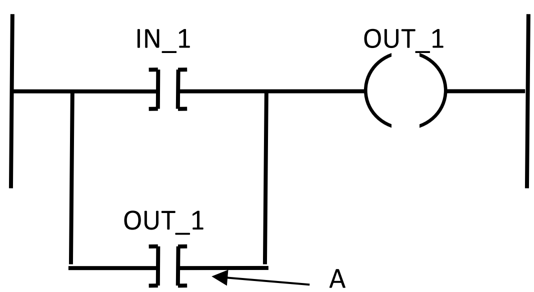
1. The AND function is developed by using _____________.
   a. parallel inputs
   b. serial inputs
   c. combination of parallel and serial inputs
   d. IN_1 plus OUT_1

2. An input that is low before actuation and high after actuation is called ________.
   a. Normally open or XIC
   b. Momentary
   c. Rotary
   d. Normally closed or EIO

3. An input that is high before actuation and low after actuation is called ________.
   a. Normally open or EIC
   b. Momentary
   c. Rotary
   d. Normally closed or XIO

4. In the following figure, the resultant logic of the output is as follows:
   a. NOT function
   b. AND function
   c. OR function
   d. NOR function

5. In the following figure, the purpose of input (marked A) is to:
   a. provide extra data input
   b. provide an input AND function
   c. makes OUT_1 stay true when IN_1 goes false
   d. short out IN_1



This page is intentionally left blank.

4 – Introduction to RSLogix™5000 software and ControlLogix® 5550 Programmable Logic Controller

Name (Last, First):	Grade Distribution:
____________________ , ____________________	13. Lab Work (0 – 40 pts.) _______
____________________ , ____________________	14. Lab Report (0 – 40 pts.) _______
____________________ , ____________________	15. Quiz (0 – 10 pts.) _______
	16. Safety Practice (0 – 10 pts.) _______
	Total _________
Section Number: IDIS400 ___________	Note: 50 pts. for lab work if no quiz is given.

Objectives:

1. Introduce RSLogix™5000 programming software utilized to program the ControlLogix®5550 PLC.
2. Create a simple ladder logic program with an input and an output using the RSLogix™5000 software.
3. Downloading the program to the ControlLogix®5550 PLC and testing it.

Caution:

Voltage: 24 V
Current: 10 A
Each test bench has equipment stored on upper shelves. If you notice equipment near the edge, please move it so that it does not fall.

Lab Notes from TAs presentation:

Overview:

ControlLogix®5550: The ControlLogix® system provides sequential process, motions and drive control together with communications and state-of-the-art I/O in a small, cost-competitive package. The system is modular, so you can design, build and modify it efficiently with significant savings in training and engineering. A simple ControlLogix® system consists of stand-alone controller and I/O modules in a single chassis.

RSLogix™5000: This software supports the ControlLogix® architecture, and specially the ControlLogix® 5550 controller. The RSLogix™5000 package builds on the easy-to-use Ladder Editor to provide a programming environment. Operating in the Microsoft Windows environment, RSLogix™5000 is compatible with programs created with any of Rockwell Software's DOS- based programming packages, as well as RSLogix™ 5000 and 500.

RSLinx™: A communication channel will be needed to be established between the RSLogix™5000 software and the ControlLogix® 5550 PLC. Communications from RSLogix™5000 software takes place through the controller software package, called RSLinx™. RSLogix™5000 utilizes a communication channel established by RSLinx™ to communicate with the ControlLogix® 5550 controller.

Generally, driver software allows a computer to talk to other components. In this case, the RSLinx™ uses drivers to connect the RSLogix™5000 software to the ControlLogix® 5550 controller. RSLinx™ requires the appropriate driver to make this connection. The driver depends on the way the controller is physically connected to the software. A serial RS-232 communication link will be used in this lab.

QUICK TOUR: GETTING STARTED WITH RSLogix™5000

Take some time to get to know the basic window components and toolbars. Please refer to the figure below for the following discussion.

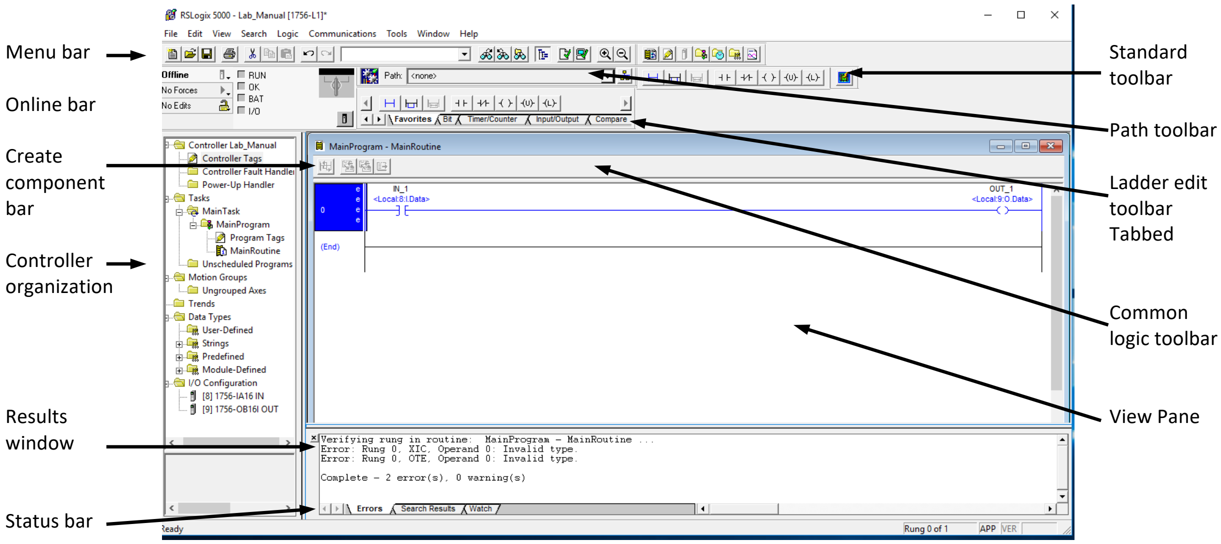


Figure 4-1 RSLogix 5000 software window components

- **The Menu Bar** – Select the functionality from the menus that appear as you click each selection on this bar.
- **The Create Component toolbar** – Use this toolbar to create new project components (E.g. tags, routines, programs, etc.)
- **The Online toolbar** – This toolbar displays the program and the controller status. It indicates the operational mode as well as edits present.
- **The Common Logic toolbar** – This toolbar contains all ladder logic items that are not instructions (E.g. rungs, branches) as well as the commonly used instructions.
- **The Standard toolbar** – This toolbar contains mainly functions (E.g. cut, copy, paste) that will be used repeatedly as the development and testing of the program logic.
- **The Ladder Edit toolbar** – This toolbar contains all of the online editing functions as well as some common editing functions.
- **The Tabbed Instruction toolbar** – This toolbar displays the instruction mnemonics in tabbed categories. When clicking on a category tab, the instruction toolbar just above it changes to show that category of instructions. Click on an instruction to insert it in your ladder program.
- **The Controller Organizer** - The Controller Organizer is a graphical representation of the contents of the controller project. Select the controller organizer by clicking View> Controller Organizer.
- **The Results window** – The window appears at the bottom of the main RSLogix™ 5000 window after the performance of an operation that yields multiple results for errors. This window provides the error and status information on an operation as it executes. Select the results window by clicking View> Results.

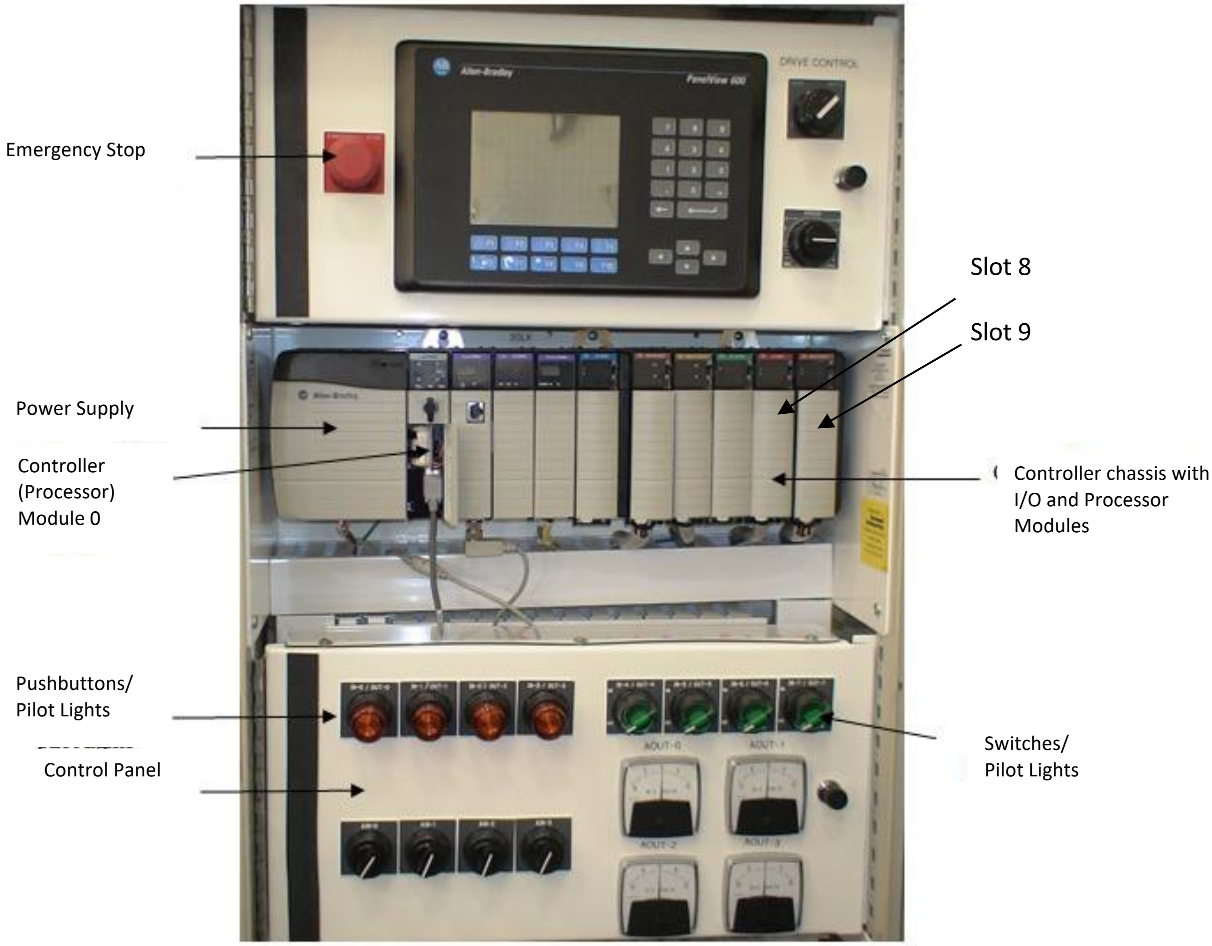


Figure 4-2 ControlLogix® 5550 test cabinet

Figure 4-3 Controller chassis power supply

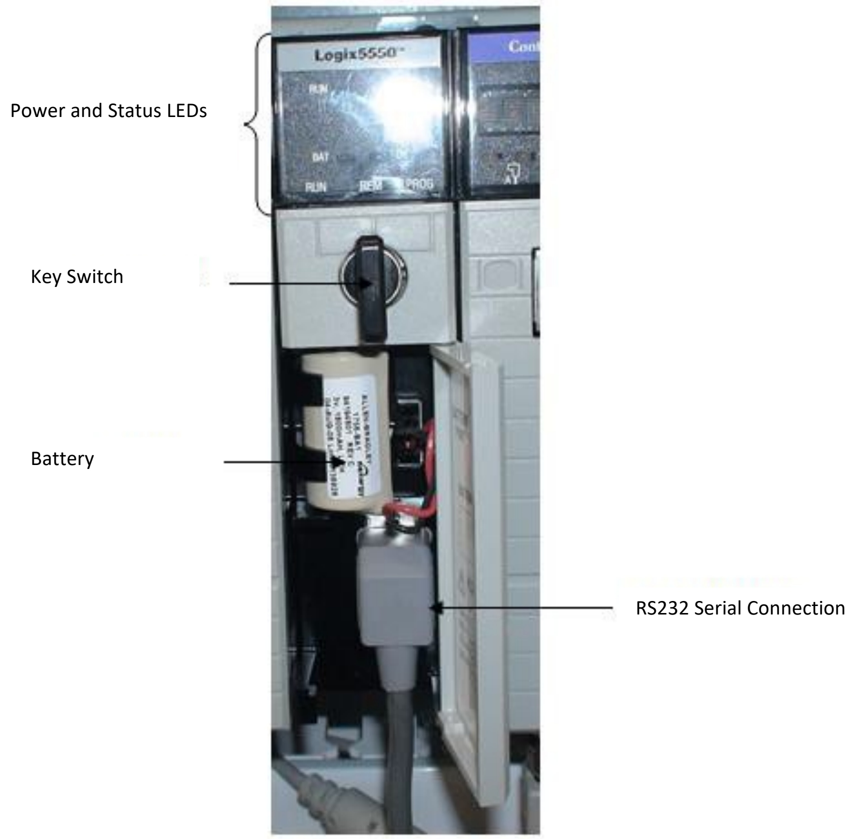


Figure 4-4 Controller processor module

CONTROLLER CONNECTIONS (Serial RS-232 Communication):

Connect one end of the RS-232 communication cable to the **Serial Port** on the Controller Processor module (located in **Slot 0** on the controller chassis) and the other end to the **COM1** port on the back of the computer processor.

OBSERVATIONS AND CHECKS:

- Check the controller chassis size by the number of slots: _________ Slots

- Place a check in the box after you locate each of the following components.

Controller Power On/Off switch	☐
Power LED	☐
Key Switch	☐
Key Switch Positions	☐
Controller status LED’s	☐

PROCEDURE A: IDENTIFYING ControlLogix® 5550 MODULES

ControlLogix® 5550 Module Information:

Located in the ControlLogix® 5550 chassis are 10 modules. Identify the following information for each module and record it in TABLE 4-1below.
Steps for identifying a module are listed on the next page.

MODULE NAME	SLOT NUMBER	ROCKWELL MODEL
	0	
	1	
	2	
	3	
	4	
	5	
	6	
	7	
AC Input Module	8	1756-IA16
	9	

TABLE 4-1 Module information

Note: The ControlNet™ module information must be read via RSLinx™. The lab instructor will explain how to obtain the information.

Example: The AC Input Module is located in Slot 8 and the Rockwell Model number is 1756-IA16.

A for AC, B for DC, F for Fast Response

1756-IA16

ControlLogix®™5550

Number of terminals

I for Input (O for Output)

Figure 4-5 AC Input module (1756-IA16)

PROCEDURE B: Creating a Ladder diagram with a Normally Open Input and downloading the projects containing the diagram to the PLC

Overview: The ladder diagram in Figure 4-6 is a Normally Open (N.O.) Input that is to be controlled by pushbutton 0 on the Rockwell cart's Control Panel of Figure 4-7. Pressing **pushbutton 0** will close the input and complete the current path between the rails. Releasing the pushbutton will break the path.

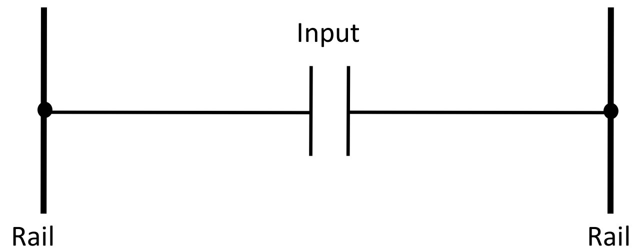


Figure 4-6 Ladder logic

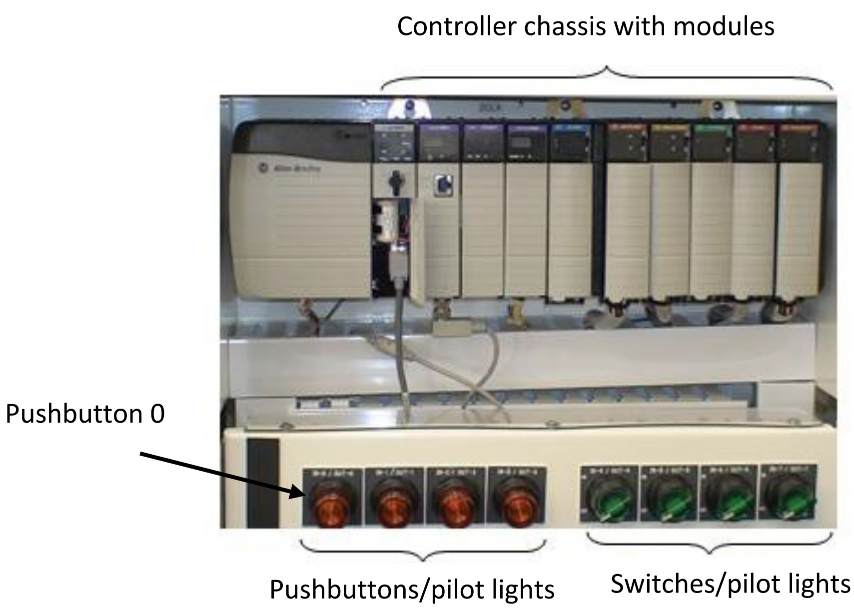


FIGURE 4-7 ControlLogix® 5550 modules and control panel

Note: Creating a ladder diagram and linking the physical act of pressing/ activating **pushbutton 0** requires a series of steps. These steps summarized below, will typically need to be carried out each time a lab is performed. Note that some steps will change in other labs as devices other than pushbuttons are utilized.

Process step overview:

A. Create a New Project File.
B. Configure the Controller Communications.
C. Configure the AC Input module.
D. Configure the Relay Output module.
E. Enter Tag Names, Aliases and Types.
F. Enter the Ladder Diagram.
G. Download Ladder Logic to the Controller.

Step A: Creating a new Project File

1. Turn on the ControlLogix® 5550 by pulling out the Emergency Stop button and moving the Controller On/Off switch, shown in Figure 4-2 ControlLogix® 5550 test cabinet, to On position.

2. Open the RSLogix™5000 software by double clicking on the RSLogix™5000 icon on the desktop. If no icon is present on the desktop, the RSLogix™5000 software may be started by utilizing the Start> Programs option.

3. Select Start>Programs> Rockwell Software> RSLogix 5000 Enterprise Series>RSLogix 5000. Maximize the window.

4. From the File Menu, choose **New**, a project file will be associated with the controller when a new file is created.

5. In the New Controller dialog that opens select the correct controller type (1756-L1 ControlLogix® 5550 Controller) from the dropdown list.

6. In the New Controller dialog box, enter the name you wish to use for the new project (the .ACD extension will be added automatically). **Note: The file name has to begin with an alphabet.**

7. Enter the description for the controller (description field describes the objective of the new project). **Make sure that the revision is 13, chassis type is 1756-A10, and the slot number is 0.**

8. Use the Browse button to choose a directory to save your project file. Save the file under C:\IDIS400 directory and click OK. If the IDIS 400 folder is not present then create it.

Step B: Configuring Controller Communications

RSLinx™ communications software is used to complete communication tasks such as uploading, downloading programs, going online with the controller and viewing the active modules on the network. The software provides the means to setup a communications path between the PC and the PLC and then to utilize the path.

Configure the Serial (AB_DF1-1) Driver:

1. Double click RSLinx™.

2. Click on the Communications Menu and select Configure Drivers.

3. If a serial driver (labeled as AB-DF1-1 in the configured driver window) is present with its status as 'Running', then skip steps 4 to 6 and close RSLinx™.

4. If no serial driver is configured, from the 'Available driver types' drop down menu, select RS-232 DF1 devices.

5. Click Add New, a default name for the driver is selected. Assign the settings as shown in the figure below and click '**AutoConfigure**'. Then click 'OK'.

6. The status for the driver created should indicate running when complete.

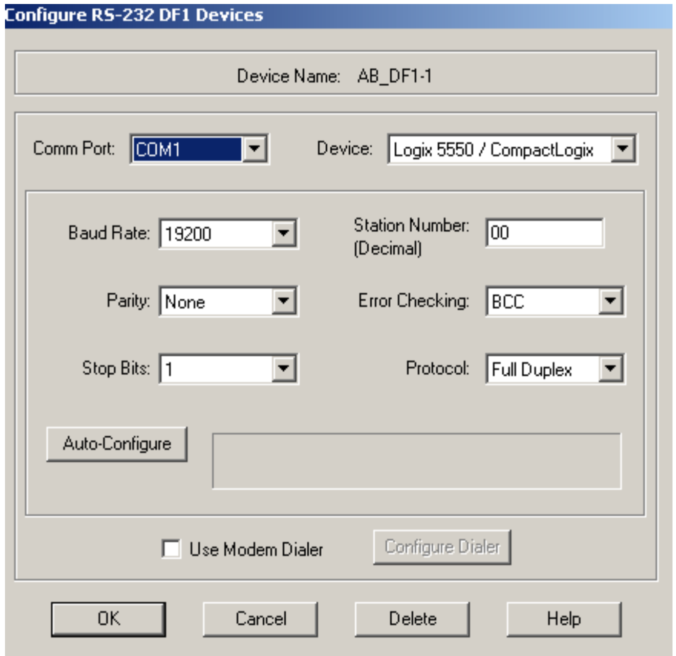


Figure 4-8 Device configuration screen

Follow the steps below to configure the controller.

1. From the Edit Menu, choose Controller Properties. The Controller Properties dialog appears, with the General tab selected.
2. Verify the Controller Name, Description, Slot Number and Chassis Type (You should be able to view the data which you already entered). Click OK.
3. From the Communications Menu, click "Who Active" to configure a communication path to the controller. This launchesthe RSLinx™communication software.
4. Expand AB-DF1-1, DF1 (This is a serial driver). Click on 01, select "Set Project Path". Click Close.The controller and RSLogix™5000 software are now communicating via an RS-232 serial connection.
5. Click Close.

Step C: Configuring the AC INPUT module

Detecting the button press requires knowledge of the device type, signal type and the module model number.

Device type refers to whether a device is an input (I) or an output (O). Pushbutton 0 on the Control Panel is an input device.

Signal type refers to whether the signal is AC or DC and either analog or digital. Pushbutton 0 is internally powered by an AC source and outputs a digital signal.

Therefore, detecting when pushbutton 0 is pressed will require an input module that accepts digital AC signals.

The pushbuttons 0 – 3 are wired to the AC input module, which conveys signals from the pushbuttons to the ControlLogix® 5550.

How does it happen?

Pressing pushbutton 0 results in an AC signal (pushbuttons powered by AC source) sent from the pushbutton to the AC Input module via an internal wiring harness. The module detects the signal as a high or (one) bit and passes it through the AC Input module backplane, across a communication bus, through the controller's backplane connection and into the controller. The controller, programmed accordingly, associates the bit information from terminal 0, of the AC Input module, with the NO input in the ladder logic. This association is performed, in the RSLogix™5000 software, by assigning the physical address of the terminal of the AC Input module to the NO input.

Steps to Configure the AC Input Module

Procedure A verified the AC Input module is installed in the ControlLogix® 5550 chassis in slot 8. Additionally, the module model number/ type should be recorded in TABLE 4-1. This information will be needed to configure the module properly as outlined in the following steps:

1. Right click "I/O configuration" folder in the Controller Organizer and choose "New Module". The window shown in Figure 4-9 will open.

1. Choose 1756-IA16 as the AC Input module. The slot number will be 8. Also select **YES** when asked for 'MajorRevision'.

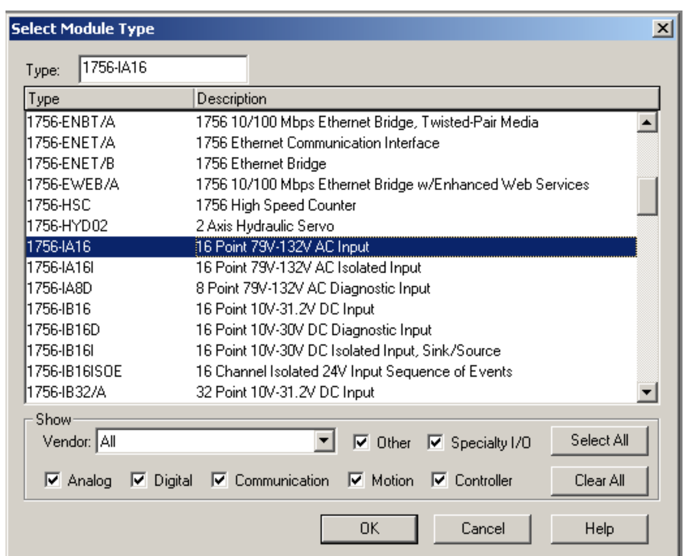


Figure 4-9 Select AC input module screen

Step D: Adding the Output capability to you Ladder Diagram (Configuring the Relay Output module)

The procedure to add an output to the project is similar to the steps which you followed to configure the input. *Configure the Relay Output module (1756-OW16I, slot 9) and choose YES for 'Major Revision'.*

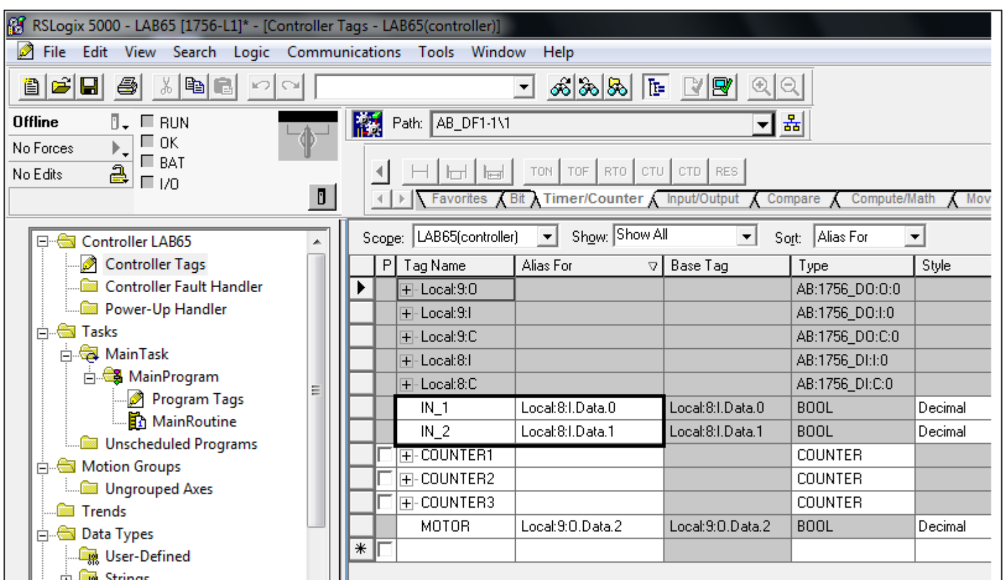
1. Configure the Relay Output module (1756-OW16I, slot 9)
2. Enter the tag name, alias and type (Output1, Local:9:O.Data.1)
3. Enter the output instruction in the ladder editor.
4. The resulting ladder diagram is depicted in Figure 4-6.

Step E: Entering the Tag names, Aliases and Types

Addressing can be a tedious operation as the number of inputs and outputs increase. Therefore, tags or textual names are created for each address. Instead of placing the complex address on each input or output of a ladder diagram, the tag is utilized. These steps outline how to create tags for addresses.

1. Right click on the Controller Tags folder in the Controller Organizer and choose "Edit Tags". The Tag window should have tags associated with the input module Local 8:I. Expand the tree to view all the input bits such as Local:8:I.Data.0 to Local:8:I.Data.31 terminal bit tree.
2. Create a new entry for the tag for your ladder diagram as follows.
3. In the create tag row, indicated by an asterisk * (scroll to the very last row which is empty), enter the name of the tag in the Tag Name column.
4. Enter the ALIASES (if applicable) to associate your tags with the module's inputs or outputs.

Example:



Entering the Tag names, Aliases and Types

Figure 4-10 Tag assignment screen

Tag Name	**Alias for**
IN_1	Local:8:I.Data.0

IN_1 is the tag name that associates the second pushbutton to the input in the ladder diagram. It is aliased as **Local:8:I.Data.0** since the input module is located in **slot 8** and **I** indicates that it is an input device.

Enter a description for the tag if required. Select the data type for the tag.
Note: BOOL is the data type for both Digital Input/ Output devices. (The data type for a timer is **TIMER**)

Step F: Entering the ladder diagram

Steps to enter the ladder logic diagram:

1. Expand the "Main Routine" under "Main Program" on the left of the screen to open the Ladder editor.

2. If there is no empty rung, click on the tab from the Ladder  Instruction toolbar to create a new empty rung.

3. Drag and drop the new empty rung where you want to begin entering the ladder instructions.

4. From the Ladder Instruction toolbar, click on the tab, which corresponds to the instruction group from which you want to add an instruction.
   Note: For common instructions, click on the "Favorites" instruction group.

5. Drag and drop the desired instruction into the ladder editor. The instruction is added to the rung or branch you chose to put it on. Its placement depends on the location of the caret (**green dot**) in the ladder editor.

6. Each instruction must be associated with one of the tags you defined. Then drag and drop the XIO (examine if Open) symbol into place, then click on the"**?**" this will allow you to enter the name of the associated tag. If you have already created the tag, a pull down menu appears on clicking "**Controller Scoped Tags**" from which can select your already defined tag name.

7. Click on the tab from the ladder Instruction toolbar to add additional rungs, branches, branch levels, or instructions as required by your routine.

Overview:

The first rung on the ladder diagram in FIGURE 4-11 has an output which is controlled by a Normally Open pushbutton on the Rockwell cart's control panel in Figure 4-2. Pressing the N.O. pushbutton will close the input and complete the logical path between the two rails to activate the output.

When finished entering your routing, from the main menu, choose **Logic>Verify>Routine**. Any errors in your rung will be sent to the Results window. Make the necessary corrections to your rung.

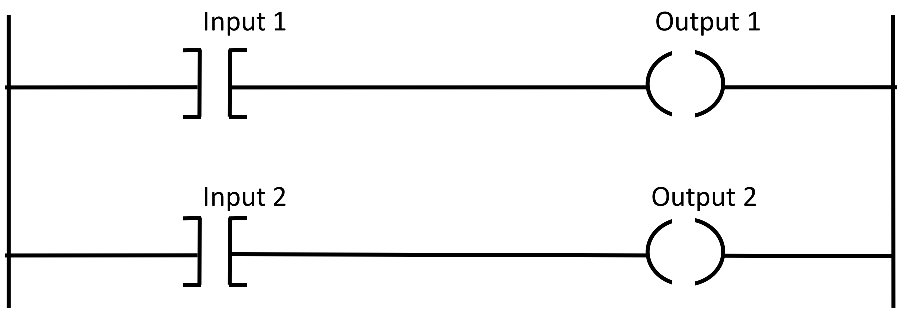


FIGURE 4-11 Controller ladder logic

PROCEDURE G: Downloading the Project to the Controller

Note: The controller key switch must be set to **Program** or **Remote Program** mode before you download the project.

Follow these steps to download your project to the controller after verifying the routine.

1. From the Communications menu, choose "Go Online". RSLinx™ is launched and you are connected to the controller.
2. Now click "**Download**".
3. You are prompted to confirm that you wish to complete the download procedure.
4. Your project is downloaded to the controller. You will see download status and compiler messages in the status bar.
5. Switch the controller to **RUN** mode by changing the controller status bar to **Run**. The controller status bar is located in the online menu bar shown in Figure 4-1.
6. Double click the main routine to open the ladder editor. You can view your ladder logic. If the controller is in **RUN** mode, the left and the right ends of the ladder will be green.

LAB EXERCISE:

1. Create the ladder diagram for the problem statement given by your instructor.
2. Create a new project file.
3. Configure the required input/output modules.
4. Enter the tags for the input/ output components in your ladder diagram.
5. Input the ladder diagram into the ladder editor.
6. Download the program to the controller.
7. Demonstrate your working program to your lab instructor.

WRITE YOUR CONCLUSIONS

Name: ___________________ Section: _________ Date: ___________

Print where we can read it!

This is an individual assessment.
Each student must turn in this page for lab credit.

Lab 4 Quiz

1. The process of setting up the communications link, go to the Communications Menu and click on the sub-menu ___________.
   a. Who Active
   b. RS5000
   c. Host computer
   d. RS232 link

2. A tag is _______.
   a. used to mark each module
   b. to establish a textual name for each address
   c. the name of the program routine
   d. placed as a holding spot for future lines of program code

3. The pushbutton switches on the control panel ________.
   a. are for input only
   b. serve as a digital input source or output indicator
   c. are DC operated switches
   d. are N.C. switches

4. The controller key must be set to Program or Remote Program __________.
   a. to run a program
   b. to turn on the controller
   c. to write a routine
   d. to download a program

5. The module ID (1756-OW16I) refers to _____________.
   a. an input module
   b. the processor module
   c. an output module
   d. the communications module

This page is intentionally left blank.

5 –Advanced Ladder Diagrams

Name (Last, First): ____________________ , ____________________ ____________________ , ____________________ ____________________ , ____________________ Section Number: IDIS400 ___________	Grade Distribution: 17. Lab Work (0 – 40 pts.) _______ 18. Lab Report (0 – 40 pts.) _______ 19. Quiz (0 – 10 pts.) _______ 20. Safety Practice (0 – 10 pts.) _______ Total _________ Note: 50 pts. for lab work if no quiz is given.

Caution:

Voltage: 24 V
Current: 10 A
Each test bench has equipment stored on upper shelves. If you notice equipment near the edge, please move it so that it does not fall.

Note: load the TA assigned problem on the PLC and test to prove your logic is working correctly.

1. Design the ladder logic for turning ON an output (OUT_1) when pushbutton (IN_1) **and** switch (IN_6) are closed. OUT_1 is always on even when IN_1 is open. IN_1 is a normally open momentary pushbutton and IN_6 is a switch. Add an additional normally open momentary pushbutton IN_2 that turns OUT_1 off when it is pressed.

   When pushbutton IN_2 is pressed momentarily, output OUT_1 will remain OFF unless IN_1 and IN_6 are again closed together.

2. Two conveyors are required to move product from production to the packaging area.
    a. Conveyor 1, located on the shop floor, feeds product onto Conveyor 2 which deposits the product in the packaging area. Conveyor 1 and Conveyor 2 are driven by drives OUT_1 and OUT_2 respectively.
    b. Drive OUT_1 is activated when normally open momentary pushbutton IN_1 is actuated and stays on even after IN_1 is released.
    c. Drive OUT_2 is activated when normally open momentary pushbutton IN_2 is actuated when and stays on even after IN_2 is released.
    d. However, drive OUT_2 will not operate unless drive OUT_1 is operating.
    e. A normally open emergency stop switch IN_3 is installed that will shut down both drives simultaneously.

    Design the ladder diagram for this process.

3. A simplified transmission has been developed for the next moon buggy. The process for changing the moon buggy transmission from forward to reverse is as follows:

    a. The buggy is placed into drive when the normally open pushbutton IN_1 is momentarily pressed and the brakes are not applied. The transmission in drive is indicated by light OUT_1. Changing from forward to reverse requires the buggy to be stopped and hence, the brakes must be applied by pressing the normally closed pushbutton IN_2.

    b. Applying the brakes not only stops the vehicles forward progress but also places the transmission in neutral. While still applying the brakes, the moon buggy transmission may now be placed in reverse by pressing the normally open pushbutton IN_3. The transmission in reverse is indicated by the light OUT_2. A safety feature built into the transmission will not allow both the forward and the reverse to be engaged at the same time.

    Design the ladder logic for this moon buggy transmission.

4. The process requirement is to control the flow of water from the overhead water tank to the distribution system.

   a) Water is pumped into a tank from a water reservoir.
   b) The water is pumped until the high level sensor in the tank is actuated. (i.e. when the water reaches the HIGH LEVEL and therefore the pump is turn off)

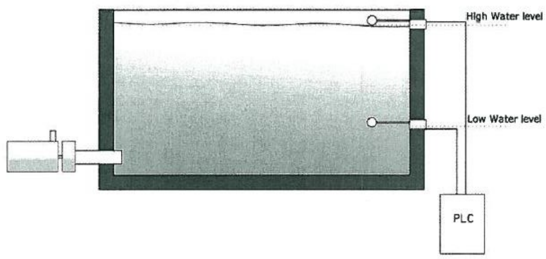


Figure 5-1 High level water tank

   c) If the water level falls below the low level sensor, the pump begins pumping water.

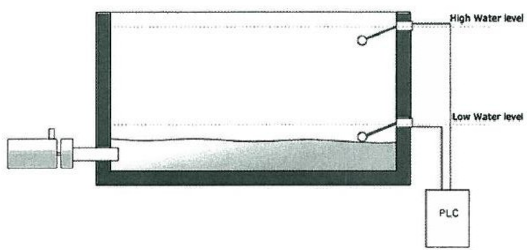


Figure 5-2 Low level water tank

Note:

To reduce complexity, assume that initially the water level is below the low level sensor; therefore, the low level sensor is activated and the pump should be ON.

The low level sensor (IN_1) and the high level sensor (IN_2) are considered as normally open and are of the same type. The high level sensor will actuate only when the water level reaches the preset value.

Design the ladder diagram to simulate this process.

5. The lighting system in a store is controlled by a PLC. The lighting system (status indicated by OUT_1) is turned ON when the switch IN_5 **and** switch IN_6 are turned ON. A status light OUT_2 placed in the maintenance room should turn ON indicating the lighting system is ON.

   Note: If there is a power failure (normally open pushbutton IN_1) the PLC will turn OFF and the design must ensure the lighting system returns to the same state it was in when the power failure occurred, once the power is restored.

   Design the ladder diagram to simulate this process.

WRITE YOUR CONCLUSIONS

Name: ____________________ Section: _________ Date: ___________

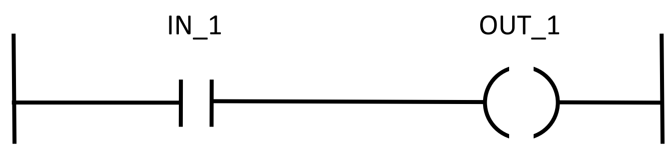
Print where we can read it!

This is an individual assessment.
Each student must turn in this page for lab credit.

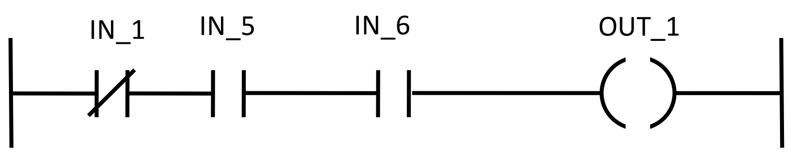
Lab 5 Quiz

1. If an XIC contact is connected to a momentary N.O. pushbutton switch, one method to maintain the logic signal is to _____________.
   a. create an AND function with another XIC contact driven by the rung output
   b. create an OR function with another XIC contact driven by the rung output
   c. loop the output back to the input
   d. use the inverter function

2. If IN_1 is true in the figure, then OUT_1 is ___________.
   a. true
   b. false
   c. 0
   d. off



3. The conditions for the rung in the figure are as follows for the following rung: _________.
   a. IN_1 true, IN_5 false, IN_6 true, OUT_1 true
   b. IN_1 true, IN_5 true, IN_6 true, OUT_1 true
   c. IN_1 false, IN_5 true, IN_6 true, OUT_1 true
   d. IN_1 false, IN_5 true, IN_6 true, OUT_1 false



4. A normally closed relay contact can be represented by a __________.
   a. jump to close routine
   b. proper tag name
   c. XIC command
   d. XIO command

5. An output (OUT_1) can be used to provide a digital signal to the output module or _____________.
   a. a program input signal
   b. a processor command
   c. an input module process variable signal
   d. a communications command

This page is intentionally left blank.

6 – Timers

<table>
<tr>
<td>Name (Last, First):
____________________ , ____________________
____________________ , ____________________
____________________ , ____________________
Section Number: IDIS400 ___________</td>
<td>Grade Distribution:
21. Lab Work (0 – 40 pts.) _______
22. Lab Report (0 – 40 pts.) _______
23. Quiz (0 – 10 pts.) _______
24. Safety Practice (0 – 10 pts.) _______
Total _________
Note: 50 pts. for lab work if no quiz is given.</td>
</tr>
</table>

Caution:

Voltage: 24 V

Current: 10 A

Each test bench has equipment stored on upper shelves. If you notice equipment near the edge, please move it so that it does not fall.

Construct a ladder diagram for each of the following problems. Input and output tags in the ladder diagram must correspond to the actual inputs and output tags called out in the problem statement. Pay attention to which inputs are pushbuttons or switches when constructing your ladder diagrams. (Answers will be graded based on whether the problem requires the use of a pushbutton or switch)

Notes on timers:

1. Initializing a timer: A timer is needs to be defined/ initialized before being used in the ladder logic. A timer tag is created by going to the tag table and creating a tag name for the timer. In the fourth column, for data type, browse for the data-type TIMER. The 'Alias For' field in the second column needs to be left empty. The tag thus created can be assigned to a T-ON or a T-OFF timer.

2. A TON timer starts and continues to time as long as the input to it is high. Also, once the DONE bit of the TON timer is set, it remains set only as long as the input to the timer is high or the timer is enabled. If the input to the timer goes low at any point of time, the timer cuts off and resets to 0 and all the bits of the timer are cleared (DN, EN, and TT).

3. A TOF timer times only when the input to it has gone low from a state of being high. The DONE bit of a TOF timer is set to high initially and reset to low once the timing interval has elapsed. If the input to a TOF timer goes high at any point during or after the timing process, the timer is reset causing the DONE bit to be set again and the remaining bits of the timer are cleared (EN, TT).

4. The only way to reset a retentive timer is to use the RES command.

5. A retentive timer is essentially a TON timer with the property of retaining its accumulated value when it is de-energized.

6. The preset of all the timers is in **milliseconds** by default and will not accept any decimal values.
   Note: In order to time 12.5 seconds, "12500" is to be typed into the preset field.

Note: Timers do not require aliases when creating the corresponding tag:

For example:	**Tag name**	**Type**
	Timer1	TIMER

Note: load the TA assigned problem on the PLC and test to prove your diagram is working correctly.

1. Using a **TON** construct ladder diagram to energize the pilot lamp (OUT_1) 10 seconds after a normally open pushbutton IN_1 is pressed momentarily. Design the ladder diagram to simulate this process.

2. Using a **TOF**, energize an output OUT_1 when the normally open pushbutton IN_1 has been false for over 12 seconds. Design the ladder diagram to simulate this process.

1. After input IN_1 (normally open pushbutton) has been held closed for at least 7 seconds, energize output OUT_1. Normally open pushbutton IN_2 resets the timer. Design the ladder diagram to simulate this process.

2. Using a bit other than **DN**, construct a ladder diagram that will turn “ON” a pilot lamp when normally open pushbutton IN_5 is turned “ON”. The pilot lamp (OUT_1) will turn “OFF” 17 seconds after IN_5 is turned “OFF”. Normally open pushbutton IN_2 resets the timer.

3. Using a **TON**, **TOF,** or both construct a ladder diagram for the following scenario:
    a. A light OUT_1 should turn "ON" when normally open pushbutton IN_1 is momentarily pressed. The light should remain "ON" for 10 seconds and then turn "OFF".
    b. A second light OUT_2 will turn "ON" 5 seconds after the first light OUT_1 turns "OFF". It should remain "ON" indefinitely. Normally open pushbutton IN_2 resets the timers.

    Design the ladder diagram to simulate this process.

WRITE YOUR CONCLUSIONS

Name: ________________ Section: _________ Date: ___________
Print where we can read it!

This is an individual assessment.
Each student must turn in this page for lab credit.

Lab 6 Quiz

1. The TON timer will continue to count time ___________.
   a. as long as the input is high
   b. until the ACC = PRE
   c. only when the preset is less than the ACC
   d. only when the input is low

2. The DN bit of a TOF timer ____________.
   a. starts low and goes high when ACC = PRE
   b. is not used
   c. starts high and goes low when the ACC = PRE
   d. tracks the TT bit

3. A retentive timer ____________.
   a. does not store timing counts
   b. TT bit is equal to the inverted EN bit
   c. does not require a reset pulse to clear the ACCUM
   d. may be started and stopped and sets the DN bit only when the ACCUM reaches PRESET

4. The RS5000 software uses a time base of __________.
   a. 1 microsecond
   b. 1 millisecond
   c. 0.1 second
   d. 1 second

5. A method used to extend a timer's maximum time is to _____________.
   a. duplex timers
   b. multiplex enable bits
   c. cascade multiple timers
   d. dedicating timer functions

This page is intentionally left blank.

7 – Counters

Name (Last, First):	Grade Distribution:
____________________ , ____________________	25. Lab Work (0 – 40 pts.) _______
____________________ , ____________________	26. Lab Report (0 – 40 pts.) _______
____________________ , ____________________	27. Quiz (0 – 10 pts.) _______
	28. Safety Practice (0 – 10 pts.) _______
	Total __________
Section Number: IDIS400 ___________	Note: 50 pts. for lab work if no quiz is given.

Caution:

Voltage: 24 V
Current: 10 A
Each test bench has equipment stored on upper shelves. If you notice equipment near the edge, please move it so that it does not fall.

Notes on counters:

Initializing a counter:

1. A counter is needs to be defined/ initialized before being used in the ladder logic program. A counter tag is created by going to the tag table and creating a tag name for the counter.
2. In the fourth column, for data type, browse for the data-type COUNTER. The 'Alias For' field in the second column needs to be left empty. The tag thus created can be assigned to an UP or a DOWN counter.
3. If a DOWN counter is used to count 'x' number of pulses, the preset of that counter must be (x-1).
4. The DONE bit of a DOWN counter is set to 1 initially and reset to 0 after the counting is done. The DONE bit of an UP counter is initially low and set to high after the counting is done.
5. Whenever we have a situation where-in the DONE bit of a counter is used to reset itself, careful observation shows that if the counter is an UP counter, it resets to "1" instead of resetting to "0".

   Note 1: This is because when the counter's accumulated value is one less than its preset; the next pulse being counted puts the counter in a state of being **DONE**. When the next pulse is given to the counter, the counter gets **DONE**, resets to "0" and counts UP by "1" during the **LOW-HIGH-LOW** transition of the pulse because we have the built the ladder with the done bit of the counter resetting it. This reset to "1" of the counter can be avoided. **How??**

   Note 2: Counters do not require aliases when creating the corresponding tag:

For example:	**Tag name**	**Type**
	Counter1	COUNTER

Note: load the TA assigned problem on the PLC and test to prove your diagram is working correctly.

1. Write a ladder diagram to energize the pilot lamp OUT_1 after the normally open pushbutton IN_1 is pressed **10** times. Normally open pushbutton IN_2 resets the counter

2. The number of people entering a room is counted by a sensor IN_1. A pilot light is turned **ON** (OUT_1), when 14 people have entered the room. The count is then manually reset using normally open pushbutton IN_2 to begin counting the next 14 people. Design the ladder diagram to simulate this process.

3. A machine M (OUT_2) is to be turned **ON** after both of the following conditions are met: Counter A reaches 10 or above and Counter B reaches -16 starting from 0. Both the counters are controlled by the normally open pushbutton IN_2. Normally open pushbutton IN_3 resets both the counters. Design the ladder diagram to simulate this process.

4. There are two conveyors FC1 and FC2 that feed parts to a main conveyor. A proximity sensor is placed at the end of each conveyor (represented by normally open pushbutton IN_1 and IN_2) to count its parts. As each part passes the proximity sensor it emits a single pulse. The proximity sensor pulses are sent to the counters. Each counter then shows the number of parts being placed on the main conveyor. An indicator light OUT_1 should be **ON** when both of the following conditions are met: FC1 has finished feeding 10 parts and FC2 has finished feeding 15 parts onto the main conveyor. A normally open pushbutton IN_3 is used to reset both the counters. Design the ladder diagram to simulate this process.

5. A motor M (OUT_3) is to be turned **ON** when both of the following conditions are met: Counter A reaches "-7" starting from 0 and Counter B reaches "15". Counters A and B are controlled by normally open pushbuttons IN_1 and IN_2 respectively. Motor M is turned **OFF** and the system is reset if the total number of input pulses from IN_1 and IN_2 equals 25. (**Do not** use any MATH functions for this task.) Design the ladder diagram to simulate this process.

6. Company XYZ manufactures a printed circuit board (PCB) for a security system. The manufacturing process requires 12 components to be placed on the board. The placing of each of these 12 components is simulated by a normally open pushbutton (IN_1). The board is then wave soldered to attach these components to the circuit board. The wave soldering process takes 5 seconds. Company XYZ's logo is then stamped on the board in three different directions (represented by normally open pushbutton IN_2). Completing the stamping process automatically triggers a system reset and another board may then be processed.

   Draw a ladder diagram to simulate this process. A light must come **ON** denoting the wave soldering process is active. Stamping the company logo is simulated by pressing a pushbutton.

WRITE YOUR CONCLUSIONS

Name: ____________________ Section: _________ Date: ___________

Print where we can read it!

This is an individual assessment.
Each student must turn in this page for lab credit.

Lab 7 Quiz

1. The CTU counter will stop counting __________.
   a. as long as its rung condition transitions from low to high
   b. until the DN bit is set
   c. only when the preset is less than the ACC
   d. when the reset transitions from low to high

2. The DN bit of a CTD counter __________.
   a. starts low and goes high when ACC = PRE
   b. toggles with the rung pulses
   c. starts high and goes low when the ACC = PRE
   d. stops the count

3. An up-down counter can be developed by ____________.
   a. selecting the up-down icon
   b. combining an CTU with a CTD, each assigned the same counter location
   c. does not require a reset pulse to clear the ACCUM
   d. may be started and stopped and sets the DN bit only when the ACCUM reaches PRESET

4. Counters require __________ to zero the ACCUM value.
   a. the CD set to 1
   b. the rung bit to be set to 0
   c. the DN bit to be set
   d. a reset pulse

5. The counter tab is under the ________.
   a. ladder edit toolbar
   b. combined logic toolbar
   c. standard toolbar
   d. common logic toolbar

This page is intentionally left blank.

8 – Arithmetic & Comparison Instructions

<table>
<tr>
<td>Name (Last, First):
____________________ , ___________________
____________________ , ___________________
____________________ , ___________________
Section Number: IDIS400 ___________</td>
<td>Grade Distribution:
29. Lab Work (0 – 40 pts.) _______
30. Lab Report (0 – 40 pts.) _______
31. Quiz (0 – 10 pts.) _______
32. Safety Practice (0 – 10 pts.) _______
Total _________
Note: 50 pts. for lab work if no quiz is given.</td>
</tr>
</table>

Caution:

Voltage: 24 V
Current: 10 A
Each test bench has equipment stored on upper shelves. If you notice equipment near the edge, please move it so that it does not fall.

Note: load the TA assigned problem on the PLC and test to prove your diagram is working correctly.

1. Design a ladder diagram that stores the number of input pulses entered manually by normally open pushbuttons IN_0 and IN_1 within a 30 second time period (This timer is trigged by normally open pushbutton IN_2). The count from IN_0 should be stored in the variable INPUT_0 and the count from IN_1 stored in the variable INPUT_1. Finally add both counts and store this value in the variable TOTAL and normally open pushbutton IN_3 resets the timer and counters.

2. Design a ladder diagram for the following formula:

$$c = \sqrt{a^2 + b^2}$$

Note: a, b and **c** must be assigned memory locations and their alias must be **a, b** and **c** respectively. Values are to be assigned to "**a"** and "**b"** by counting the input pulses from normally open pushbuttons IN_0 and IN_1 respectively. Finally the formula above should only execute when IN_3 is pressed, signifying the appropriate values are loaded into" **a"** and "**b"**. **Hint:** The square root of an integer is not necessarily an integer. Normally open pushbutton IN_2 resets all the counters.

3. A main conveyor is fed by two conveyors "A" and "B" controlled by normally open pushbuttons IN_0 and IN_1 respectively. Feeder conveyor A (IN_0) inputs 6 cans of canned soda on the main conveyor at one time. Feeder conveyor B (IN_1) inputs 8 cans of canned soda on the main conveyor at one time. Conveyors "A" and "B" have counters that count the number of **sets** leaving them.

   Construct a PLC program to give a total can count on the main conveyor and store the result in TOTAL_CAN memory location.

4. Two robots are utilized to place parts on a fork lift. The maximum number of parts the fork lift may carry is 10. The number of parts placed by robot 1 is counted by the normally open pushbutton IN_0. The number of parts placed by robot 2 is counted by normally open pushbutton IN_1.
   a. Pilot light OUT_0 should come **ON** when the maximum capacity of the fork is reached. Use a manual operation (normally open pushbutton IN_3) to reset the count on both the robots.
   b. **Note:** Although the problem can be solved using counters alone, use arithmetic and comparison instructions.

5. Design a ladder diagram for the following formula:

$$z = \left(\sqrt{x} + \sqrt{y}\right)^2$$

Note: **x**, **y** and **z** must be assigned memory locations and their alias must "**x**", "**y**" and "**z**" respectively. Values are to be assigned to "**x**" and "**y**" by counting the input pulses from normally open pushbuttons IN_0 and IN_1, respectively. Finally the formula above should only execute when normally open pushbutton IN_3 is pressed, signifying appropriate values are loaded into "**x**" and "**y**" memory locations.

Hint: The square root of an integer is not necessarily an integer.

WRITE YOUR CONCLUSIONS

Name: ___________________ Section: _________ Date: ___________

Print where we can read it!

This is an individual assessment.
Each student must turn in this page for lab credit.

Lab 8 Quiz

1. The ADD instruction will _____________.
   a. require an output to function
   b. send its sum to a defined destination
   c. require an equation expression
   d. resets upon execution when its enable is low

2. The CPT instruction requires __________.
   a. a high enable input pulse to function
   b. a destination and mathematical expression
   c. is limited to addition and subtraction
   d. requires a reset

3. What is true for a divide operation?
   a. The task can be accomplished by either CPT or DIV when the function is enabled.
   b. It only functions when source A or B is fixed.
   c. Its output must be a integer.
   d. It performs the task when the enable line goes low.

4. Complex math formulas often follow _________.
   a. individual math functions
   b. combined math functions
   c. a math loop
   d. limited applications

5. The compute function requires _____.
   a. a reset
   b. a source location
   c. an enable
   d. a math expression (formula)

This page is intentionally left blank.

9 – Timers, Counters and Arithmetic Combinations

Name (Last, First):	Grade Distribution:
____________________ , ____________________	33. Lab Work (0 – 40 pts.) _______
____________________ , ____________________	34. Lab Report (0 – 40 pts.) _______
____________________ , ____________________	35. Quiz (0 – 10 pts.) _______
	36. Safety Practice (0 – 10 pts.) _______
	Total _________
Section Number: IDIS400 ___________	Note: 50 pts. for lab work if no quiz is given.

Caution:

Voltage: 24 V
Current: 10 A
Each test bench has equipment stored on upper shelves. If you notice equipment near the edge, please move it so that it does not fall.

Note: load the TA assigned problem on the PLC and test to prove your diagram is working correctly.

1. Design a ladder logic diagram to simulate a clock. The clock must include hours, minutes, seconds, and indicate AM and PM cycle. Time will be indicated in military time, so afternoon hours must be indicated as 12- 24. The AM cycle must be indicated by a pilot lamp (OUT_1) being OFF and the PM cycle is indicated by the pilot lamp being **ON**. Switch IN_6 will start the clock.

   Note that due to timing considerations, you may have to modify or reduce the actual timing sequence, instead of waiting 24 hours for a complete cycle.

2. A sensor (normally open pushbutton IN_0) detects when a car enters a currently empty parking spot. The owner of the car inserts 4 quarters (normally open pushbutton IN_1) into the parking meter for the car for a 5 minute time period. If the car remains in the parking space for more than 5 minutes, a pilot lamp turns **ON** indicating the car needs to be ticketed. A normally open pushbutton (IN_2) is used to reset the timer and the pilot lamp once the car is removed from the spot. Additionally, count the number of cars parked in that spot for a random 40 minute period during the day that is triggered by a switch IN_5. Normally open pushbutton IN_6 is used to reset the system manually Design a ladder logic diagram to perform this function.

WRITE YOUR CONCLUSIONS

Name: ___________________ Section: _________ Date: ___________

Print where we can read it!

This is an individual assessment.
Each student must turn in this page for lab credit.

Lab 9 Quiz

1. The processor completes tasks _______.
   a. in sequential order in the routine starting at the top proceeding to the bottom
   b. counters and timers first then math operations
   c. in groups until the first reset
   d. as long as the number of rungs does not exceed 5

2. The tasks being performed must be __________.
   a. sequenced by the overall requirements then individual steps
   b. sequenced by individual steps then overall requirement
   c. sequenced in order of performance
   d. randomized

3. Common practice is to ________.
   a. perform any count operations first then timer operations
   b. reset counters and timers after the routine is completed
   c. reset the enable lines to false upon completion of routine completion.
   d. limit the use of the DN bit

4. The clock example requires _________.
   a. the hour counter to be the first counter
   b. the minute counter to be the first counter
   c. the A.M/P.M. counter to be the first counter
   d. the second timer to be the first operation

5. The counter and timer tabs is under the ________.
   a. ladder edit toolbar
   b. combined logic toolbar
   c. standard toolbar
   d. common logic toolbar

This page is intentionally left blank.

10 – Jump to Subroutine, Subroutine, and Return instructions (Demo)

Name (Last, First):	Grade Distribution:
______________, ______________	37. Lab Work (0 – 40 pts.) ______
______________, ______________	38. Lab Report (0 – 40 pts.) ______
______________, ______________	39. Quiz (0 – 10 pts.) ______
	40. Safety Practice (0 – 10 pts.) ______
	Total ________
Section Number: IDIS400 __________	Note: 50 pts. for lab work if no quiz is given.

Objective:

Upon completion of this topic the student would be able to:

1. Demonstrate the knowledge of how a subroutine executes
2. Use a subroutine
3. Create a subroutine

Caution:

Voltage: 24 V

Current: 10 A

Each test bench has equipment stored on upper shelves. If you notice equipment near the edge, please move it so that it does not fall.

Overview:

SBR (Subroutine) is a ladder routine called from another ladder program.

JSR (Jump to Subroutine) instruction jumps execution to a different routine. The SBR and RET instructions are not optional instructions that exchange data with the JSR instruction.

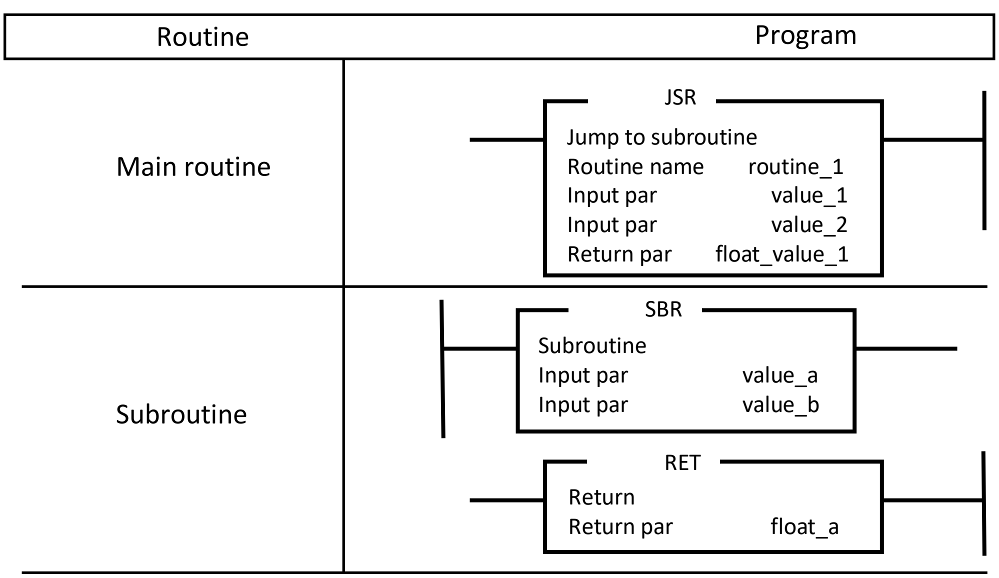


Figure 10-1 Jump routine

SBR and **RET** instructions are found under Program Controls in the Instruction Toolbar.

Task: Create an RSLogix™ 5000 program for following condition:

A High Intensity Activated Crosswalk (HAWK) traffic light signal is activated when a pedestrian pushes the pedestrian crossing Stop Button. When the Stop Button is **not** activated by pedestrians, the green light indicates it does not restrict vehicle movements on the underlying roadway and pedestrians are shown a RED light signal. When a pedestrian activates the HAWK, vehicle movements on the underlying roadway are warned to stop by a yellow signal for 5s.

The signal will turn red from yellow after 5s, and stay red for 10s to allow pedestrians to cross.

During this 10s, the green light will stay on for first 5s to allow pedestrians start passing.

After the first 5s, green light will become red blinking light to warn the pedestrians of pending change. After that, traffic signal becomes green again which allow vehicle passing.

Figure 10-2 Traffic control problem

To solve this problem, we can follow these steps:

1. Go to Start > Programs > RSLogix™5000
2. From the File Menu, create a new project. In the New Controller dialog box, select the correct controller type; enter the name you wish to use; choose revision 13.

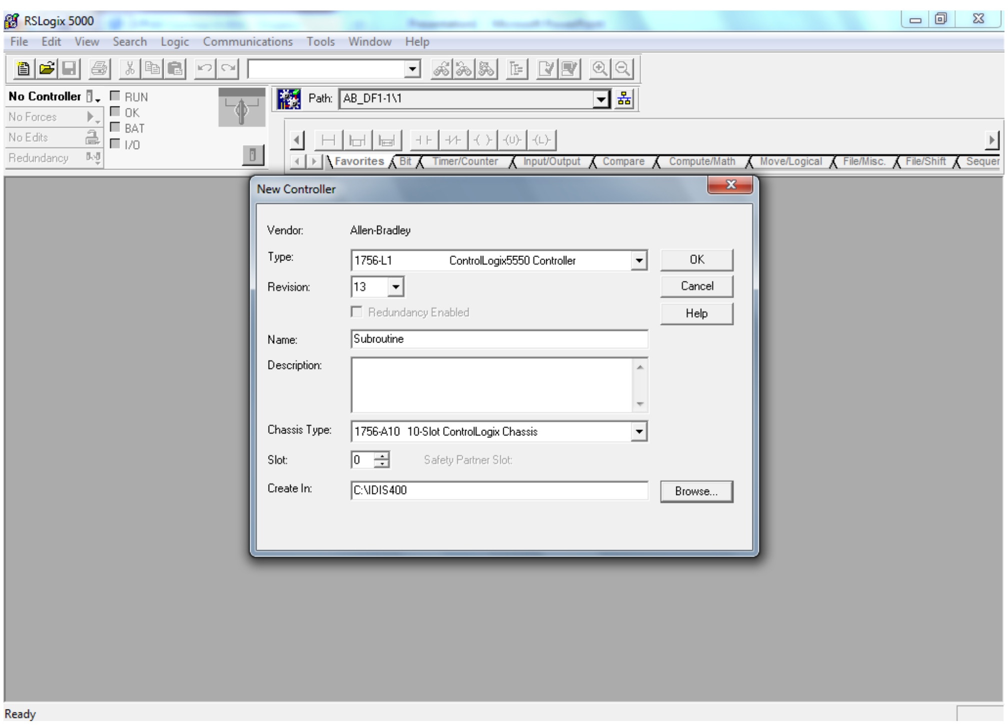


Figure 10-3 Controller dialog screen

3. Configure the Input Module and Output Module. Choose 1756-IA16 as the AC Input module with slot number 8. Also select **YES** when asked for 'Major Revision'. Choose 1756-OW16I as the AC output module with slot number 9, also select **YES** when asked for 'Major Revision'.

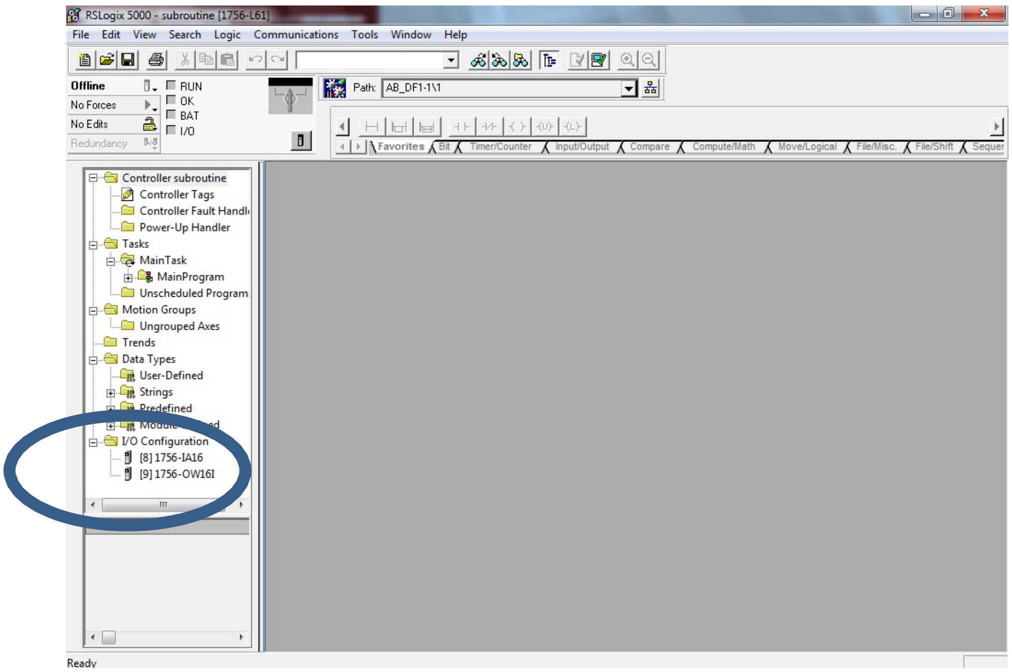


Figure 10-4 I/O selection screen

4. Entering the Controller Tag names, Aliases and Types.

5. Create subroutine:

- Expand the '**Main Routine**' under '**Main Program**' on the left of the screen. Right click and choose '**New Routine**'. Enter the name you wish to use for the subroutine. You can have multiple subroutines but only one main routine in your main task. Just make sure different subroutines have different names.

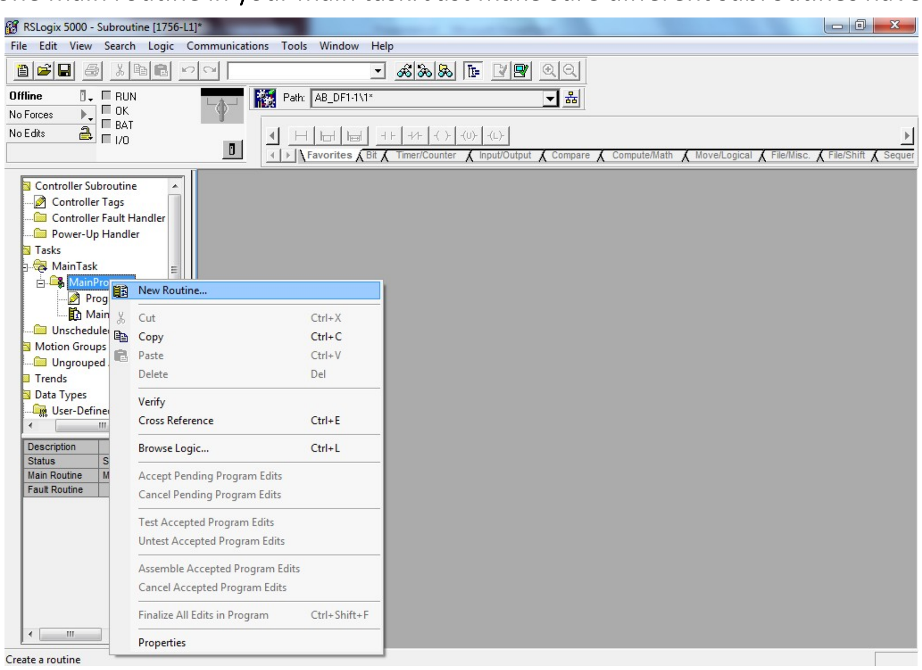


Figure 10-5 Selection of new routine screen

- Open the subroutine Ladder editor and enter your ladder diagram. Subroutine ladder diagram start with **SBR** Instruction (under Program Controls in the Instruction Toolbar) and **NOP** (No Operation Performed). Enter Input Parameter should be the same as JSR Input Parameter in the main routine (described in step 6). Subroutine ends with **RET** (Return).
- Finish the subroutine by inserting body part of the ladder diagram.

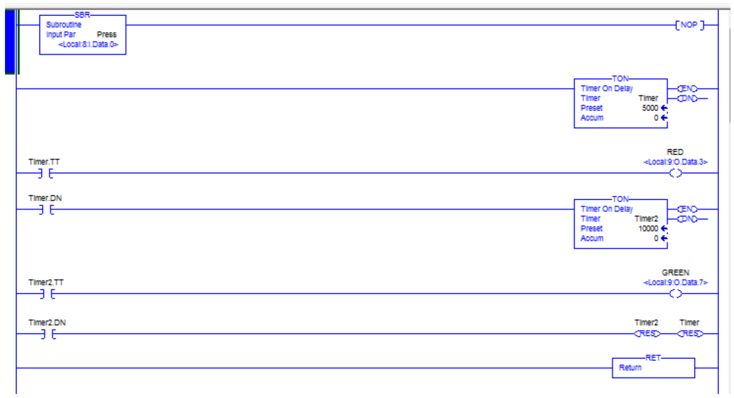


Figure 10-6 Subroutine body

3. Create Main Routine and call subroutine

- Open main routine ladder editor and build main ladder diagram.
- To call subroutine which was built before, insert JSR Instruction (under Program Controls in the Instruction Toolbar). Enter subroutine's name and Input Parameter.

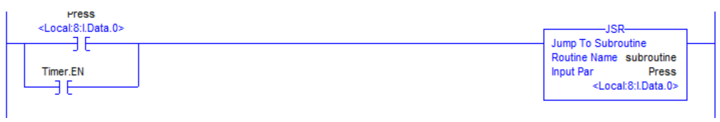


Figure 10-7 Calling the subroutine

4. Pedestrian passing light blinking

- In the real application, most of the High Intensity Activated Crosswalk (HAWK) signal start blinking before it turns red. In this exercise, assume the red light starts blinking after 10s.
- There are many ways to achieve the light blinking, we can set the light on for 1s and then off for 1s.

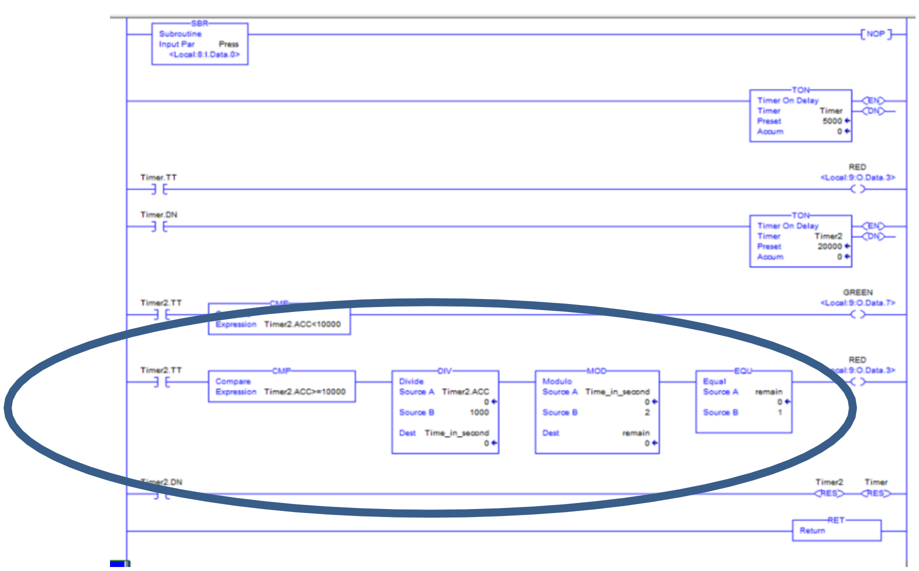


Figure 10-8 Light blinking routine

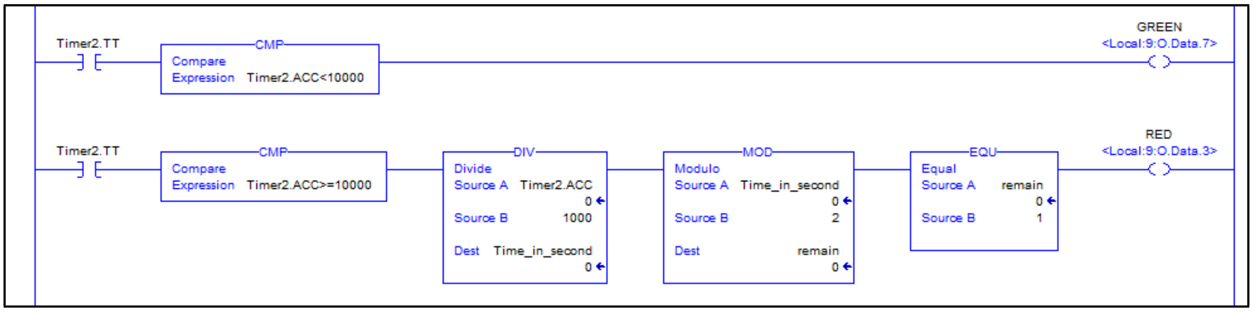
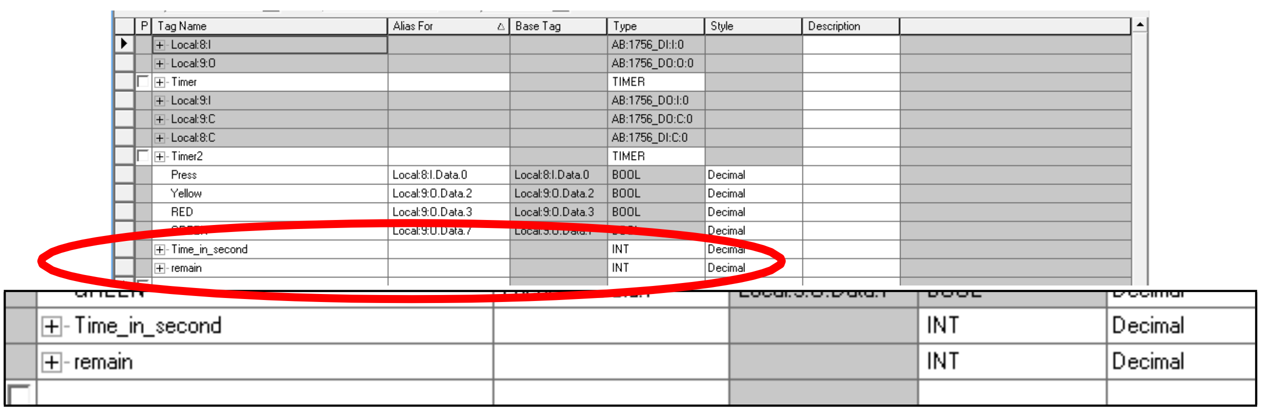


Figure 10-9 Blinking light timer

- These two rungs can make sure green light can stay on for 10s, and red light signal start blinking after 10s (second rung). The **MOD** function is used to check if the ACC number of the time is odd or even number, and make the red signal on if the ACC is odd number.

- Also we need to define new parameters in the control tags as following figures. Pay attention to the data type of new parameters.



P	Tag Name	Alias For	Base Tag	Type	Style	Description
	+ Local:8:I			AB:1756_DI:I:0		
	+ Local:9:O			AB:1756_DO:O:0		
	+ Timer			TIMER		
	+ Local:9:I			AB:1756_DO:I:0		
	+ Local:9:C			AB:1756_DO:C:0		
	+ Local:8:C			AB:1756_DI:C:0		
	+ Timer2			TIMER		
	Press	Local:8:I.Data.0	Local:8:I.Data.0	BOOL	Decimal	
	Yellow	Local:9:O.Data.2	Local:9:O.Data.2	BOOL	Decimal	
	RED	Local:9:O.Data.3	Local:9:O.Data.3	BOOL	Decimal	
	GREEN	Local:9:O.Data.7	Local:9:O.Data.7	BOOL	Decimal	
	+ Time_in_second			INT	Decimal	
	+ remain			INT	Decimal	

Figure 10-10 Expanded view of tag name dialog box

CONCLUSION:

This page is left intentionally blank.

Name: ___________________ Section: _________ Date: ___________

Print where we can read it!

This is an individual assessment.
Each student must turn in this page for lab credit.

Lab 10 Quiz

1. The subroutine will have ____________.
   a. a RET command to return to the main routine
   b. a reset to return to the routine
   c. to complete the program scan before returning to the main routine
   d. limited functions in a program routine

2. A subroutine is ________________.
   a. used to stop the program scan
   b. used only once during a program scan
   c. a complete routine with a beginning and end
   d. designed for a one-time application

3. A jump (JMP) command can _______.
   a. be used to skip portions of a routine
   b. cannot go backwards
   c. use different 3-digit codes to be effective
   d. provide a temporary end to the program scan

4. The return (RET) at the end of the subroutine returns the program scan to __________.
   a. the main routing line that has the JMP instruction
   b. the rung just below the main routine line that has the JMP instruction
   c. the rung just above the main routine line that has the JMP instruction DN bit to be set
   d. the beginning of the subroutine

5. Subroutines are used to ________.
   a. expand the ladder logic design
   b. simplify the coding process and reduce the number of rungs
   c. skip program components
   d. limit the ladder logic design

This page is intentionally left blank.

11 – Analog Inputs/Outputs

Name (Last, First):	Grade Distribution:
____________________ , ___________________	41. Lab Work (0 – 40 pts.) _______
____________________ , ___________________	42. Lab Report (0 – 40 pts.) _______
____________________ , ___________________	43. Quiz (0 – 10 pts.) _______
	44. Safety Practice (0 – 10 pts.) _______
	Total _________
Section Number: IDIS400 ___________	Note: 50 pts. for lab work if no quiz is given.

Objectives:

At the end of this lab you will be able to:

1. Configure Analog input/output modules
2. Integrate simple Analog inputs and outputs in a ControlLogix® 5550.

Caution:

Voltage: 24 V
Current: 10 A
Each test bench has equipment stored on upper shelves. If you notice equipment near the edge, please move it so that it does not fall.

Procedure A: Configuring analog I/O

In this procedure you will configure the analog input (1756-IF8) and analog output (1756-OF8) modules in the ControlLogix® 5550 system.

1. Start the RSLogix™ 5000 software and create a **New** project file. Don't forget to name the controller and designate the correct slot that it occupies in the chassis.
2. Right click on "**I/O Configuration**" in the controller organizer.
3. Select "**New Module**"… and select the correct module. To reduce the length of the list to choose from, try the following:
   a. Click "**clear all**"
   b. Check the **Analog Box**
4. Select the appropriate analog input module from the list. The proper number can be seen in the inside front cover of the module (assuming someone installed the correct sticker on the panel) or on the side of the module (if you open the module cover).
5. Select the correct slot number and enter an appropriate name and slot number for the module.
6. Repeat the above to configure the analog output module.
7. Right Click on **Analog Input Module**, Select Properties.
8. Click Configuration.
   a. Set Input Range → -10V to +10V
   b. Set High Signal value to **+10**
   c. Set High Engineering value to **+10**
   d. Set Low Signal value to **0**
   e. Set Low Engineering value to **-10**
9. Repeat steps 7 & 8 to configure the analog output module.
10. Lastly, configure the AC Input and Relay output modules needed to utilize the pushbuttons and lights.

Procedure B: Designing the Ladder and Tags

There are other properties that probably can be set on the modules, but we will use the default settings and write a simple program.

1. Write a ladder diagram to monitor the level of the fluid in a tank as follows:
   a. An operator starts a pump by momentarily pressing a pushbutton. The pump will continue to pump water into the tank until the level sensor indicates 750 gallons (indicated by 7.5 volts from the sensor). The water level of the tank is reported on an analog gauge.

   b. The operator start button is a momentary pushbutton, a pilot light must be lit whenever the pump is operating, and the level sensor is simulated by turning a knob (tied to memory location:5:I.Ch0Data) that represents the rising water level. The water level is reported on an analog gauge (ties to memory location local:6:O.Ch0Data) that ranges from -10 volts to 10 volts.

   [**Note**: the inputs from the water level sensor and the output to the gauge are analog so bit instructions cannot be used. The **MOV** instruction can be used to copy the sensor reading to the output gauge tag.]
2. Before you can enter the program on the ControlLogix® 5550 controller, you must create all the appropriate tags ("start", "pump", "sensor" and "gauge"). Don't forget to ALIAS them to the **local:5:I.Ch0.Data** and the **local:6:O.Ch0Data** for the input and output, respectively. Of course, the discrete tags are aliased to tags associated with the discrete input and output modules.

3. Enter the program in RSLogix™ 5000 and download it to the ControlLogix® 5550…run the program.

4. Monitor the ladder diagram and observe the values of the analog input and analog output as you increase/decrease the analog dial…it should range from -10 to 10 (it might have some other bipolar response like from -5 to 5 Volts. If it does then you will have to compensate for it on the "**configuration tab**" as is demonstrated in the next procedure). The output analog gauge should range from -10 volts to 10 volts.

5. Demonstrate the working program to your T.A.

Procedure C: Scaling the analog output

In this procedure you will change the configuration settings on the input module and observe the effect.

1. Note that in the previous example, turning the analog dial fully CCW resulted in a minimum reading of -10 volts. This would say that the tank is not only empty, but has a negative amount in it. That doesn't make sense.
2. Change the online menu back to offline.
3. Right click on the analog input module in the controller organizer window. Select "**properties**".
4. Click the "**configuration tab**".
5. Select the required input range. Change it to **0 to 10** volts. (This is the range of voltages that the module will acknowledge).
6. Change the scaling parameters based on the following:
   a. **High Signal:** specifies how the card is to interpret the high value if the "input range"
   b. **Low Signal:** specifies what value the module will assign to the low value of the "input range"
   c. Example, set the low signal to zero and the low engineering to 0 V the module will interpret a zero value as "zero"]
7. Similarly, change the Low Signal and Low Engineering values to **0.0** for the analog output module.
8. Download the program and monitor the ladder diagram.

9. Vary the analog dial from minimum to maximum values. The tag values should now range from zero to 10 volts. If necessary adjust the “high engineering” value on the configuration tab until you have right calibration.
10. Show the results to your T.A.

Note: the High/Low configuration for the output module could also be adjusted.

WRITE YOUR CONCLUSIONS

This page is intentionally left blank.

Name: ___________________ Section: _________ Date: ___________
Print where we can read it!

This is an individual assessment.
Each student must turn in this page for lab credit.

Lab 11 Quiz

1. Analog signals are __________.
   a. expressed as highs or lows
   b. supplied by pushbutton switches
   c. processed by the AC module
   d. expressed in ranges from ± voltage or current

2. Scaling is required to ________.
   a. strip noise from the input signal
   b. set the limits of the signal
   c. provide an input that the A to D converter can handle.
   d. limit the type of input signal received

3. An analog output is used to__________.
   a. turn on a relay
   b. set panel lights
   c. provide a discrete output
   d. drive an actuator such as a valve

4. An analog input module ________.
   a. sends the analog signal directly to the processor
   b. must be scaled to match the A/D input range
   c. operates on a positive voltage level only
   d. is reset at a specific frequency

5. To change the range of the analog module, go to the ________.
   a. write a limit command
   b. go to set range on the analog module tab
   c. voltage change line on the toolbar
   d. properties in the controller organization tab.

This page is intentionally left blank.

12 – Proximity and Photoelectric Sensors to the PLC (Demo)

Name (Last, First):	Grade Distribution:
____________________ , ____________________	45. Lab Work (0 – 40 pts.) _______
____________________ , ____________________	46. Lab Report (0 – 40 pts.) _______
____________________ , ____________________	47. Quiz (0 – 10 pts.) _______
	48. Safety Practice (0 – 10 pts.) _______
	Total _________
Section Number: IDIS400 ___________	Note: 50 pts. for lab work if no quiz is given.

Objectives:

At the end of the lab you will:

1. Understand the functioning of inductive proximity sensor and reflective photoelectric sensor.
2. Be able to integrate sensors with a PLC.

Caution:

Voltage: 24 V
Current: 10 A
Each test bench has equipment stored on upper shelves. If you notice equipment near the edge, please move it so that it does not fall.

Inductive Proximity Sensor (14 LS in Allen-Bradley™ experiment station)

A proximity sensor is an inductive sensor consisting of a coil, oscillator, detector circuit and solid state output. When power is provided to the sensor, the oscillator operates to generate a high frequency field around the sensor. When a metallic object enters the high frequency field, eddy currents experienced on the surface of the metallic object will cause smaller amplitude of oscillation on the detector causing it to turn **ON** the solid-state output. When the metallic object leaves the sensing area, the oscillator regenerates and hence the sensor returns to its normal state.

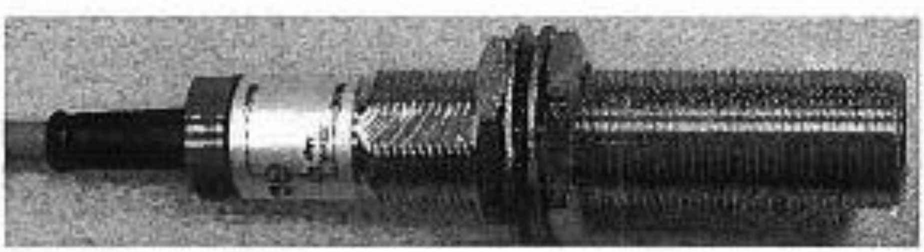


Figure 12-1 Proximity Sensor

Reflective Photoelectric Sensor (14 LS1 in Allen-Bradley™ experiment station)

Photoelectric sensors consist of a light beam transmitter, receiver and a reflector. The sensors use a light emitting diode (LED) for the transmitter, a phototransistor to sense the presence or absence of light, and a reflector to return the light to the phototransistor. The output of the sensor is energized (TURNED ON) when the sensor does not receive or detect the light beam.

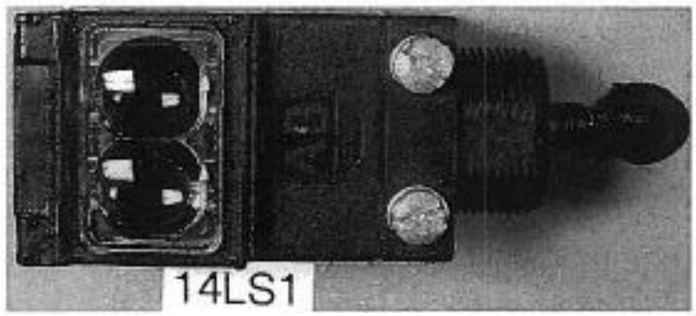


Figure 12-2 Retroreflective Sensor

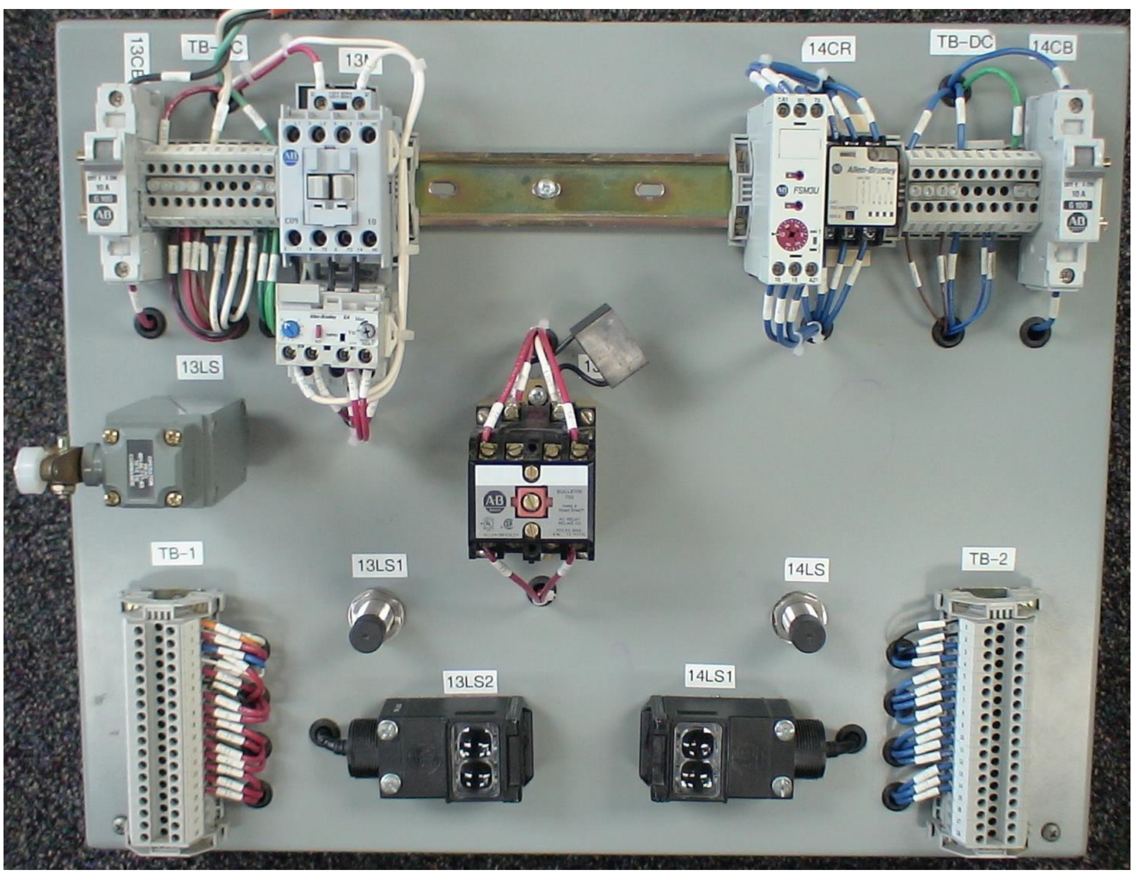


Figure 12-3 Experiment Station

Description of the Allen-Bradley™ (AB) experiment station:

Figure 12-3 shows the layout of the different components on the AB experiment station. The devices mounted are photoelectric and inductive proximity sensors, control relays and limit switches. The devices are powered by either AC or DC depending on their functionality.

In this lab, you are going to use only the sensors which are powered by the DC voltage source. Identify the devices and power supplies before starting the connections.

Trainer identifications

1. TB-DC: Terminal block-DC is used to tap power supply.
2. Identify the +24V and the -24V terminals.
3. 14LS1: Photoelectric sensor which is powered by the DC source.
4. 14LS: Inductive proximity sensor powered by the DC source.
   Note: The internal connections are shown in Figure 12-4.
5. TB-2: Terminal block which enables easy connections for the sensor outputs. The output of the sensors is internally connected to the terminals of TB-2. The output of the proximity sensor is connected to the first terminal (**1**) on TB-2 as shown in the schematic diagram.

Schematic Diagram of the internal connections for the sensors:

Note: Figure 12-4 shows the internal connections for the photoelectric (14LS1) and the proximity (14LS) sensors.

1. The proximity sensor is powered by 24V DC supply. The brown wire connects "+24V" and the blue wire connects to the "-24V" from the TB-DC to the sensor. The output of the sensor is connected to the first terminal (**1**) on TB-2 by a black wire.

2. The photoelectric sensor is powered by 24V DC supply. The brown wire connects "+24V" and the blue wire connects "-24V" from the TB-DC to the sensor. The output of the sensor is connected to the second terminal (**2**) on TB-2 by a black wire.

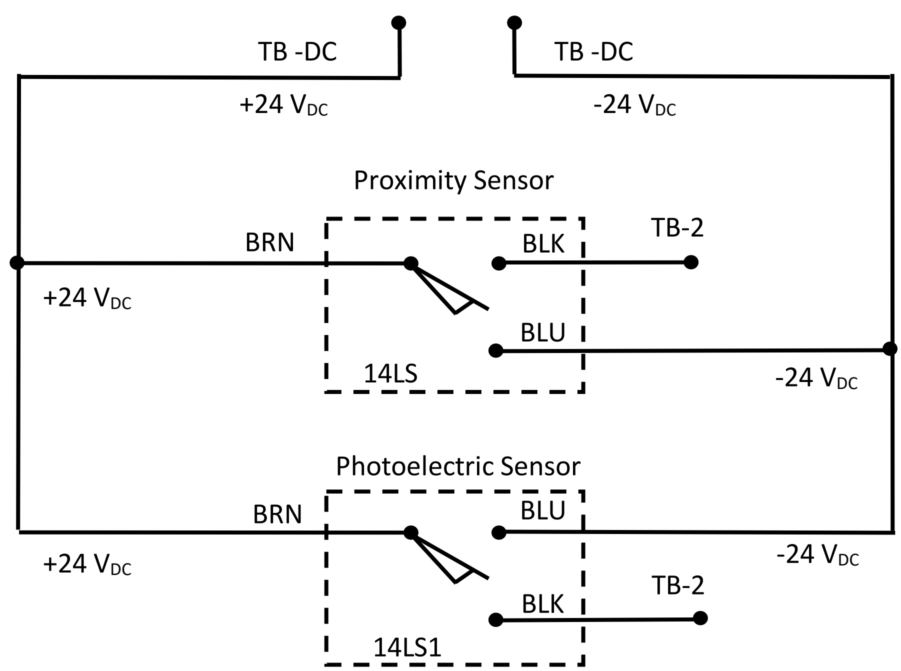


Figure 12-4 Experiment Station Schematic

Integrating the sensor with the PLC (ControlLogix®™5550)

Larger PLCs typically have separate input modules to physically connect the PLC to a sensor. To facilitate easy wiring, the ControlLogix®™5550 modules by AB have a removable terminal block (RTB). The field devices are connected to the RTB which simply snaps into the front of the module. With this system, modules can be replaced without redoing any of the wiring.

Procedure A:

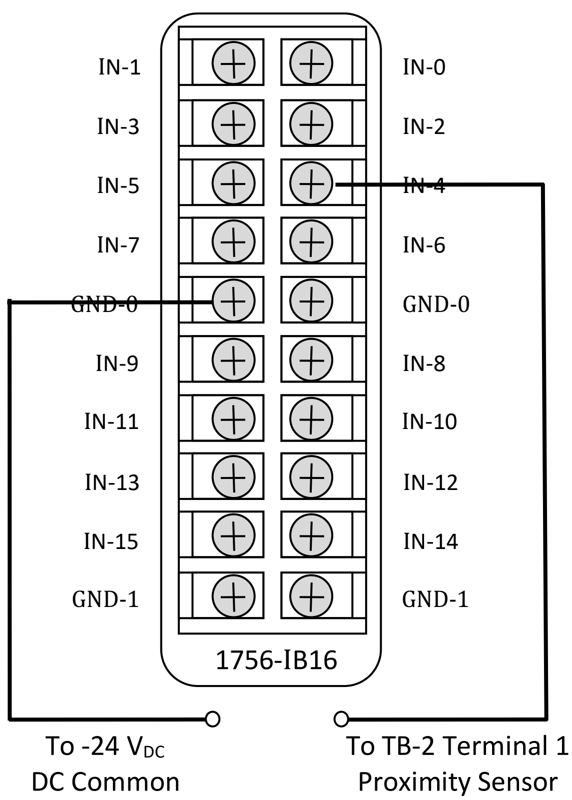
In this procedure the properties of the Allen-Bradley™ proximity sensor will be investigated by connecting the sensor to the PLC. The Allen-Bradley™ proximity sensor is pre wired (power supply is pre-connected) and the output of the sensor is connected to TB-2, terminal 1 on your Allen-Bradley™ experiment station.

1. Identify the inductive proximity sensor (14LS) and terminal block (TB-2) on the Allen-Bradley™ experiment station.
2. Identify the DC terminal block (TB-DC) on the top right corner of the Allen-Bradley™ experiment station.
3. Follow these steps to connect the sensor to the PLC.

Note: Always make sure you power OFF the PLC and the Allen-Bradley™ experiment station when making connections.

- Connect a wire from Terminal 1 of TB-2 to IN-4 (one of the inputs of the DC input module 1756-IB16 of the PLC).

- Connect a wire from -24V DC terminal block TB-DC to the GND-0 pin of the DC input module of the PLC.



12-5 RSLogix™ 5000 DC input Module

4. Open a new project file in RSLogix™ 5000 and configure the controller as you do in all the labs.

5. The sensor signals will be read into the PLC through the DC Input Module. Follow these steps to configure the DC input module of the controller.

 - Right click "I/O configuration" folder in the controller organizer, choose "New Module".
 - Select the DC input module **1756-IB16** and click OK. The module properties window will pop up, in which you need to enter the following information.
 - Name: DC_Input_Module Slot Number: 4
 - Click finish when done.

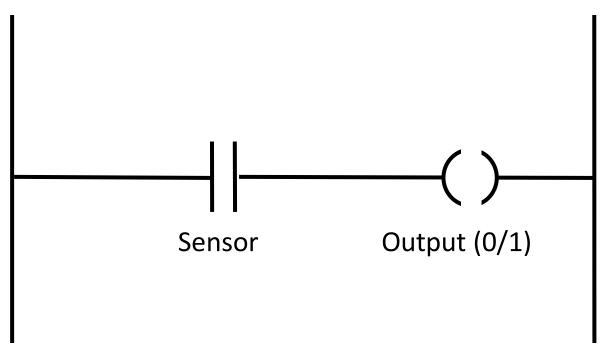
6. Program the ladder diagram for the problem statement given below to test the sensor: Design a ladder diagram for this system:
   a. A stacker arm (S) stacks metal sheets at position A. When an inductive proximity sensor (IN_4) senses a metal sheet, the stacker arm (S) stacks the sheet at position A. The stack height is controlled by a PLC counter.
   b. When 10 parts are stacked, the conveyor (OUT_1) moves the stack to position B. A limit switch IN_6 at position B is used to stop the conveyor at position B.
   c. Then, at B, a sealer is applied to the stacked metal sheets for 14.5 seconds. Turn ON pilot light OUT_2 while the seal is being applied.
   d. After sealing is complete, the entire system is manually reset by a pushbutton IN_3. Assume that there is only one stack on the conveyor at a time.

7. Now create the tags and define the aliases that are required for your ladder diagram. Example of creating a tag for the proximity sensor input:

Tag Name	Alias for
Sensor	Local:4:I.Data.4 (IN_4)

Note: IN_4 corresponds to the data input pin 4 on the DC input module.

Example: Reading the sensor input



12-6 Sensor I/O Configuration

Sensor is an alias to IN_4 (input data pin 4 of the DC input module). When a metal object is placed in front of the sensor, the sensor is activated and pilot light O/1 turns **ON**.

8. Enter the ladder diagram for the problem statement in the main routine.

9. Download the project to the controller and put the controller in **RUN** mode.

10. Switch **ON** both the circuit breakers (13 CB and 14 CB) to energize the Allen-Bradley™ experiment station. Turn the PLC on and test the program.

11. Choose a metallic object (ferrous) and pass the object in front of the sensor. Note the following information. What is the range of detection if the object is directly in front of the sensor? (____inches)

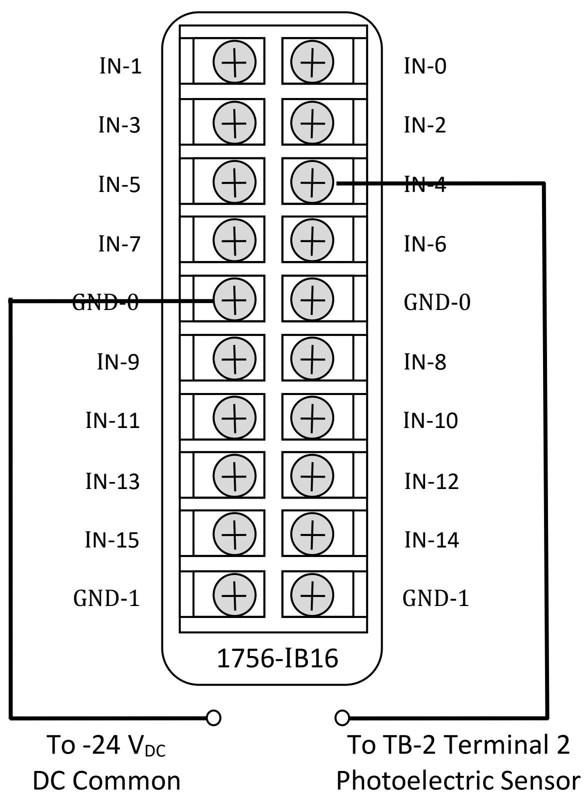
12. Choose a plastic object and pass the object in front of the sensor. Note the following information: What is the range of detection if the object is directly in front of the sensor? (_______inches)

Procedure B: Photoelectric Sensor

In this procedure, the properties of the Allen-Bradley™ photoelectric sensor will be investigated by connecting the sensor to the PLC. The procedure is very similar to Procedure A except that the proximity sensor is replaced by the photoelectric sensor. The Allen-Bradley™ photo electric sensor is pre wired (power supply is pre-connected) and the output of the sensor is connected to TB-2, terminal 2 on your Allen-Bradley™ experiment station.

1. Identify the inductive proximity sensor (14LS) and terminal block (TB-2) on the Allen-Bradley™ experiment station.
2. Be sure that the power is **OFF** the PLC and the Allen-Bradley™ experiment station.
3. Identify the DC terminal block (TB-DC) on the top right corner of the Allen-Bradley™ experiment station.
4. Follow these steps to connect the sensor to the PLC.
   a. Connect a wire from Terminal 2 of TB-2 to IN-4 (one of the inputs of the DC input module 1756-IB16 of the PLC).
   b. Connect a wire from -24V DC terminal block TB-DC to the GND-0 pin of the DC input module of the PLC.

Note: Refer to the figure below.



12-7 Photoelectric Sensor Connections

5. Program a ladder diagram for the problem statement below to test the photoelectric sensor:
    a. A restaurant has two doors Door A and Door B and a maximum capacity of 15 people at any time. Both the doors are initially open. People entering the restaurant through Door A and Door B are simulated by the photoelectric sensor (IN_4) and a manual pushbutton (IN_1) respectively.
    b. A maximum of 5 people can enter through Door B after which it should close. Otherwise, Door B can remain open until the maximum capacity of the restaurant is reached. Pilot light OUT_2 turns ON and remains ON until the maximum capacity of Door B is reached.
    c. Both the doors are automatically shut when the maximum capacity of the restaurant is reached. Simulate the scenario with a reset functions that clears all counters, timers etc.
6. Program the ladder diagram by replacing the previous ladder logic with the new design from step 5.
   Note: Follow steps 7 to 9 from Procedure A to create aliases. Enter the ladder diagram and download the program to the controller.
7. Switch **ON** both the circuit breakers (13 CB and 14 CB) to energize the Allen-Bradley™ experiment station.
8. Pass your handover the sensor and verify the program as the sensor detects the hand.

WRITE YOUR CONCLUSIONS

This page is intentionally left blank.

Name: ___________________ Section: _________ Date: ___________

Print where we can read it!

This is an individual assessment.
Each student must turn in this page for lab credit.

Lab 12 Quiz

1. The proximity sensor is used to __________.
   a. detect presence of metal in the magnetic field
   b. detects the presence of light in its field
   c. identify and count wooden blocks
   d. provide analog signals to the processor

2. A unique feature of the reflective sensor is that it ________.
   a. needs a reflector to reflect its LED signal back to the detector
   b. used the reflection of an object to detect presence
   c. turns off if the light beam is broken
   d. uses a magnetic field to sense the presence of an object

3. A photoelectric sensor is used to ________.
   a. provides images of people passing past it
   b. detect presence by a light beam reflected from the target
   c. has a long range of detection
   d. sense intense lighting conditions

4. The proximity sensor uses ____________.
   a. external AC power to operate
   b. WiFi to connect to the PLC
   c. the analog input module for connections
   d. the DC module for connections

5. The photoelectric sensor uses ________.
   a. external AC power to operate
   b. WiFi to connect to the PLC
   c. the analog input module for connections
   d. the DC module for connections

This page is intentionally left blank.

13 – Variable Frequency drives

<table>
<tr>
<td>
Name (Last, First):
____________________ , ____________________
____________________ , ____________________
____________________ , ____________________
Section Number: IDIS400 ___________
</td>
<td>
Grade Distribution:
49. Lab Work (0 – 40 pts.) ______
50. Lab Report (0 – 40 pts.) ______
51. Quiz (0 – 10 pts.) ______
52. Safety Practice (0 – 10 pts.) ______
Total _________
Note: 50 pts. for lab work if no quiz is given.
</td>
</tr>
</table>

Caution:

Voltage: 24 V
Current: 10 A
Each test bench has equipment stored on upper shelves. If you notice equipment near the edge, please move it so that it does not fall.

What do we do in this lab? - We control the speed and direction of a 3 Phase AC (Induction) motor using a PLC.

How is it done? –

The Motor is driven using a Variable frequency Drive. The Drive is connected to the PLC through the DeviceNet module. We configure and program the PLC to run the motor.

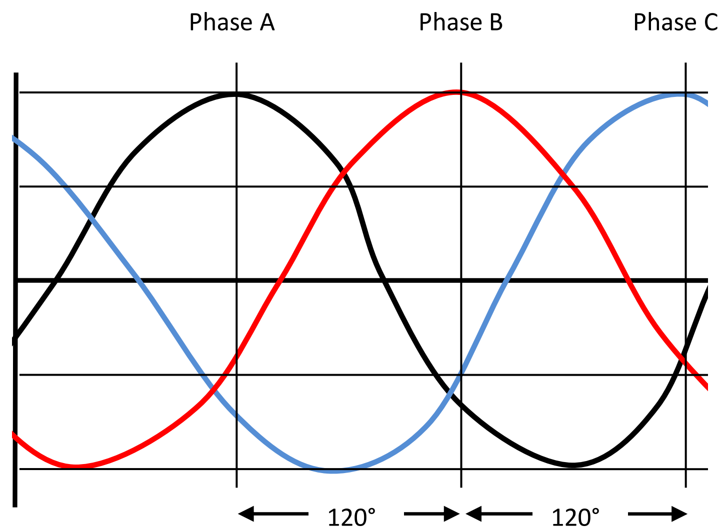
What is a 3 Phase AC Motor*?

An **induction motor** (or **asynchronous motor** or **squirrel-cage motor**) is a type of alternating current motor where power is supplied to the rotor by means of electromagnetic induction.

An electric motor converts electrical power to mechanical power in its rotor (rotating part). There are several ways to supply power to the rotor. Induction motors are now the preferred choice for industrial motors due to their rugged construction, absence of brushes (which are required in most DC motors) and — thanks to modern power electronics — the ability to control the speed of the motor is possible.

A three-phase induction motor has a simple design. It has inherently high starting torque, and high efficiency. Such motors are applied in industry for pumps, fans, blowers, compressors, conveyor drives, and many other kinds of motor-driven equipment. A three-phase motor is more compact and less costly than a single-phase motor of the same voltage class and rating; and single-phase AC motors above 10 HP (7.5 kW) are uncommon. Three-phase motors also vibrate less and hence last longer than single-phase motors of the same power used under the same conditions. **Three-phase** electric power systems have at least three conductors carrying voltage waveforms that are 2π/3 radians (120°, 1/3 of a cycle in-phase) offset in time.

One voltage cycle of a three-phase system, labeled 0 to 360° (2 π radians) along the time axis. The plotted line represents the variation of instantaneous voltage (or current) with respect to time. This cycle will repeat 50 or 60 times per second, depending on the power system frequency. The colors of the lines represent the American color code for 120v three-phase. The standard is black for V_{L1}, red for V_{L2} and blue for V_{L3}.



13-1 Three-phase AC Voltage

How do we control the speed of an AC Motor?

The speed of an AC motor can be controlled by varying the Frequency of the 3 phase current supplied to it. The speed of an AC motor is determined by the following formula:

$$v = \frac{120 \times f}{p}$$

Where p= number of magnetic poles in the motor.

For example a 6 pole motor operating on 60Hz power would have speed:

$$v = \frac{120 \times f}{p} = 1{,}200 \text{ rpm}$$

Where *v* is the speed of the rotor (in rpm), *f* is the frequency of the AC supply (in Hz) and *p* is the number of magnetic poles.

Variable-frequency drives

A variable-frequency drive (VFD) is a system for controlling the rotational speed of an alternating current (AC) electric motor by controlling the frequency of the electrical power supplied to the motor. A variable frequency drive is a specific type of adjustable-speed drive. Since the voltage is varied along with frequency, these are sometimes also called VVVF (variable voltage variable frequency) drives. Variable-frequency drives are widely used. In ventilation systems for large buildings, variable-frequency motors on fans save energy by allowing the volume of air moved to match the system demand. They are also used on pumps, conveyor and machine tool drives.

When an induction motor is connected to a full voltage supply, it draws several times (up to about 6 times) its rated current. As the load accelerates, the available torque usually drops a little and then rises to a peak while the current remains very high until the motor approaches full speed.

By contrast, when a VFD starts a motor, it initially applies a low frequency and voltage to the motor. The starting frequency is typically 2 Hz or less. Thus starting at such a low frequency avoids the high inrush current that occurs when a motor is started by simply applying the utility (mains) voltage by turning on a switch. With a VFD, the stopping sequence is just the opposite as the starting sequence. The frequency and voltage applied to the motor are ramped down at a controlled rate. When the frequency approaches zero, the motor is shut off. A small amount of braking torque is available to help decelerate the load a little faster than it would stop if the motor were simply switched off and allowed to coast.

The VFD that we use here is *PowerFlex® 70* by Allen Bradley. This Drive can be start up the motor in two ways. One way is the Assisted Start up. This requires two levels of information – Basic and Detailed Information about different start up parameters and I/O parameters and Used only for enhanced control. The other way is the S.M.A.R.T startup. This start up routine simply requires minimal of 5 different parameters.

Operating a variable-frequency drive

With the following parameters, you can set the following functions:

S - Start Mode and Stop Mode (Assisted Start or SMART Start)

M - Minimum and Maximum Speed (Specified in Hz)*

A - Acceleration Time and Deceleration Time (Time taken to accelerate to required speed and vice versa, given is Seconds.)*

R - Reference Source

T - Thermal Motor Overload *-The Parameters that we will change for the purpose of this exercise.

Let us become familiar with the PowerFlex® 70. The following diagram shows the PowerFlex® 70.

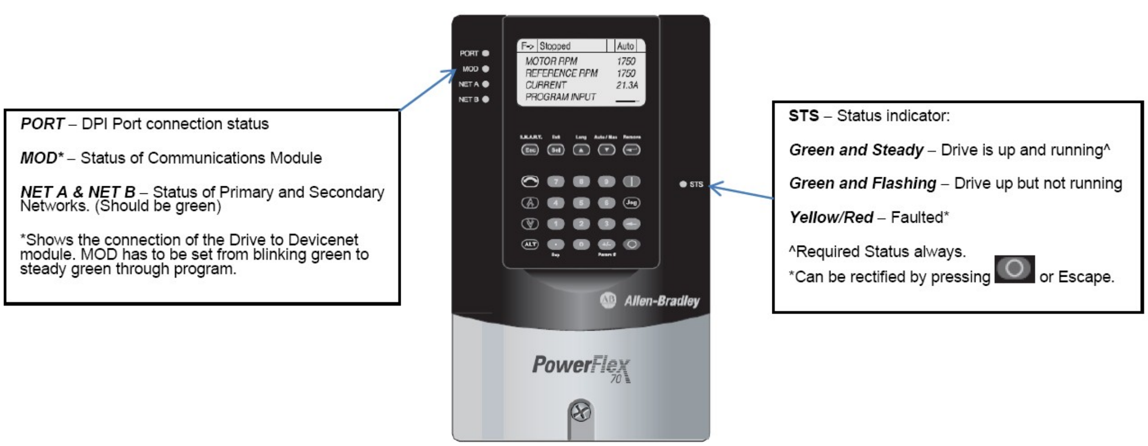


Figure 13-2 PowerFlex® 70 variable speed drive

Let's see if the motor runs, Press 'JOG'. This jogs the motor at 10 Hz. And this is used to test if the motor runs.

Run the motor directly from the PowerFlex® 70 drive by changing the two parameters mentioned above. Let us run the Motor with a minimum frequency of 10Hz and a maximum frequency of 40Hz. Let the Acceleration and Deceleration times be 5 seconds each. The following steps are to be followed in order to set the drive to these parameters and start up.

Procedure

- Press **ALT -> ESC**. This loads the S.M.A.R.T Menu.
- Use the arrow keys and press **enter** at the respective menu items and enter the given values.
- Press **ESC** and then press the green button above '**JOG**' to start the motor.
- Press the **Direction Change** button (Left of 7) to change direction.
- Press the **red** button to stop the motor.

EXERCISE:

Program the drive to minimum of 10 Hz and maximum of 50 Hz and an Acceleration time of 3 seconds and a Deceleration time of 7 seconds.

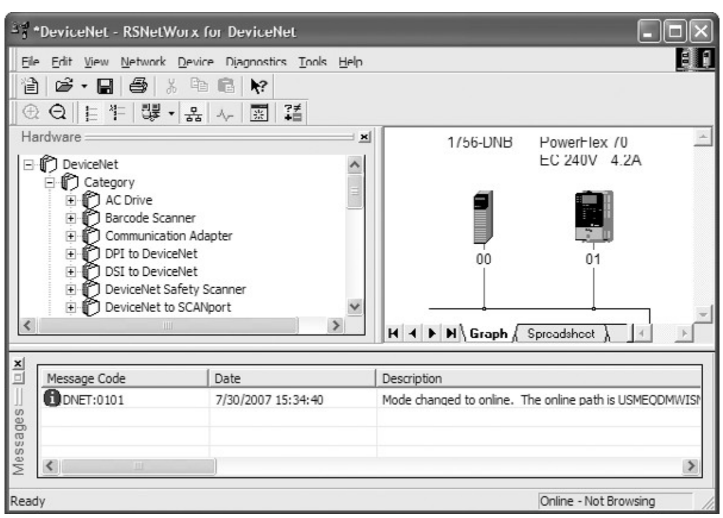
Controlling the PowerFlex® VFD through the PLC via the Devicenet module:

DeviceNet: DeviceNet is a communication protocol used in the automation industry to interconnect control devices for data exchange. It uses Controller Area Network as the backbone technology and defines an application layer to cover a range of device profiles. Typical applications include information exchange, safety devices, and large I/O control networks.

The DeviceNet for the PLC is configured as follows:

1. *Click on **START -> RSNetWork for DeviceNet**.* (This software is used to detect the components connected to DeviceNet as a Network)
2. *Click on **NETWORK -> GO ONLINE*** and expand the controller network (the way you do in RSLogix5000 when you 'Set Project Path'). This time expand to get DeviceNet. Click on the **DeviceNet** and click O.K.

   Note: The software will browse the devices connected to the controller network and return the following screen. You should get a DeviceNet Scanner (1756-DNB) and the PowerFlex® 70 drive on the graph. Don't mind the slot numbers.



13-3 DeviceNet™ screen view

3. Now Right Click on 1756-DNB and click on properties. Go to Module tab and Change the Slot to **3**.
4. Go to **FILE-> SAVE AS** and give a separate filename and save it in any location.

Note: The configuration of the network from DeviceNet™ is complete.

5. Open RSLogix™ 5000.
6. Follow the standard configuration for adding AC input (1756-IA16) and AC output (1756-OW161) modules to complete the set-up of the PLC.
7. Right click on I/O Configuration and add a new module.
8. Select the 1756-DNB DeviceNet™ Scanner and add it to slot 3. Click on next twice, in the next screen add the DeviceNet™ network that you saved to the properties and click apply. The Configurations of the DeviceNet™ is now complete.

Logic Command Word and Speed Reference

Next we need to control the motor - Start, Stop, Direction control, etc. We can achieve that setting certain bits to a value of 1. The Logic command word used to control the motor is the first 16 bits of the word **Local:3:O.Data[0].** However for our purpose of programming, we will use only the first six bits (0-5 Bits) which are represented as follows.

Logic Command Word

Logic Bits																	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Command	Description
															x	Stop (1)	0 = Not Stop 1 = Stop
														x		Start (1)(2)	0 = Not Start 1 = Start
													x			Jog	0 = Not Jog 1 = Jog
												x				Clear Faults	0 = Not Clear Faults 1 = Clear Faults
										x	x					Direction	00 = No Command 01 = Forward Command 10 = Reverse Command 11 = Hold Direction Control

Table 13-1 Logic command word

The above table indicates that if we set ***Local:3:O.Data[0].1*** to 1, the motor will start running. Likewise we can set the other bits to 1 or 0 respectively and achieve what is required.

The next important issue to note is the ***Speed Reference***. Speed reference is a value that is sent to the Command Data register to set the speed of the motor. The Value is based on the Size of the Command register; in this case, it is 16 bit. 16 bit has valid values from -32768 to 32767. So, In order for the motor to run at a 100% of the rated speed, 32767 is used. Likewise for 50% it is 32767/2 = 16384 and for 25% it is 32767/4 = 8192.

Reference Value	Scale		Output Speed
	Percent	Value	
32767	100%	130 Hz	60 Hz
16384	50%	65 Hz	60 Hz
8192	25%	32.5 Hz	32.5 Hz
0	0%	0 Hz	0 Hz

Table 13-2 Motor speed scale factor

The above table shows the rated speeds of the motor. The actual values of the speed reference are shown. They are not translated into respective output speeds because a parameter (082) in the PowerFlex® 70 has to be changed to full maximum speed.

EXERCISE: *What should be the reference value and output speed for 35% & 65% scaling?*

Reference Value (35%): _______Output Speed (35%): ______________

Reference Value (65%): _______Output Speed (65%): ______________

EXERCISE: *The following is a simple program demonstrates the Start/Stop and Direction control of the motor. Program it and verify the output with the instructor.*

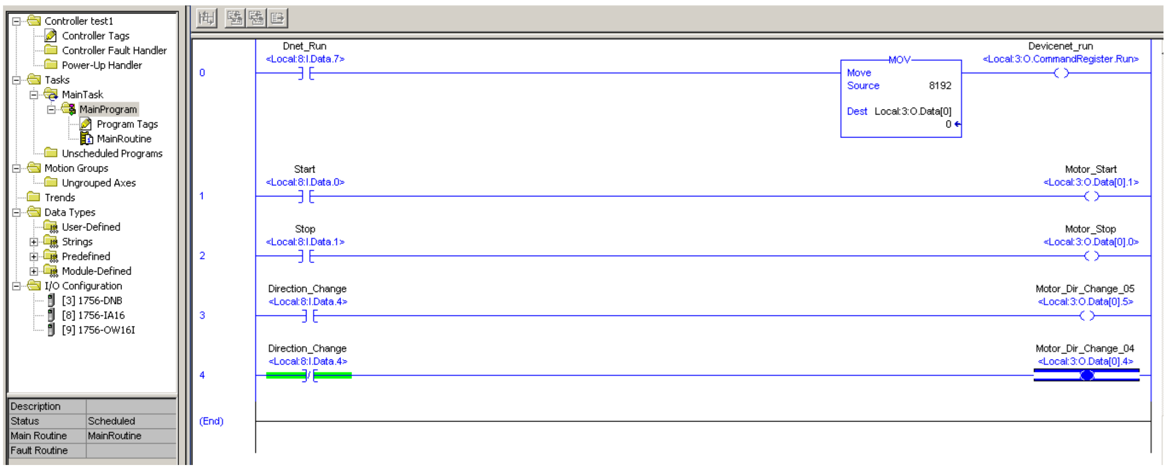


Figure 13-4 Start/Stop ladder logic

Note: Create all the 9 controller tags shown by *Right Click Controller Tags* ***Edit Tags****.*

Rung 0: Switch 7 Moves the Speed reference of 8192 (25%) to command register and puts the Device net, which is at slot 3, into'**Run**' mode.

Rungs 1 & 2: Starts and Stops the Motor.*

Rungs 3 & 4: Switch – IN_4 is used here for direction change. The switch position sets values 0 or 1 to the bits 4 and 5 of the Logic command word. Thus changing the position of the switch when the motor is running will make the motor stop and change directions.*

Download the program, Put the controller in *'RUN MODE'* and then check if the statuses in the drive are all green. Then run the motor.

*Look up the Logic Command word table in the previous page.

WRITE YOUR CONCLUSIONS

Name: ___________________ Section: _________ Date: ___________

Print where we can read it!

This is an individual assessment.
Each student must turn in this page for lab credit.

Lab 13 Quiz

1. What is the reference value for a 75% scale factor?
   a. 32767
   b. 24575
   c. 16383
   d. 3192

2. Which bits are used to set up the PowerFlex® 70?
   a. 11 – 15
   b. 10 – 7
   c. 0 – 5
   d. all 16

3. An asynchronous motor is also called a _____________ motor.
   a. repulsion
   b. DC
   c. induction
   d. servo

4. Communications to the PowerFlex® 70 is through __________.
   a. Ethernet
   b. RS 5000
   c. DeviceNet
   d. ModuleNet

5. The parameters where the reference value is loaded are accomplished by a ________.
   a. ADD command
   b. CMP command
   c. CPT command
   d. MOV command

This page is intentionally left blank.

APPENDIX

Getting Started with PLC Programming

1. **Open the software**
    a. Go to the Windows Start command
        i. Scroll down to RS Software
        ii. Scroll down to RS Logix 5000
    b. Refer to page 29 and check the various toolbars available. Compare them with your software.
2. **Create a New Project file:**
    a. Go to File on the header toolbar
    b. Scroll down to Creating New File
    c. Note: make sure that the revision is 13.
    d. Give a suitable name to your PLC code like "timer-labX"
    e. Choose the folder path "C:\Users\'Your name'\Documents\IDIS_400-'Semester'-'year"
3. **Configure Input and Output Modules:**
    a. Note: The model depends on the kind of task needed. For example: the task is to create AC Input and Output modules.
    b. Create an **Input** module
        i. Go to controller organizer
        ii. Right click on I/O configuration
        iii. Choose new module
        iv. For AC Input module – select 1756-IA16
            1. Note: Keep the major revision as 2
            2. Name it as **IN**
            3. Choose slot 8
            4. Press "Finish"
    c. Create an **Output** module
        i. Go to controller organizer
        ii. Right click on I/O configuration
        iii. Choose new module
        iv. For AC Output module – select 1756-OWA161
            1. Note: Keep the major revision as 2
            2. Name it as **OUT**
            3. Choose slot 9
            4. Press "Finish"
4. **Entering Tag Names, aliases, types:**
    a. Select "Controller Tags" in the Controller Organization menu.
    b. Select "Edit Tag" located at the end of the page.
    c. Press next row to enter a new tag name.
    d. <u>Input Tag</u>
        i. Write **IN_1** in the tag name column
        ii. Use the drop down button to find available module "Local:8:I" for input module.
        iii. Select "Local:8:I:data"
        iv. Select a number assignment from 0 – 3
        v. To save the tag, click on the next blank tag row
    e. Output Tag
        i. Write **OUT_1** in the tag name column
        ii. Use the drop down button to find available module "Local:9:O" for input module.
        iii. Select "Local:9:O:data"
        iv. Select a number assignment from 0 – 3
        v. To save the tag, click on the next blank tag row
5. **Entering a Ladder Diagram:**
    a. In the Controller Organization menu, select "Task" + to expand to the sub-menu, select "Main Task". + to expand the sub-menu and find "Main Program" + to expand the sub-menu and find "MainRoutine". This opens the View Pane to enter the ladder logic.
    b. Draw the ladder diagram using symbols on the ladder edit tabbed toolbar.
    c. Note: if an "e" is located to the left of a rung, there is an error in the rung configuration or in a previous step.
6. Running the Program:
    a. Select the "Communications" tab on the top of the page.
    b. Select "Who Active"
    c. Choose AB_DF1,DF1, press "Go Online"
    d. Select "Download"
    e. Note: make sure that the key switch on the PLC Processor is in the "Run Mode"

ASSIGNMENT – IDIS 400 – DISCRETE INPUT MODULES FOR PLC

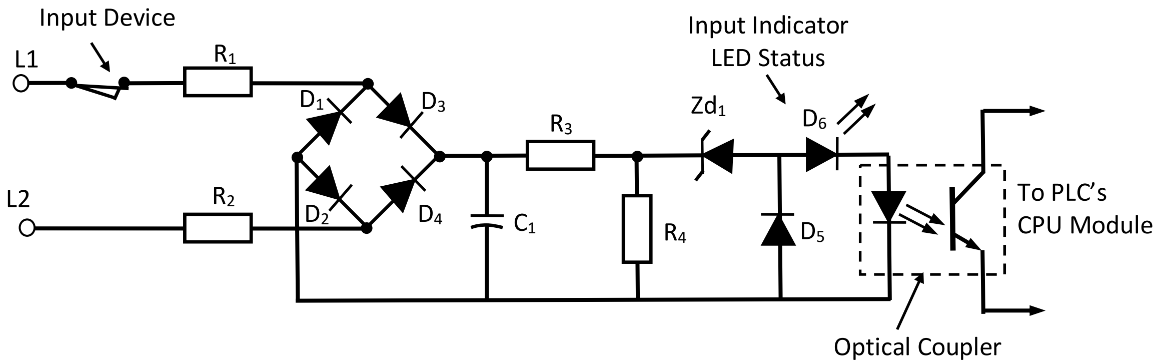


Fig 2-7 from Textbook – AC Discrete Input

1. Explain the function of the **AC** Discrete Input Module
2. Explain what L1 and L2 represent
3. Explain the function of R_1 and R_2
4. Explain the function of the Bridge Rectifier (D_1, D_2, D_3, & D_4)
5. Explain the function of the Filter Network (C_1, R_3, & R_4)
6. Explain the function of the Basic Regulator (Zd_1 & R_4)
7. Explain the function of D_5
8. Explain the function of D_6
9. Explain the function of the OPTO-ISOLATOR
10. Explain the function of the LED within the Optical Isolator
11. Explain the function of the PHOTO-TRANSISTOR within the Optical Isolator

ASSIGNMENT – IDIS 400 – DISCRETE OUTPUT MODULES FOR PLC

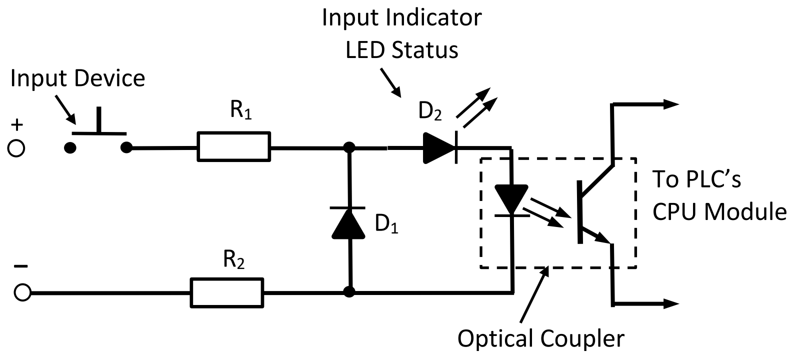


Fig 2-13 from Textbook – DC Discrete Input Module

Explain the function of the **DC** Discrete Input Module

1. Explain what "+" and "-" (at the left of the picture) represent
2. Explain the function of R_1 and R_2
3. Explain why there is no Bridge Rectifier
4. Explain why there is no Filter Network
5. Explain why there is no Basic Regulator
6. Explain the function of D_1
7. Explain the function of D_2
8. Explain the function of the OPTO-ISOLATOR
9. Explain the function of the LED within the Optical Isolator
10. Explain the function of the PHOTO-TRANSISTOR within the Optical Isolator

N.O.-N.C. DISTINCTION

The following strategy can be used to distinguish the use of N.O. and N.C. type of contacts in ladder diagrams:

- First, the —] [— symbol stands for Examine if Closed (XIC) and the symbol stands for Examine if Open (XIO).
- N.O. or N.C. refers to the way in which an input device can be a push button, switch, photo sensor, proximity sensor, etc. or any input device has been wired in the field.
- N.O. or N.C. type of input can be associated with either of —] [— or —]/[— depending on the requirements.
- When a Normally open input is actuated, it closes and when a normally closed input is actuated, it opens up.
- In order to better understand the distinction between the use of N.O. and N.C. type inputs in ladder logic, let us imagine that the input devices are interactive and speak to the instruction that they have been associated with, in the ladder diagram.
- When an XIC is associated with any input in the ladder diagram, it continuously asks the following question to the input with which it has been associated.

  “Are you closed?”

  As long as the answer it receives from the input device is “No”, it blocks the power or cuts off the output device. The moment it receives the answer “Yes” form the input device, it aids in the turning on of the output device.
- Similarly, when, XIO is associated with any input in the ladder diagram, it continuously asks the following question to the input with which it has been associated

  “Are you open?”

  As long as it receives the answer “Yes” form the input device, it aids in the energizing the output device. The moment it receives the answer “No” from the input device, it blocks the power or cuts off the output device.

To better illustrate the above example, let us consider the following example. Consider a case where in an output is turned on when the start switch is actuated and turned off when the stop switch is actuated. Given that both the start and stop switches are wired normally open. The following ladder illustrates this case.

Start- Normally open (N.O.) Stop-Normally open (N.O.)

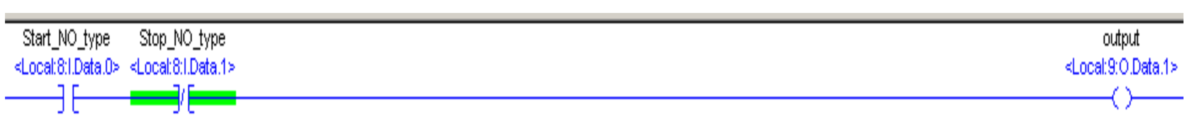


Figure 0-1 Start/Stop logic

The Start switch is associated with XIC instruction in the ladder diagram. Hence, the XIC instruction keeps asking the start switch, the question: “Are you closed?” The normally open start switch when actuated goes into a state of being closed. Consider the following table where in the Start switch is actuated on the seventh time step.

Time step	Question posed to the input device (start switch)	Answer received	State of the output
1	Are you closed?	N	Not energized
2	Are you closed?	N	Not energized
3	Are you closed?	N	Not energized
4	Are you closed?	N	Not energized
5	Are you closed?	N	Not energized
6	Are you closed?	N	Not energized
7 (Start switch actuated)	Are you closed?	Yes	Energized
8	Are you closed?	Yes	Energized
9	Are you closed?	Yes	Energized
10	Are you closed?	Yes	Energized
11	Are you closed?	Yes	Energized

Table 0-1 Examine if closed

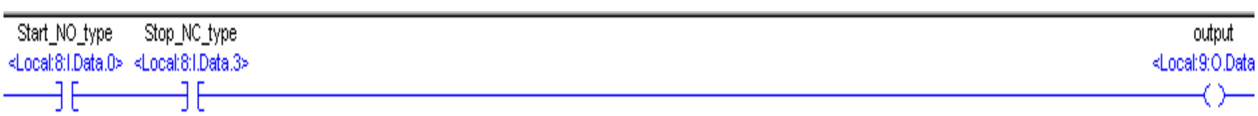
The Stop switch is associated with XIO instruction in the ladder diagram. Hence, the XIO instruction continually asks the stop switch, the question: “Are you open?” The normally open stop switch when actuated goes into a state of being closed. Consider the following table where in the Stop switch is actuated on the time step while the motor is already running (has been started using the Start switch)

Time step	Question posed to the input device (stop switch)	Answer received	State of the output
1	Are you open?	Yes	Energized
2	Are you open?	Yes	Energized
3	Are you open?	Yes	Energized
4	Are you open?	Yes	Energized
5	Are you open?	Yes	Energized
6	Are you open?	Yes	Energized
7 (Stop switch actuated)	Are you open?	No	Not energized
8	Are you open?	No	Not energized
9	Are you open?	No	Not energized
10	Are you open?	No	Not energized
11	Are you open?	No	Not energized

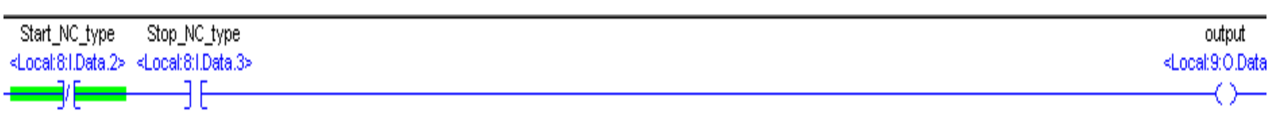
Table 0-2 Examine if open

Similarly, we could have the following ladders for the following combinations of the inputs for the case presented above. Kindly generate the table as done for the above case for each of the following cases.

Start- Normally open (N.O.) Stop-Normally closed (N.C.)



Start- Normally closed (N.C.) Stop-Normally closed (N.C.)



Start- Normally closed (N.C.) Stop-Normally open (N.O.)

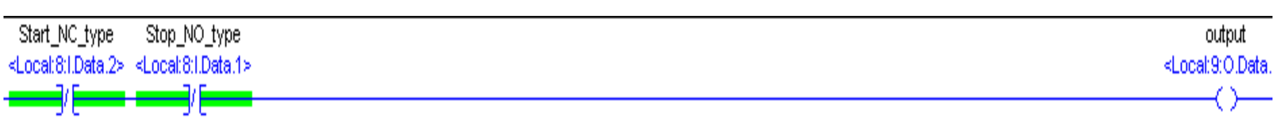


Figure 0-2 Start/Stop options

LORENZO, 2012

EXAMINE ON (XIC):
(True if ON)

1. If input is true, then the output is true.
2. If the input is false, then the output is false.

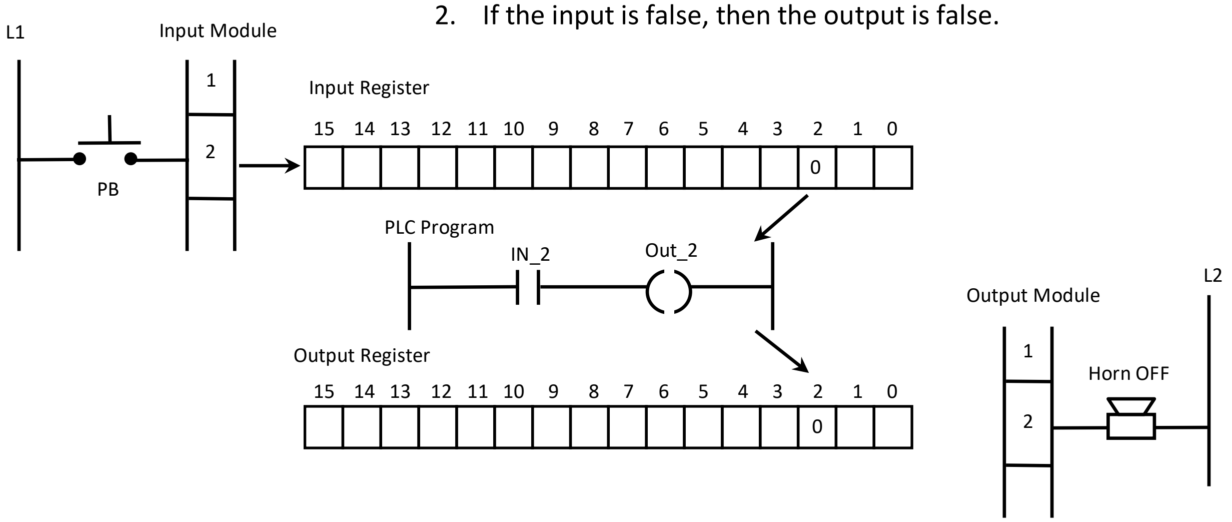


Figure 0-3 Examine if closed connections

EXAMINE OFF (XIO):
(True if OFF)

1. If input is true, then the output is true.
2. If the input is false, then the output is false.

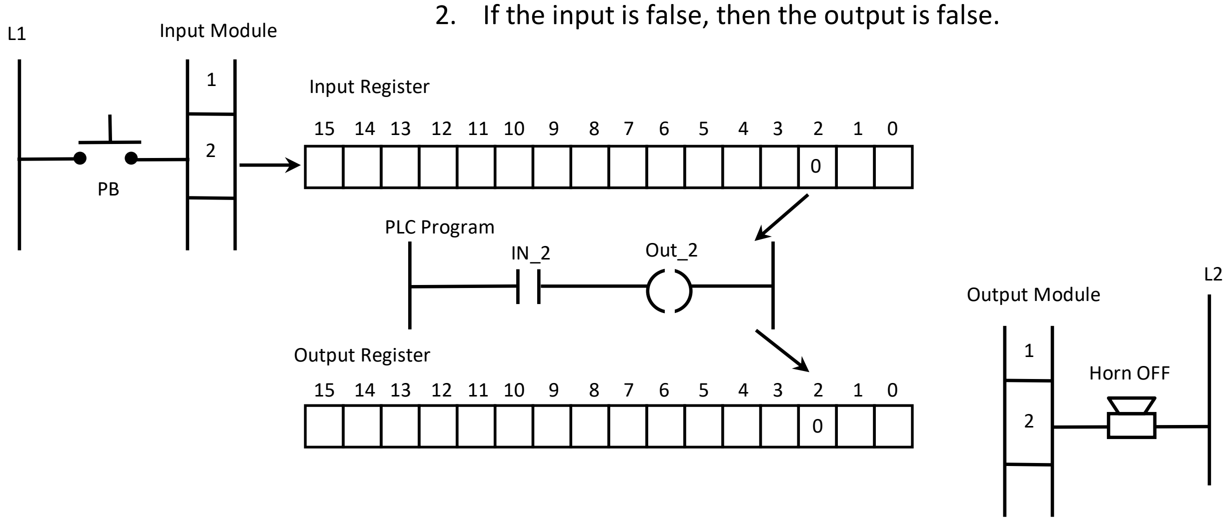


Figure 0-4 Examine if open connections

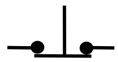
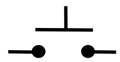
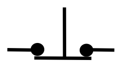
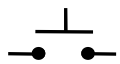
Condition of input device	Program instruction	Logic: True - False
 ON	—| |— "True if ON" "XIC"	T (ON)
 OFF	—| |— "True if ON" "XIC"	F (OFF)
 ON	—|/|— "True if OFF" "XIO"	F (OFF)
 OFF	—|/|— "True if OFF" "XIO"	T (ON)

Table 0-3 Input functions

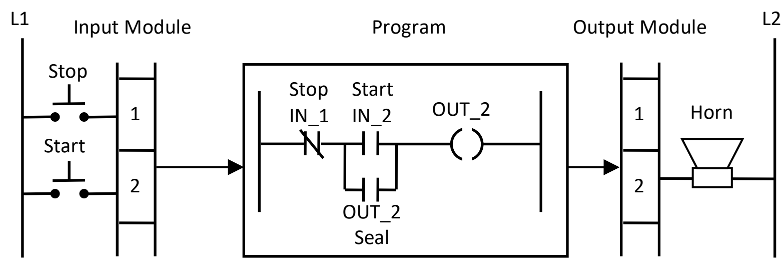
START – STOP – SEAL PROGRAM

LORENZO, 2012

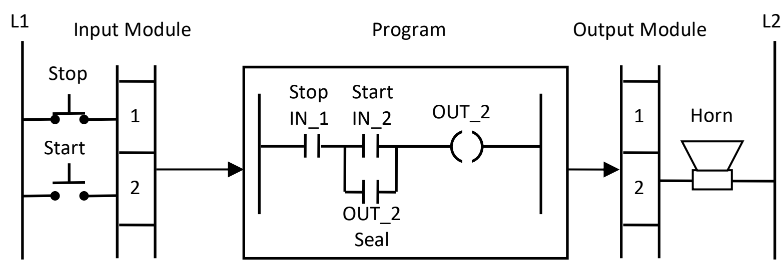
Program a START – STOP – SEAL system to control a warning HORN signal.

Output = OUT_2
Stop = IN_1
Start = IN_2

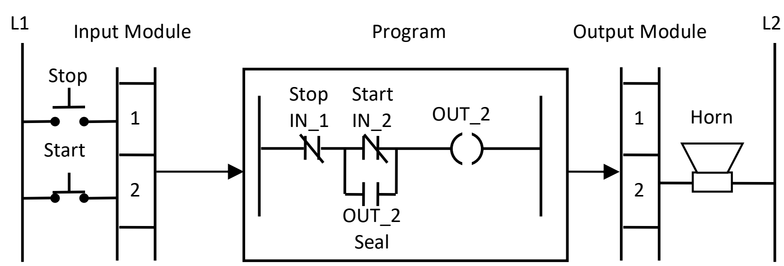
SOLUTION 1:



SOLUTION 2:



SOLUTION 3:



SOLUTION 4:

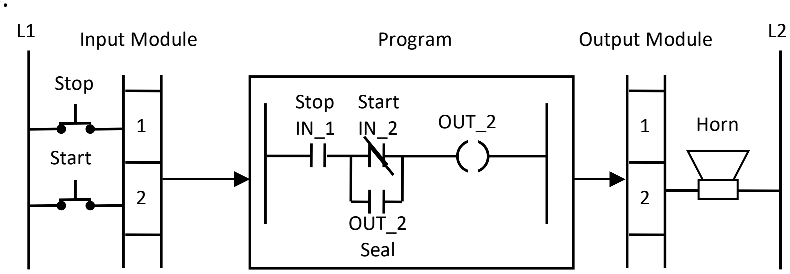


Figure 0-5 Start/Stop Seal options

NOTES:

A. Solution (2) is the industrial standard.

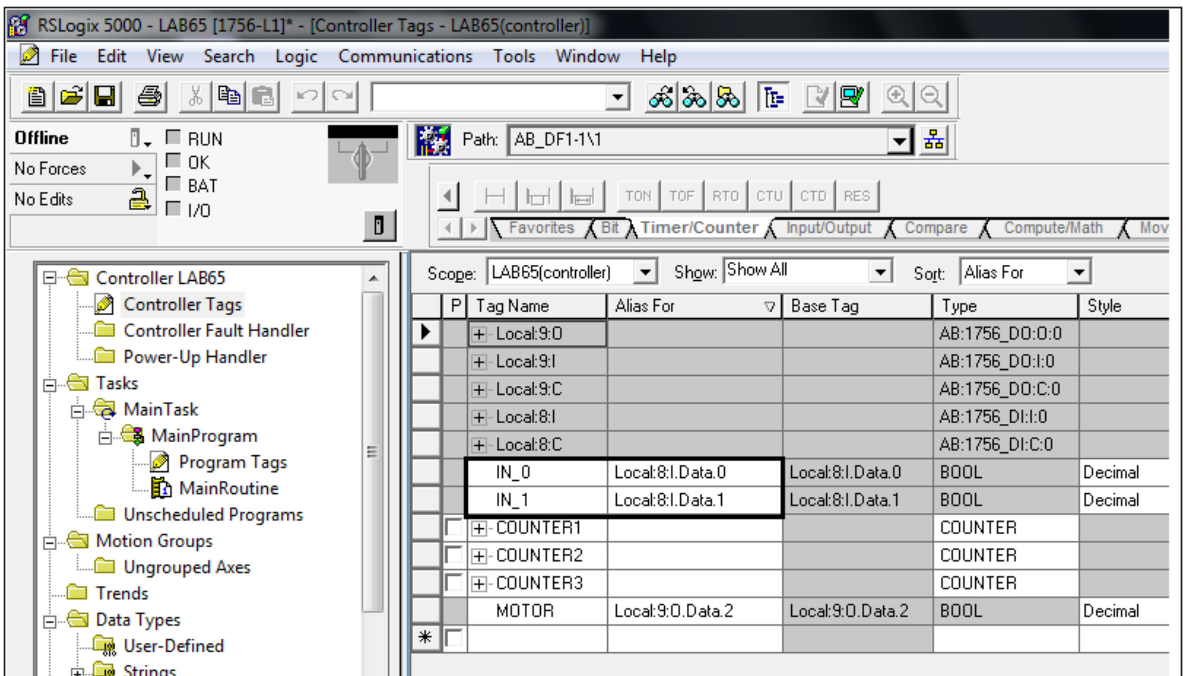
ALLEN BRADLEY CONTROLOGIX ADDRESSING FORMAT

- Chassis reference – Defines whether the address module resides in the local PLC or a remote PLC cabinet.
- Slot location – Location of module in PLC rack.
- Type – Input or Output represented by I or O, respectively.
- Data – Bit type.
- Terminal – Module terminal to which the device is wired.

Local : 8 : I . Data . 0

Addressing can be a tedious operation as the number of inputs and outputs increase. Therefore, tags or textual names are created for each address. Instead of placing the complex address on each input or output of a ladder diagram, the tag is utilized. These steps outline how to create tags for addresses.

Example:



Entering the Tag names, Aliases and Types

Figure 0-6 Tag name screen

Tag Name	**Alias for**
IN_1	Local:8:I.Data.1

IN_1 is the tag name that associates the pushbutton to input 1 in the ladder diagram. It is aliased as **Local:8:I.Data.1** since the input module is located in **slot 8** and **I** indicates that it is an input device. Other possible tags that we can use:

Local:8:I.data.12 may be tagged with: I_12, or I-12, or I/12, or I 12

Local:9:O.data.06 may be tagged with: O_06, or O-06, or O/06, or O 06

ON – DELAY TIMER (TON)

Lorenzo – 2012

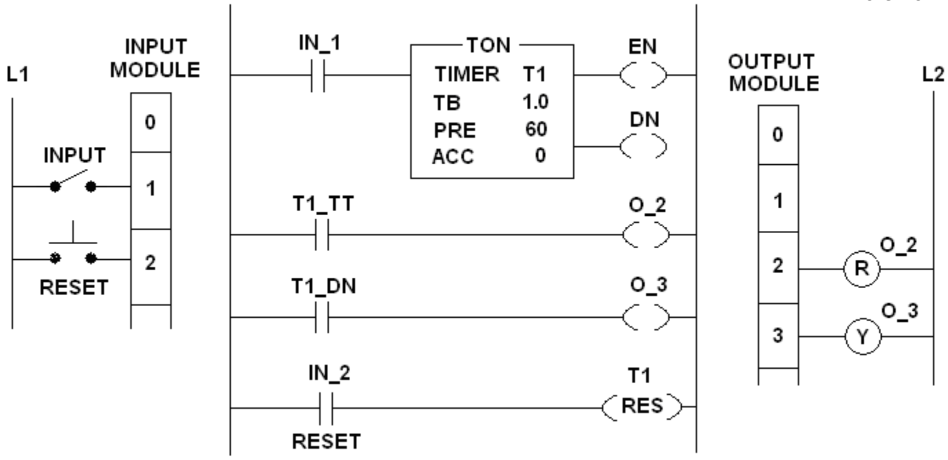


Figure 0-7 TON ladder diagram

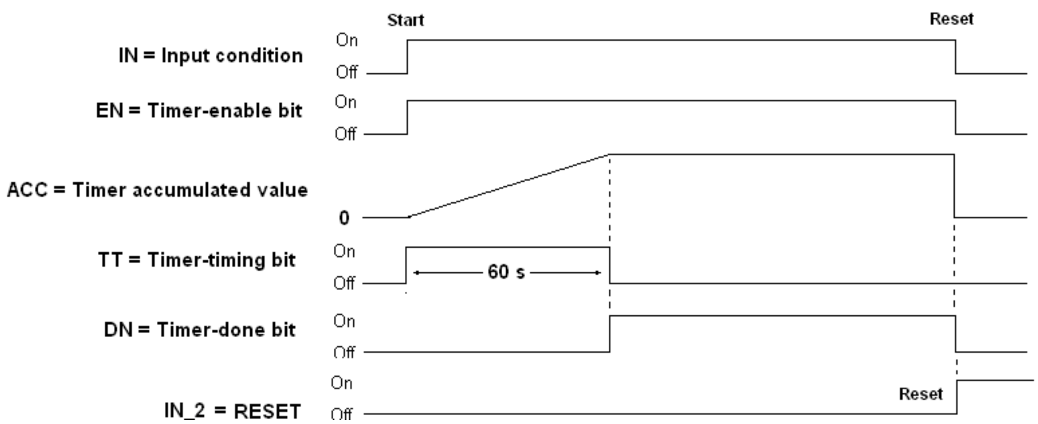


Figure 0-8 TON Timing diagram

	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
First Word	EN	TT	DN													
Second Word					PRE	VAL	UES	16	BITS							
Third Word					ACC	VAL	UES	16	BITS							

Figure 0-9 TON memory register

Addressable bits:
EN = Bit 15 enable
TT = Bit 14 timer timing
DN = Bit 13 done

Addressable words:
PRE = Preset value
ACC = Accumulated value

TON Timer can be reset with IN_2 or with the **NEGATIVE** edge of IN_1.
ACC stops accumulating time after ACC = PRE
ACC = 0 after RESET
DN = 1 when ACC = PRE
EN = 1 is a required condition for the timer to count time. If a push-button is used as IN_1, the rung must be sealed.
IN_2: transition from low to high resets the timer

OFF – DELAY TIMER (TOF)

Lorenzo – March 2, 2010

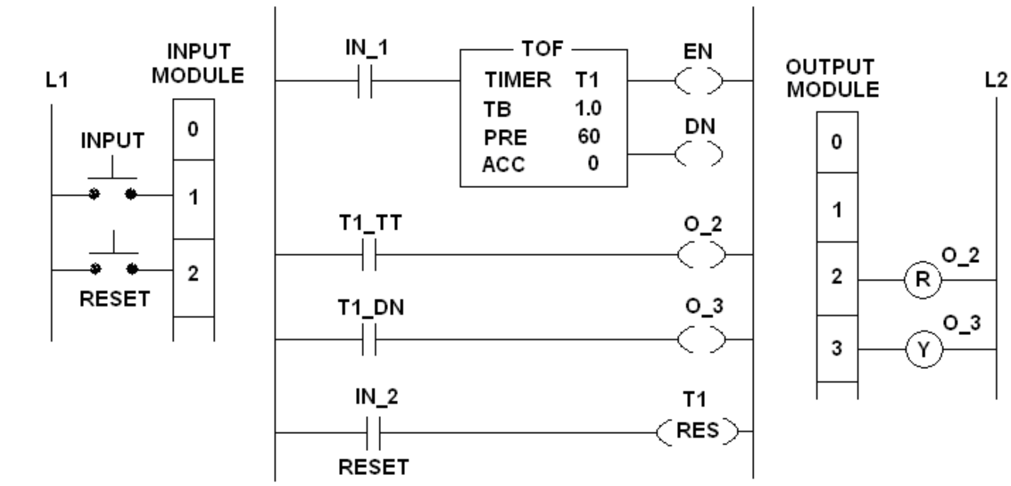


Figure 0-10 TOF Timer ladder logic

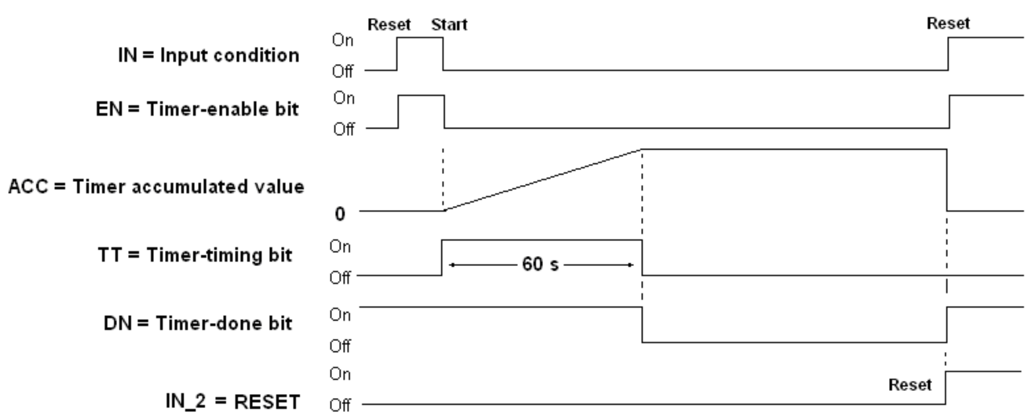


Figure 0-11 TOF Timing diagram

	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
First Word	EN	TT	DN													
Second Word					PRE	VAL	UES	16	BITS							
Third Word					ACC	VAL	UES	16	BITS							

Figure 0-12 TOF Timing register

Addressable bits:
EN = Bit 15 enable
TT = Bit 14 timer timing
DN = Bit 13 done

Addressable words:
PRE = Preset value
ACC = Accumulated value

TOF timer can be reset with IN_2 or with the **POSITIVE** edge of IN_1
ACC stops accumulating time after
ACC = PRE. ACC = 0 after RESET
DN = 0 when ACC = PRE, otherwise is equal to 1.
EN = 0 is a must condition for the timer to time, therefore we DO NOT need to seal the rung..

RETENTIVE TIMER (RTO) - Theory

- RTO accumulates time whenever the device receives power, and maintains the current time should power be removed from the device.
- Once the device accumulates time equal to its pre-set value, the contacts of the device change state.
- Loss of power to the device after reaching its pre-set value does not affect the state of the contacts.
- The retentive timer must be intentionally reset with a separate signal for the accumulated time to be reset and for the contacts of the device to return to their shelf state.

EXAMPLE

- Use an RTO to detect whenever a piping system has sustained a cumulative overpressure condition of 60 seconds. At that point, a horn must sound automatically to call attention to the malfunction.
- When they are alerted, maintenance personnel can silence the alarm by activating the RESET switch IN_2. This is the reset condition.

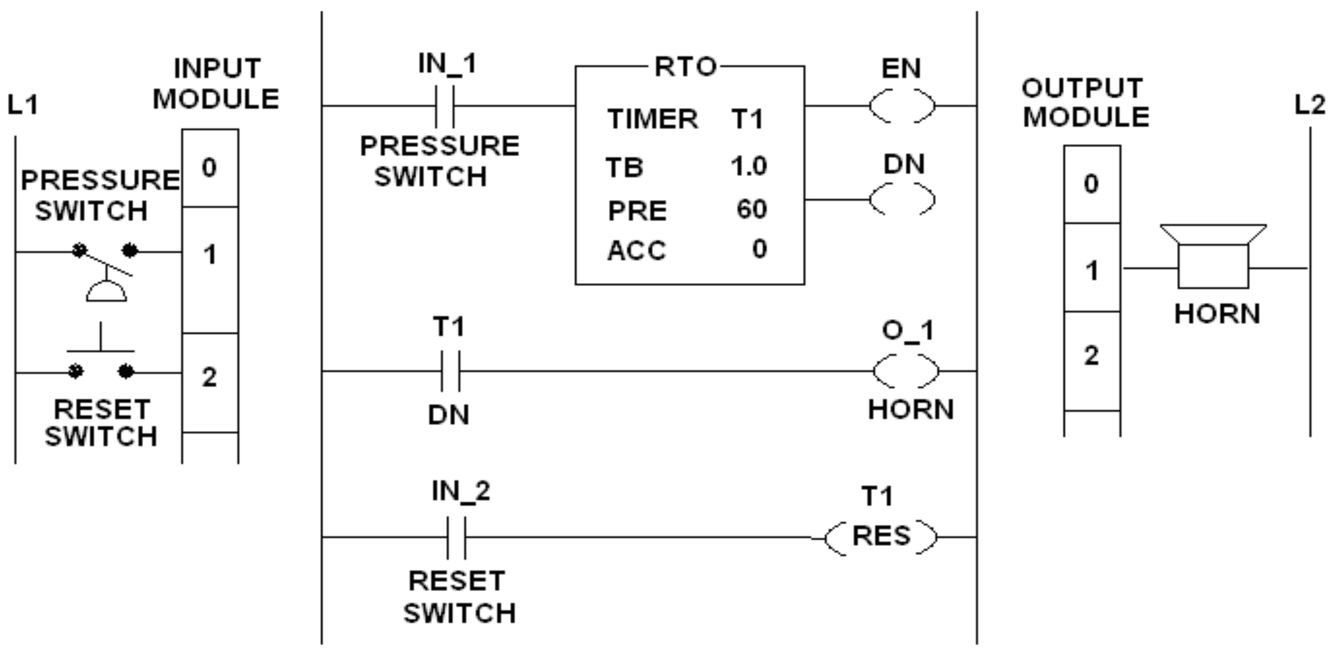


Figure 0-13 Retentive timer ladder logic

RETENTIVE TIMER (RTO) - QUIZ

- RTO accumulates time whenever the device receives power, and maintains the current time should power be removed from the device.
- Once the device accumulates time equal to its preset value, the contacts of the device change state.
- Loss of power to the device after reaching its preset value does not affect the state of the contacts.
- The retentive timer must be intentionally reset with a separate signal for the accumulated time to be reset and for the contacts of the device to return to their shelf state.

QUIZ:

- Use an RTO to detect whenever a piping system has sustained a cumulative overpressure condition of 60 seconds. At that point, a horn must sound automatically to call attention to the malfunction.
- When they are alerted, maintenance personnel can silence the alarm by switching the key switch S1 to the reset position.
- After the problem has been corrected, the alarm system can be reactivated by switching the key switch to the ON position.

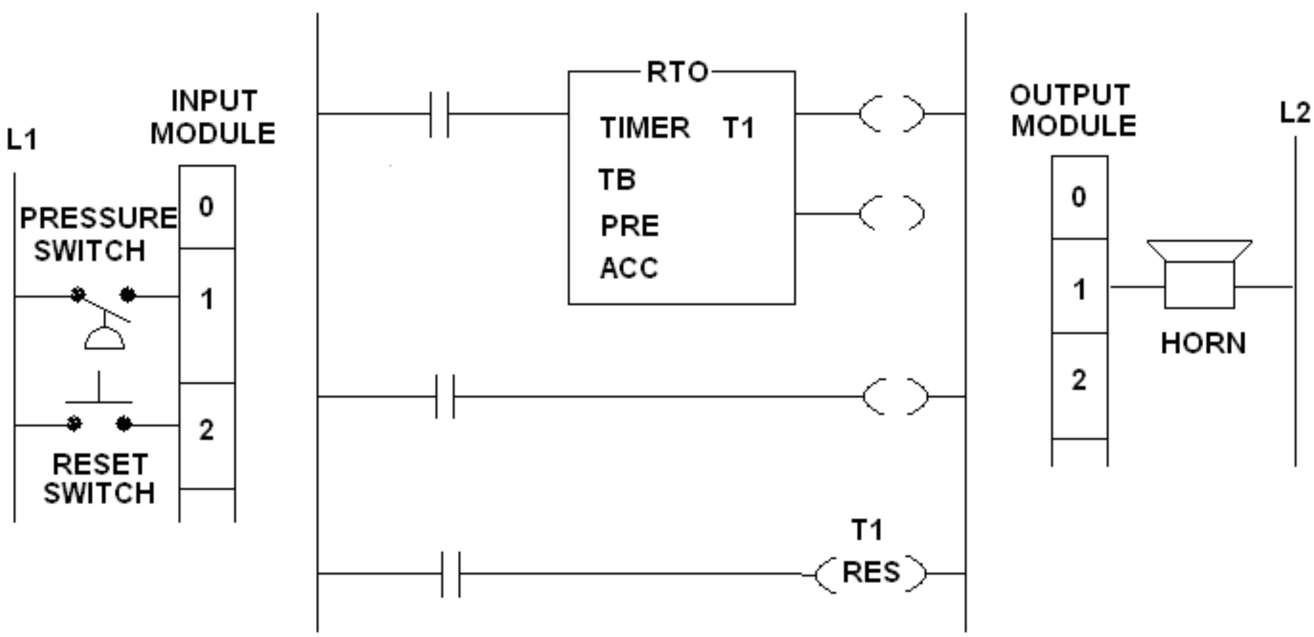


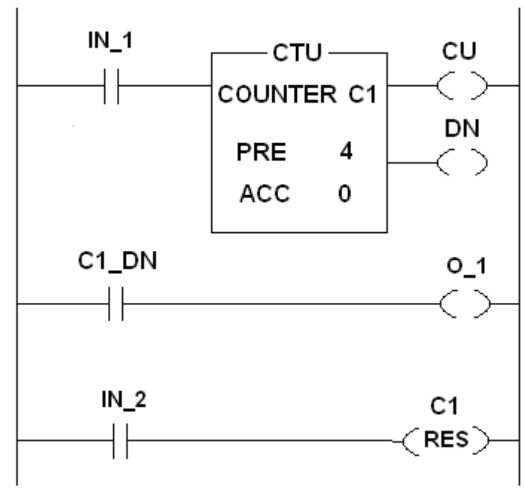
Figure 0-14 RTO Timing circuit

COUNTERS

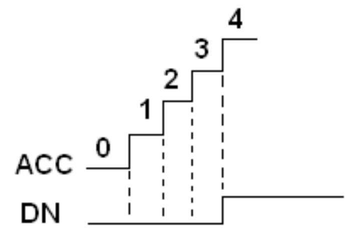
FOR BOTH COUNTERS: **DN = 1 when ACC ≥PRE**

ACC CONTINUES INCREMENTING AFTER IT REACHES PRE (Different that timers) THE COUNTERS MUST BE INTENTIONALLY RESET WITH A SEPARATE SIGNAL

UP COUNTER



DN = 1 when ACC ≥PRE



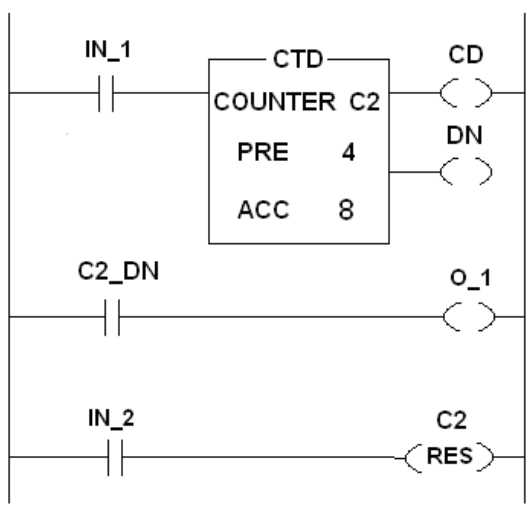
PULSES NEEDED TO CHANGE DN: **|ACC–PRE|+0**

Examples for programmed values of Pre and Acc:

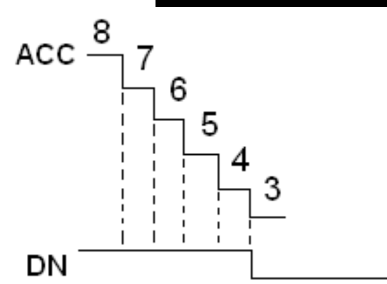
If Acc = 0, Pre = 4, then pulses needed are $|0 - 4| + 0 = 4$

If Acc = 5, Pre = 9, then pulses needed are $|5 - 9| + 0 = 4$

DOWN COUNTER



DN = 1 when ACC ≥ PRE



PULSES NEEDED TO CHANGE DN: **|ACC–PRE|+1**

Examples for programmed values of Acc and Pre:

If Acc = 8, Pre = + 4, then pulses needed: $|8 - 4| + 1 = 5$

If Acc = 0, Pre = - 4, then pulses needed: $|0 - (-4)| + 1 = 5$

Each counter requires 3 words of memory, as shown below.

	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
First Word	CU	CD	DN	OV	UN											
Second Word					PRE	VAL	UES	16	BITS							
Third Word					ACC	VAL	UES	16	BITS							

This page is intentionally left blank.

PARKING GARAGE PROBLEM

Use Counters to design the following assignment:

Part I (70%)

1.- 20 Cars max. (Start with an empty garage, ACC = 0)
2.- One entrance, one car at a time (Car-In @ IN-1)
3.- One exit, on car at a time (Car-Out @ IN-2)
4.- A Red Light indicates that the garage is full and no car is admitted…
5.- A Green Light must be ON when the garage is open to more cars.

Part II (30%)

1.- 20 Cars max. (Start with an empty garage, ACC = 0)
2.- One entrance, one car at a time (Car-In @ IN-1)
3.- One exit, on car at a time (Car-Out @ IN-2)
4.- A Red Light indicates that the garage is full and no car is admitted…
5.- … no car is admitted until the garage occupancy drops to 15.
(Therefore the Red Light must be ON when count reaches 20, 19, 18, 17,16,15)
6.- A Green Light must be ON when the garage is open to more cars.

RULES:

1. The objective of this project is to verify that the student is able to write a simple program, download it, and run it by himself (Without any help).
2. This is an individual project. Each student must work alone at all steps of the project. Image that you are on a moon base and you are the only one that knows about PLCs and your project is essential for the base survival.
3. The grade of this project will count as one lab experiment as follows:
   a) Downloading and running Part I of the program to the PLC will count as 50 % of grade.
   b) Downloading and running Part II of the program to the PLC will count as 50 % of grade.
   c) You can run the programs many times until it runs successfully. If you cannot correct errors in your program by the due date, the TA will assigned you a grade according with performance.

This page is intentionally left blank.

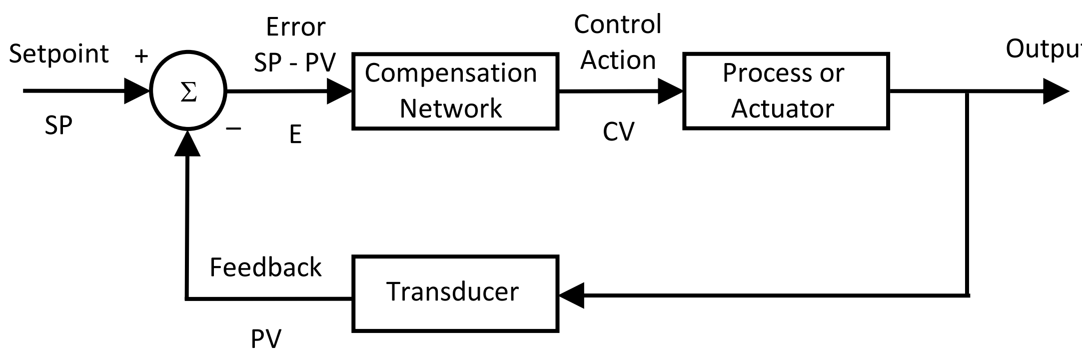
PROPORTIONAL + INTEGRAL + DERIVATIVE CONTROL (PID CONTROL)

CLOSED-LOOP CONTROL SYSTEMS **Roger Lorenzo – Jan 2011**

A closed-loop control system is an arrangement of physical components that achieve self-regulation. It requires an input known as the **setpoint** which provides the system with a goal for self regulation.

To obtain self-regulation the system requires knowledge of performance which is obtained at the **summing junction** (Σ) as an **error**, which should approach **zero**.

To obtain this error signal, **feedback** of the system output is required, which is performed by a suitable **transducer**. The feedback signal is sent to the summing junction and compared to the setpoint. Any error between the setpoint and feedback of the output is then passed to the **compensation network** for amplification and further processing and becomes the control action. This signal becomes the driving force for the process or actuator under control. The process or actuator then responds by modifying the output and the control system is complete.



PID is an acronym for Proportional + Integral + Derivative.

PID control is based on a math instruction performed by the compensation network. This math instruction acts on the error signal and provides the values of the control variable **CV** for the process to respond accordingly.

The PID instruction is:

$$CV = Kp * E + Ki * IE + Kd * DE$$

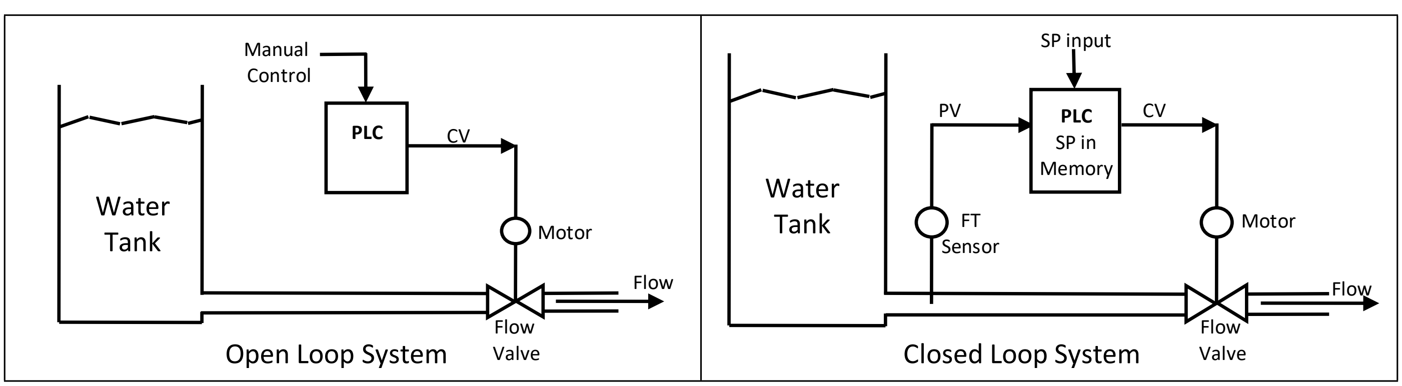
The greater the error between the set point and process variable input, the greater the output signal will change and vice versa. The PID instruction will automatically control the output signal, up or down, until the error between the setpoint and process variable is nearly gone.

The Allen-Bradley SLC 500 PID instruction uses the standard PID equation in the following form:

$$CV = Kc\,[(E + 1/Ti)\int (E)dt + Td \bullet D(PV)/dt] + bias$$

where; Kc = Proportional Gain Constant, 1/Ti = Reset Gain or Integral, Td = Rate Gain or Derivative, E = Error

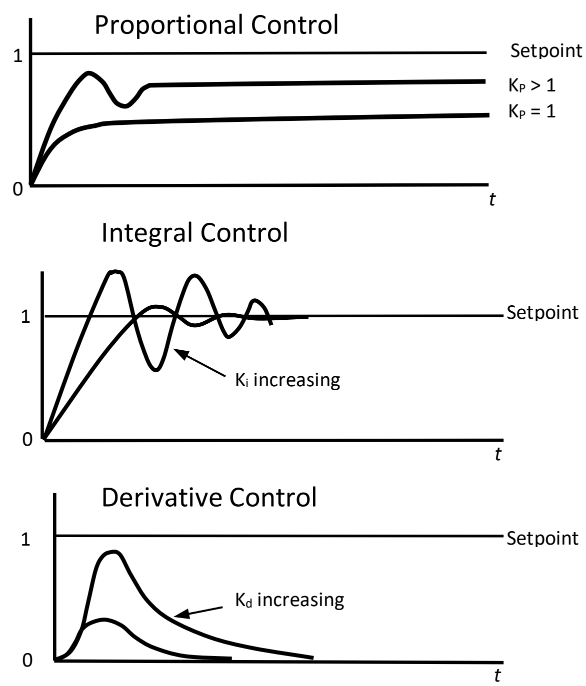
Examples of control systems:

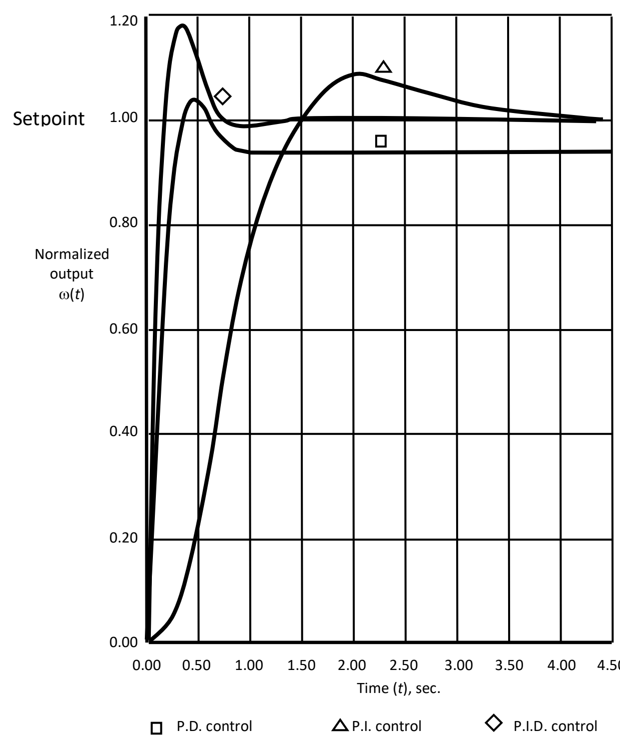


Electrical and pneumatic signal are used to convey the information between the different parts of the system. They are typically generated by field-mounted devices such as transducers and transmitters. These process signals are sent to PLCs, loop controllers, displays, and other devices as one of the following standard analog type signals:

a. Low DC electrical current (4 – 20 ma, 0 – 20 mA)
b. Low DC electrical voltage (0 – 10 V DC, 1 – 5 V DC, +/– 10 V DC)
c. Low air pressure signal (3 – 15 psig)

MORE ON PROPORTIONAL INTEGRAL DERIVATIVE CONTROL





P.D., P.I., and P.I.D. system response

- In proportional control alone we find a constant error present at all times, which diminishes as gain increases, but <u>never</u> reaches zero.
- In Integral control we find a tendency towards instability as gain increases, the system is slow, but it promotes zero error control.
- In derivative control we find that the system responds fast to change, but the error is 100%.
- It is evident that neither integral nor derivative control can be use alone.
- Proportional plus derivative control provides a quick response with good overshoot control, but a constant error is present at all times.
- Proportional plus integral control provides zero error performance, but increases the settling time and the system is sluggish.
- Proportional plus integral plus derivative control provides an exceptional control system with zero error, acceptable overshoot and excellent control over settling time.

SYSTEM RESPONSE AND STABILITY

Control systems provide smooth responses to a change in conditions. The goal is to provide a fast, accurate and stable output. However, this is not a simple task because the mechanical systems are composed of some mechanical characteristics, such as inertia and friction, which affect the output response. Ringing and Hunting are two kind of responses due to this characteristics. The following terms are used to explain how a drive is performing:

(1) Stable response, (2) Ringing, and (3) Hunting.

(1) STABLE RESPONSE: Figure A shows a stable response to a rapid change in the command signal. This can be speed, temperature, position, or any other system variable sensed by the drive. Notice the presence of the overshoot before reaching the final set point.

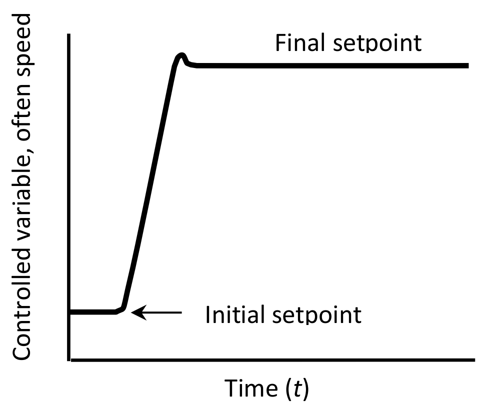


Figure A

(2) RINGING: Figure B shows a slight instability, called "Ringing". Here, the motor hunts during a short period for the commanded signal and finally reaches the desired speed.

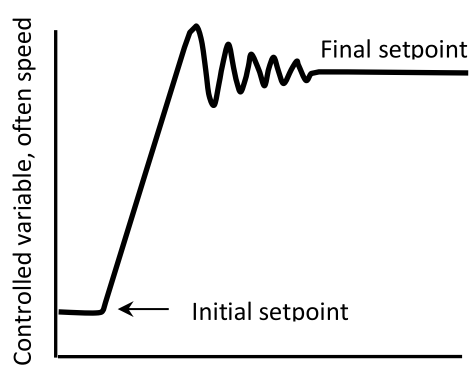


Figure B

(3) HUNTING: Figure C shows an unstable operation known as "Hunting". This is the most undesirable of all situations.

Here, the motor continually hunts around de desired value, but never settles there.

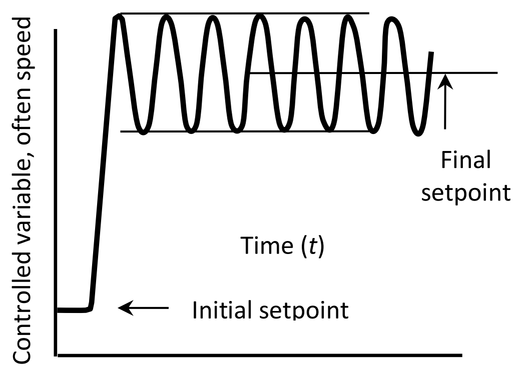


Figure C

RESPONSE TERMINOLOGY

TIME CONSTANT: Time constant is the length of time, in seconds, required to reach 63.2 % of the final value. System response with a single time constant (First order systems) achieve the final value in five time constants as shown in Figure D.

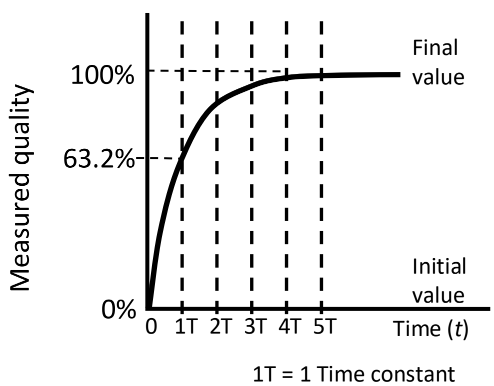


Figure D

HIGHER ORDER SYSTEMS

Energized drive systems are higher order systems containing inertia, a spring-like characteristic, and damping. Their response can be explained as shown in Figure E.

INERTIA: Is the sum of the inertias of the drive motor's rotor and the driven loads.

SPRING: Electrical characteristics in the motor and mechanical in twisting of the shaft.

DAMPING: Combined action of the control system, the motor, and other mechanical components.

RESPONSE TERMS

Figure E shows the response of higher order systems to a rapid change in operation conditions (Speed, temperature, or position). Five specific terms are used to define the desired of any closed-loop control system:

(1) Delay Time, (2) Rise Time, (3) Maximum overshoot, (4) Peak Time, and (5) Settling time.

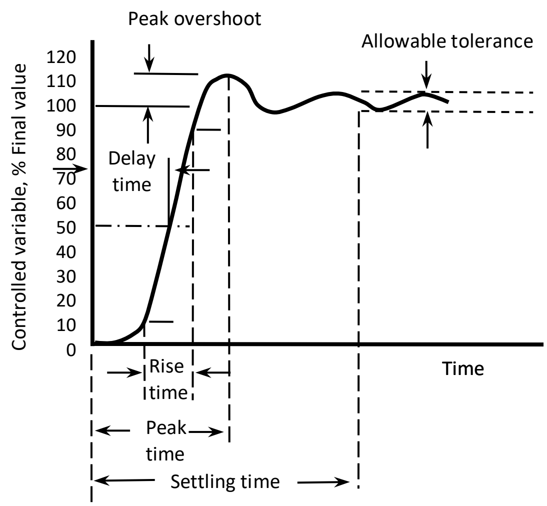


Figure E

(1) Delay Time: The time required to reach 50 % of the final value.

(2) Rise Time: The time required to go from 10 % to 90 % of the final value.

(3) Maximum Overshoot: the maximum peak value above the final desired value. Maximum overshoot is expressed as a percent of the final value.

(4) Peak Time: The time required to reach the first peak overshoot.

(5) Settling Time: the time required to reach and stay within the allowable tolerance band around the final value, such as 2%, 1%, etc.

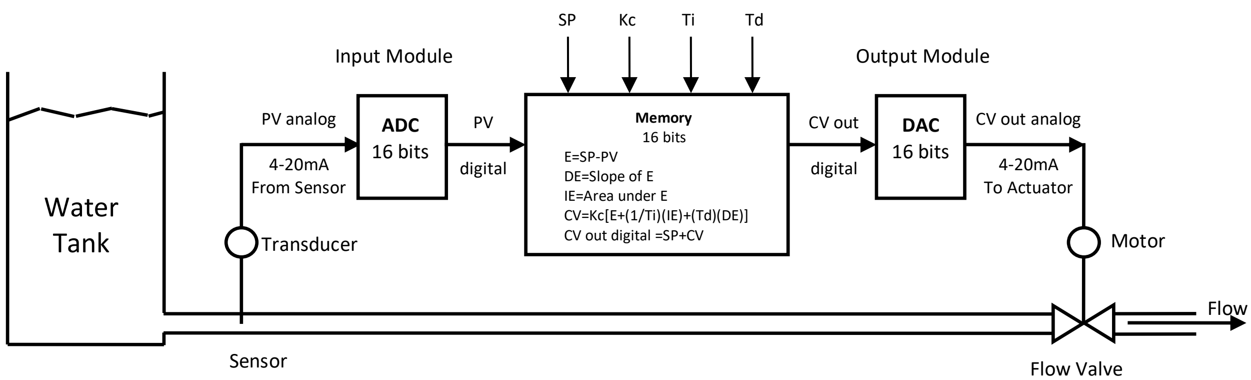
PID – CONTROL EXAMPLE

Given:

1. The process under control is the outlet rate of a liquid storage tank.
2. The flow rate is controlled by a flow valve located in the outlet pipe.
3. The PLC will output a signal to control the valve using a 4-20 ma connection.
4. To generate the output signal the PLC will have two inputs:
    a. SP (Set Point). This value can be a constant in the PLC memory or an input signal.
    b. PV (Process variable) or feedback with actual flow.

Required:

1. Write the code to read the feedback value (PV)
2. Write the equation for the PLC output (CV)



Sequence for PLC program:

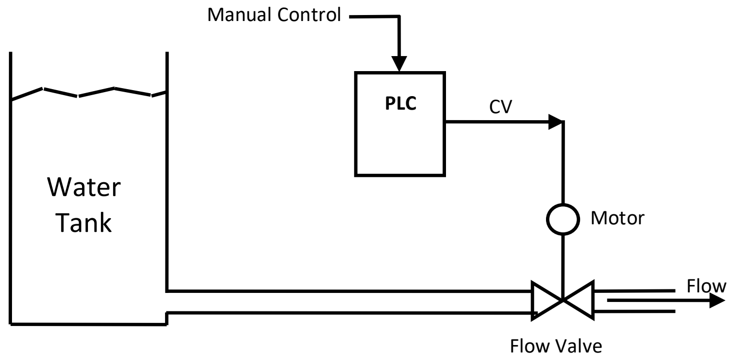
1. Input PV from sensor.
2. Program evaluation and upgrade of CV   CV = Kc [E + (1/Ti) (IE) + (Td) (DE)]
    - Kc, Ti, and Td are constants calculated before running the process.
    - E = SP – PV
    - IE = Area under ΔE curve (Integral)
    - DE = slope of ΔE curve (Derivative)

3. Calculate and output [CV_{out} digital = SP + CV] to output module and actuator (Motor)

1. CALCULATING Kc, Ti, and Td. (P.I.D. TUNING)

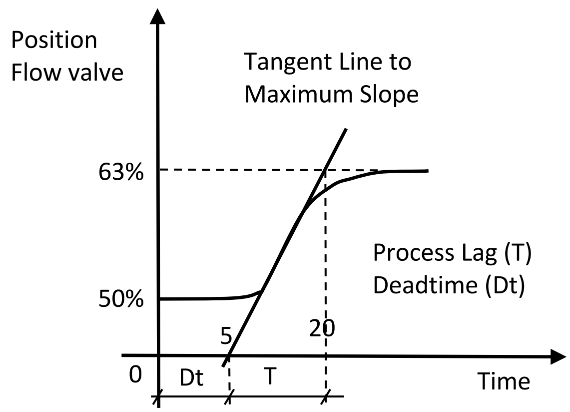
(a) Ziegler and Nichols “REACTION CURVE METHOD” – Open loop technique

We use a technique developed by Ziegler and Nichols called “Reaction Curve Method” of “Open-Loop Technique”. The technique consists of setting a manually- operated open-loop operation of the process as follows:



With this setup, the following sequence is done:

1. Manually control the process to the flow valve open 50%.
2. Let the system to get stable.
3. Manually change the input by 10% from 50 % to 60 %.
4. Record using a strip-chart the process response.
5. Use the graph to calculate Kc, Ti, and Td as follows:



K = {Actual change %} / {Desired change %} Kc = {(1.2) (T)} / {(Dt) (K)}
Ti = 2 Dt
Td = Dt / 2
T_{Loop} = (Dt) x (0.8) , ……. T_{Loop} = Loop update time.

Using the data from the graph above:

K = {13%} / {10%} = 1.3
Kc = {(1.2) (15)} / {(5) (1.3)} = 2.77
Ti = 2 (5) = 10 seconds
Td = (5) / 2 = 2.5 seconds
T_{Loop} = (5) x (0.8) = 4 seconds

CV_1 can be calculated with these numbers and the values of SP and PV since E = SP – CV

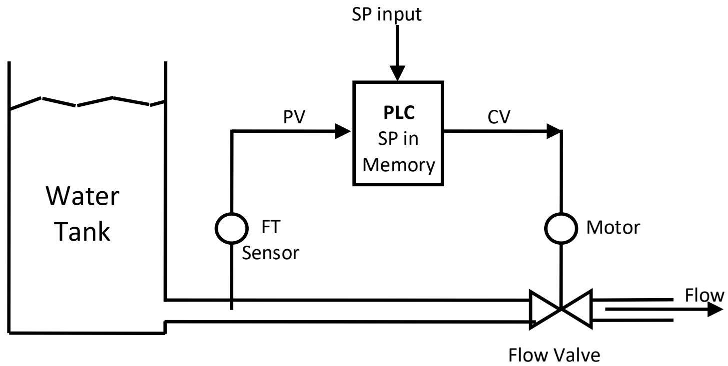
$$CV_1 = Kc \, [\, E + (1/Ti)\,(IE) + (Td)\,(DE)\,]$$

IMPORTANT

Notice that Kc, Ti, and Td are constant. Once they are determined they do not change during the process. On the other hand, CV changes after every iteration of the process.

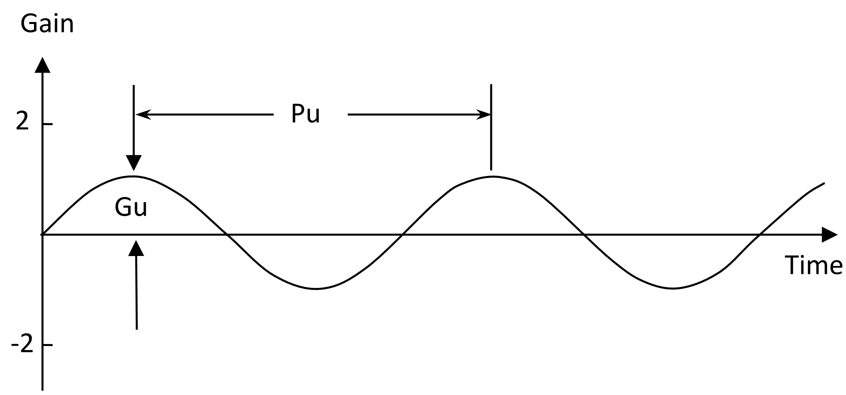
The iteration is not the same as a scan cycle. Scan cycles take milliseconds; iterations are periodic reading of the liquid flow to update PV and CV controlled by a timing that depends on the process under control. Iteration time is T_{Loop}.

(b) Ziegler and Nichols "ULTIMATE GAIN METHOD" – Closed loop technique



CV = Kc [E + (1 / Ti) (IE) + Td (DE)]

Kc makes a faster response but leaves a permanent error
Ti makes the error equal to zero but the system may become unstable
Td, together with Kc and Ti, provides for a faster, stable, and error free system



Method to find or calculate Kc, Ti, and Td.

1. Connect your system in a closed loop configuration.
2. Turn the integral or reset time (Ti) to its longest or highest setting (1000 is OK).
3. Turn the derivative or rate time (Td) to zero.
4. Increase the gain Kc gradually until the gain loop maintains a stable sine wave form.
5. Measure the Ultimate Gain (Gu) and Ultimate Time Period (Pu) of the stable sine wave form.
6. Calculate Kc, Ti, and Td as follows:

   Kc = Gu x 0.6 Ti = Pu / 2 Td = Pu / 8

7. Calculate the loop update time (Time between process data sampling) as follows:

Example: If Gu = 4.62, and Pu = 20 seconds,

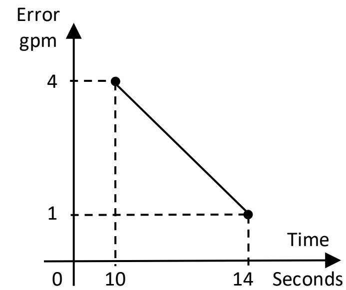
Kc = 4.62 x 0.6 = 2.772 = 2.77
Ti = 20 / 2 = 10 seconds.
Td = 20 / 8 = 2.5 seconds.
T_{loop} = 20 / 5 = 4 seconds
T_{loop} = Pu / 5

2. CALCULATING E, IE, and DE

To find E, IE, and DE we need to use data acquired by the PLC for two values of time (Present and before), for example:

Time	PV (Actual flow)	E = SP – PV Assume SP=100 gpm
10 sec	96 gpm	4 gpm
14 sec	99 gpm	1 gpm

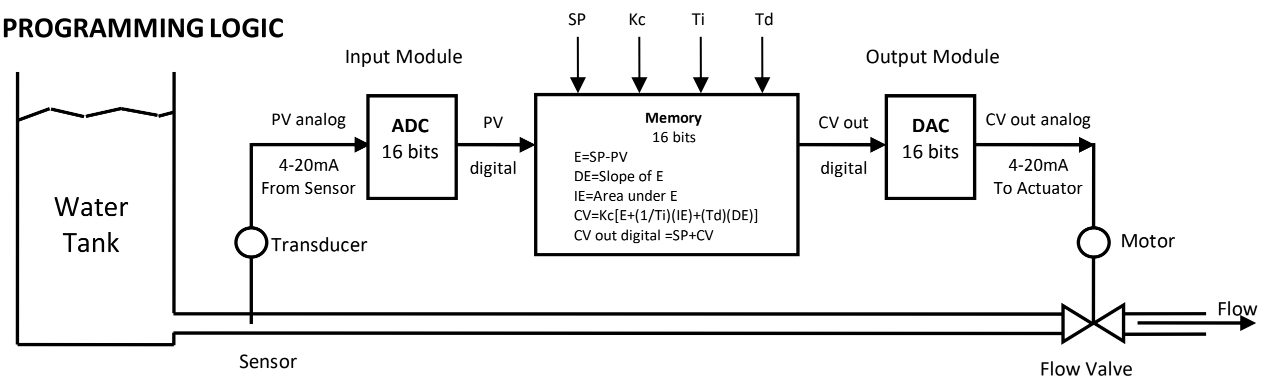
Next, we graph the E function:



From here we calculate E, IE, and DE

- E = Present error = 1 gpm (@ t=14 seconds)
- IE = (Area under E for 10 ≤ t ≤ 14) = [(4 – 1) (14 – 10) / 2 + (1 – 0) (14 – 10)] = 10 gpm*sec
- DE = (Slope of E-line) = (4 – 1) / (10 – 14) = – 0.75 gpm/sec

3. PROGRAMMING LOGIC



Data: Kc = 2.77 E = 1gpm Ti= 10sec IE = 10gpm*sec Td= 2.5 sec DE= - 0.75gpm/sec

Therefore :

CV = Kc [E + (1/Ti) (IE) + (Td) (DE)]

CV = 2.77 [1gpm + (1/10sec) (10gpm*sec) + (2.5sec) (- 0.75gpm/sec)] = 0.346 gpm

CV out = PV + CV = 99 + 0.346 = 99.346 gpm

QUESTIONS: : (Input and output ranges = 4 to 20 mA and 0 to 125 gpm)

1. If the signal from the sensor is 90 gpm, how many milliamperes outputs the transducer? ____________ Answer: 15.52 mA.
2. If the data from the output module is 18 ma, then the motor will set the flow to be_________. Answer: 109.4 gpm.

ORAL PRESENTATION– Understanding Communication Networks

RULES:

1. Grade: Your presentation will be graded as follows:
   a. Written report from the team (required for a grade) 10 points
   b. Written peer evaluation (weight 30%) 27 points
   c. Subjective instructor evaluation (weight 70%) 63 points
2. Grade will count as a lab exercise
3. Four to Eight minute Power-Point presentation
4. Use Textbook as reference and look in the Internet and/or library to expand the presentation.
5. Use pictures and graphs to enhance your presentation. (From Internet and/or library)
6. Some questions of final test will come from these topics.
7. No humor (of any kind) is allowed during the presentations.
8. Attendance is not optional. If you do not attend and participate, you will get a zero for that lab assignment. Your peer evaluation will be the record of your attendance and participation. A form will be provided for your evaluation of the presentation. Remember that you will also be evaluated so be fair and professional to the other members of the class.

ORAL PRESENTATION QUESTIONS. 6th edition pages (5th edition pages in parenthesis)

Presentation 1: History and Network Principles – Pages 378 to 380 (358 to 360).

1. A communication network exists when two or more devices are connected together by some type of media for the sole purpose of exchanging information.
   a. True b. False
2. Bandwidth is the speed at which information can be transferred. a. True b. False

Presentation 2: Network Categories – Pages 380 to 382 (360 to 362).

1. Which of the following is not a type of Network Category?
   a. WAN b. CAN c. MAN d. LAN
2. The LAN system is the largest type of Network. a. True b. False

Presentation 3: Network Configurations – Pages 382 to 384 (362 to 364).

1. The three network topologies are:
   a. Bus, Taxi and Train b. Ring, Bus and Sun c. Sun, Ring and Star d. Ring, Bus and Star
2. The two topologies that are not commonly used in industrial plants area;
   a. Bus and Ring b. Ring and Star c. Bus and Star

Presentation 4: Network Media and Twisted-Pair Cable – Pages 384 to 386 (364 to 366).

1. Shielded twisted-pair cable is most often used in offices and buildings?
   a. True b. False
2. Which of the following is not a type of twisted pair cable?
   a. UTP
   b. STP
   c. BTP-45
   d. CAT-5

Presentation 5: Coaxial Cable – Pages 387 to 388 (367 to 368).

1. Coaxial cable is used widely in industrial control networks because:
   a. It is very easy to work with.
   b. It is cheaper than twisted pair cable.
   c. Its immunity to electrical noise generated by equipment.

2. A Coaxial Cables have characteristic impedances, 50 Ω, 75 Ω, etc.
   a. True b. False

Presentation 6: Fiber Optic Cable – Pages 389 to 391 (369 to 371).

1. The two typical devices to convert electrical signals into light are:
   a. LED and PHASORS b. LED and LASERS
2. Fiber optic is the best type of network media because:
   a. Its high data rates over long distances. b. Its immunity to electrical noise.
   c. It carries no current, so that is not a fire hazard.
   d. All of the above.

Presentation 7: Reducing Electromagnetic Interference – Pages 391 to 392 (371 to 372).

1. EMI is:
   a. An electrical motor used in industry.
   b. An industrial magnet used in electrical equipment.
   c. Electrical noise.
   d. Electrical resistance.
2. Running a cable in steel conduit does not provide excellent EMI protection:
   a. True b. False

Presentation 8: Network Addressing – Pages 392 to 393 (372 to 373).

1. Any device connected to a network must have some means of identification so that it only receives the messages that are intended for it.
   a. True b. False
2. Which of the following is an example of a typical residential IP network address?
   a. C:\My Computer
   b. 255.255.0.0
   c. “Port Restricted Cone” NAT
   d. None of the above

Presentation 9: Network Access Methods – Pages 394 to 395 (374 to 375)

1. What is the most common access method used in industrial control networks?
   a. Master/Slave Access Method
   b. Token Passing
   c. Polling Access Method
   d. CSMA/CD
2. Which Access Method gives nodes equal right to transmit, without waiting for permission?
   a. CSMA/CD
   b. Polling Access Methods
   c. Master/Server Access Method
   d. Token Passing

Presentation 10: Network Protocols – Pages 395 to 396 (375 to 376)

1. Network communication protocols are sets of formal rules on how to transmit and share data across a network. OSI is one of these protocols and stands for:
   a. Operation Standard Interconnect b. Open Standard Interconnect
   c. Open Solution Interface
   d. Open Systems Interconnect
2. What layers are required for the OSI model to operate?
   a. Physical, Network, and Transportation layers
   b. Transportation and Application layers
   c. Physical, Data, and Application layers d. All layers

Presentation 11: Network Messages – Pages 396 to 399 (376 to 379)

1. A message is a common term used to describe information transmitted via a network
   a. True b. False
2. If a message is too large for one data packet, the message is broken up into several data packets and each is given a sequence number.
   a. True b. False

Presentation 12: Network Com. Inst, I/O Device Net, Control Networks, Information Networks – Pages 399 to 403 (379 to 383)

1. Proprietary networks, as opposed to open system network protocols, allow devices from many different manufacturers to be used. a. True b. False
2. Some of the most common device & process bus protocols used today are:
   a. DeviceNet, Foundation Fieldbus, and Profibus-DP
   b. Foundation Fieldbus, Profibus-DP, and PID-Control
   c. Profibus-DP, PID-Control, and Open-Loop Control
   d. PID-Control and Open-Loop Control
3. Information networks work at the middle level of the control system. a. True b. False

Presentation 13: Ind Protocols, Device Net, Foundation Fieldbus - Pages 403 to 405 (383-385)

1. Control networks use which network access protocol?
   a. Polling (Master-Slave)
   b. Token (Peer-to-Peer)
   c. CSMA/CD
   d. All of the Above

1. Foundation Fieldbus, established in 1994, is a single international open bus standard for hazardous environments that enables devices of different manufacturers
2. to be integrated into one system.
   a. True b. False
3. Which bus has a faster communication speed?
   a. H5
   b. HRE
   c. HSE
   d. H1

Presentation 14: Profibus+Modbus+Data Hwy+ControlNet – Pages 406 to 407 (386 to 387)

1. Profibus was developed by ________, Modbus by _______, and ControlNet by ________.
   a. Allen Bradley, Modicon, Siemens
   b. Modicon, Allen Bradley, Siemens
   c. Siemens, Modicon, Allen-Bradley

3. Modbus has a slower data transfer rate than DH+.
   a. True b. False

Presentation 15: Ethernet – Pages 407 to 411 (387 to 391)

1. The main difference between Data Terminal Equipment (DTE) and Data Communication Equipment (DCE) is that DTE produces and consumes data on network, while DCE receives and forwards data across the network.
   a. True b. False
2. Who invented the Ehernet Network Connection?
   a. Apple Corporation
   b. Intel Corporation
   c. Microsoft Corporation
   d. Xerox Corporation
3. Devices connected to an Ethernet network from different manufacturers are not guaranteed to communicate with each other, even though they are on the same network called “Ethernet”.
   a. True b. False
4. A Network Interface Card (NIC) is a built-in Ethernet network interface port identified by three parts: its transmission rate, transmission method and cable type.
   a. True b. False

END OF LAB MANUAL