Programmable Logic Controllers

Sixth Edition

Frank D. Petruzella

McGraw Hill

PROGRAMMABLE LOGIC CONTROLLERS, SIXTH EDITION

This book is printed on acid-free paper.

1 2 3 4 5 6 7 8 9 LWI 27 26 25 24 23 22

ISBN 978-1-264-16334-2 (bound edition)
MHID 1-264-16334-7 (bound edition)
ISBN 978-1-264-44684-1 (loose-leaf edition)
MHID 1-264-44684-5 (loose-leaf edition)

Portfolio Manager: *Beth Bettcher*
Product Developer: *Heather Ervolino*
Marketing Manager: *Lisa Granger*
Content Project Managers: *Maria McGreal, Samantha Donisi*
Buyer: *Laura Fuller*
Content Licensing Specialist: *Shawntel Schmitt*
Cover Image: *Xmentoys/Shutterstock*
Compositor: *Aptara®, Inc.*

Library of Congress Cataloging-in-Publication Data

Names: Petruzella, Frank D., author.
Title: Programmable logic controllers / Frank D. Petruzella.
Description: Sixth edition. | Dubuque : McGraw Hill Education, 2022. | Includes index.
Identifiers: LCCN 2021032477 (print) | LCCN 2021032478 (ebook) | ISBN 9781264163342 (hardcover) | ISBN 9781264446841 (spiral bound) | ISBN 9781264446759 (ebook) | ISBN 9781264446872 (ebook other)
Subjects: LCSH: Programmable controllers.
Classification: LCC TJ223.P76 P48 2022 (print) | LCC TJ223.P76 (ebook) | DDC 629.8/95—dc23
LC record available at https://lccn.loc.gov/2021032477
LC ebook record available at https://lccn.loc.gov/2021032478

mheducation.com/highered

About the Author

Frank D. Petruzella has extensive practical experience in the electrical control field, as well as many years of experience teaching and authoring textbooks. Before becoming a full-time educator, he was employed as an apprentice and electrician in areas of electrical installation and maintenance. He holds a Master of Science degree from Niagara University, a Bachelor of Science degree from the State University of New York College–Buffalo, as well as diplomas in Electrical Power and Electronics from the Erie County Technical Institute.

Contents

Preface

Programmable logic controllers (PLCs) continue to evolve as new technologies are added to their capabilities. As PLC technology has advanced, so have programming languages and communications capabilities. Today's PLCs offer faster scan times, space efficient high-density input/output systems, and special interfaces to allow non-traditional devices to be attached directly to the PLC.

The primary source of information for a particular PLC is always the accompanying user manuals provided by the manufacturer. This textbook is not intended to replace the vendor's reference material, but rather to complement, clarify, and expand on this information. The text covers the basics of programmable logic controllers in a manner that complements instruction with an RSLogix 500 or RSLogix 5000 platform. The underlying PLC principles and concepts covered in the text are common to most manufacturers. They serve to maximize the knowledge gained through on-the-job training and programs offered by different vendors.

The text is written in an easy-to-read style that is designed for students with no prior PLC experience. For example, when the operation of a program is called for, a bulleted list is used to summarize its execution. The bulleted list replaces a lengthy paragraph and is especially helpful when covering the different steps related to the execution of a program.

Each chapter begins with a brief introduction outlining chapter coverage and learning objectives. When applicable, the relay equivalent of the virtual programmed instruction is explained first, followed by the appropriate PLC instruction. Chapters conclude with a set of review questions and problems. The review questions are closely related to the chapter objectives and require students to recall and apply information covered in the chapter. The problems range from easy to difficult, thus challenging students at various levels of competence.

Chapter changes in this edition include:

Chapter 1

- Eliminated pictures of outdated/obsolete equipment.
- Updated programming methods to reflect that laptop computers are now most commonly used.
- Clarified I/O connections for the example in Figures 1-15 and 1-16.
- Illustrated the relative ease of modifying processes in a PLC system compared to conventional methods.
- Updated comparison of PLCs and personal computers.
- Removed obsolete references to memory sizes for specific AB controllers.

Chapter 2

- Removed I/O addressing as a chapter objective.
- Added term: Distributed I/O and definition.
- Added some clarification to tag-based addressing used with RSL5K.
- Added Image of Control Logix controller with analog I/O along with a better description of what analog I/O is.
- Removed references to SCP instruction, since that function is now usually performed when setting up analog channels in module properties.
- Removed references to outdated/obsolete I/O modules.
- Expanded on PLC CPU communication functions.
- Removed obsolete references to memory sizes for specific AB controllers.
- Updated data type sizes used for RSL5K.
- Removed thumbwheel switch problem and replaced with RSL5K tag creation problem.

Chapter 3

- Added more detail to binary math examples.

Chapter 4

- Minimal changes were made in this chapter.

Chapter 5

- Expanded on CPU scan cycle and logic evaluation.
- Removed reference to hand-held programming device.

Chapter 6

- Revised motor control circuit drawing.
- Removed instances of outdated/obsolete equipment.
- Updated sequential process programming example with one from RSL5K.

Chapter 7

- Clarified time base for timers.
- Removed references to coil-based instructions.

Chapter 8

- Removed references to coil-based instructions.

Chapter 9

- Removed obsolete instruction SUS.
- Added Label instruction.
- Removed SLC file references.
- Updated “force” definition.
- Removed section on STI.
- Updated fault routine definition.

Chapter 10

- Removed SLC data file map.
- Updated MOV example.
- Updated A to D converter graphic.

Chapter 11

- Minimal updates were made in this chapter.

Chapter 12

- Updated Sequencer problem.
- Updated question to reflect tag and array type variables.

Chapter 13

- Revised PLC power distribution drawing.
- Revised PLC grounding drawing.

Chapter 14

- Added section on variable frequency drives.

Chapter 15

- Corrected SINT definition.
- Revised timing diagram for OSF.
- Corrected tag name for Part 5 problem 1.

Acknowledgments

I would like to thank the following reviewers for their comments and suggestions:

Mahmoud A. Abdallah
Central State University

Michael Beavers
Lake Land College

Stephen Crosskno
Arkansas Northeastern College

Darrel Cunningham
Oklahoma State University Institute of Technology

Keith Dinwiddie
Ozarks Technical Community College

Charles J. Eckard
North Seattle College

Nathan Esbeck
East Central College

Montie Fleshman
New River Community College

Michael Garner
Lakeland Community College

Steve Gellings
Highland Community College

Jim Heffernan
Quinsigamond Community College

Landon Johnson
Bates Technical College

Gregory Jourdan
Wenatchee Valley College

Maged Mikhail
Purdue University

Jeffery Phelps
Southcentral Kentucky Community and Technical College

Sai Ramaswamy
Wenatchee Valley College

Richard Ramey
Wake Technical Community College

Darrell D. Smith
North Central Texas College

Scott Walker
Utah Valley University

Wade Wittmus
Lakeshore Technical College

Wesley Womack
Central Carolina Community College

Donna Zimmerman
Ivy Tech Community College

A special thanks to Brett Blasiman, Instructor, Electrical Engineering Technology, Gateway Technical College, for his outstanding work on performing a technical edit of the text and providing us with detailed feedback, suggestions and recommendations.

Frank D. Petruzella

P*rogrammable Logic Controllers* **makes it easy to learn PLCs from the ground up! Up-to-the-minute revisions include all the newest developments in programming, installing, and maintaining processes. Clearly developed chapters deliver the organizing objectives, explanatory content with helpful diagrams and illustrations, and closing review problems that evaluate retention of the chapter objectives.**

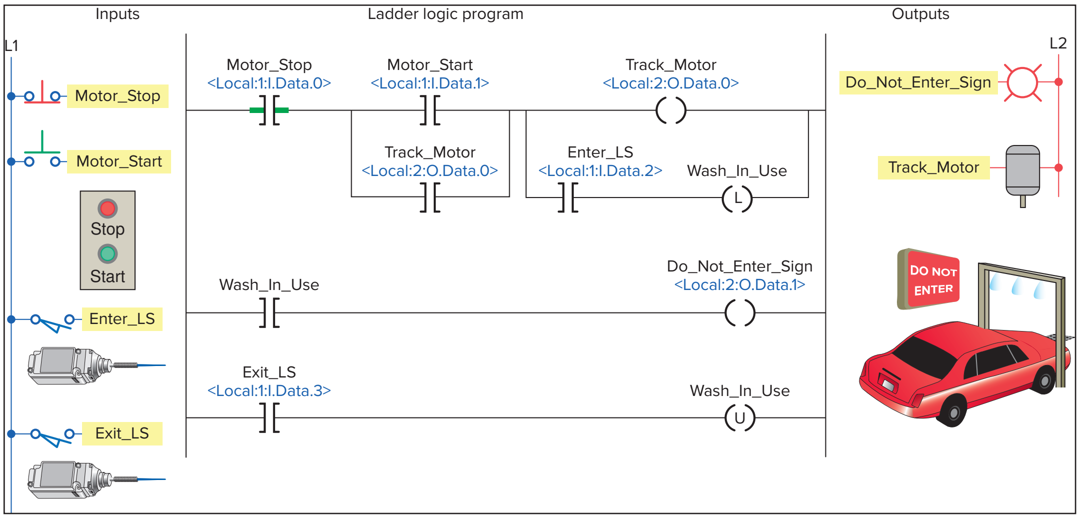
CHAPTER OBJECTIVES overview the chapter, letting students and instructors focus on the main points to better grasp concepts and retain information.

Chapter Objectives

After completing this chapter, you will be able to:

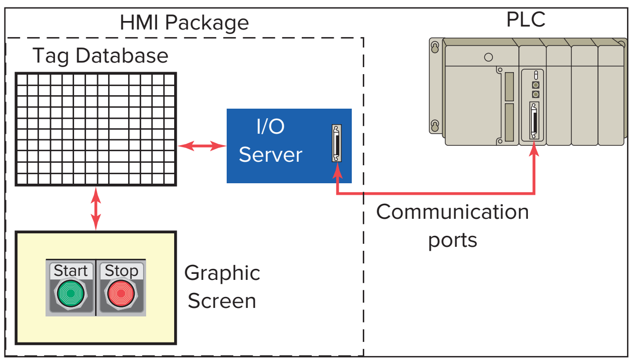
- Describe the operation of pneumatic on-delay and off-delay timers
- Describe PLC timer instruction and differentiate between a nonretentive and retentive timer
- Convert fundamental timer relay schematic diagrams to PLC programs
- Analyze and interpret typical PLC timer programs
- Program the control of outputs using the timer instruction control bits

Chapter content includes rich illustrative detail and extensive visual aids, allowing students to grasp concepts more quickly and understand practical applications

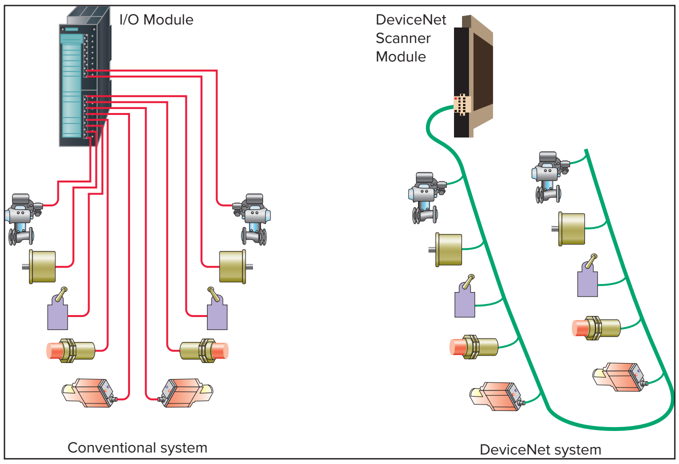


Here, drawings of real-world input and output devices have been included

In chapter 2, students not only read about but can also see how HMIs fit into an overall PLC system, giving them a practical introduction to the topics



Coverage of communications and control networks utilizes clear graphics to demonstrate how things work



A hardwired pilot light motor control circuit is shown in Figure 15-39. The operation of the circuit can be summarized as follows:

- The Stop/Start pushbutton station controls relay coil CR.
- When CR is de-energized, the green standby pilot light is ON, the red run pilot is OFF, and the motor is not operating.

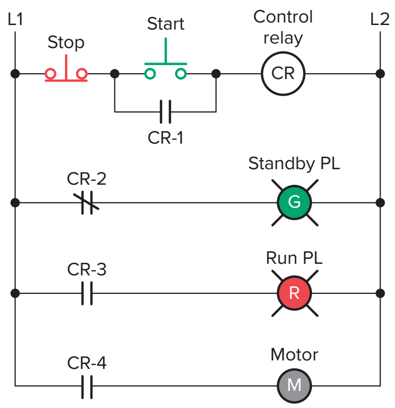


Figure 15-39 Hardwired pilot light motor control circuit

BULLETED LISTS break down processes to helpfully summarize execution of tasks

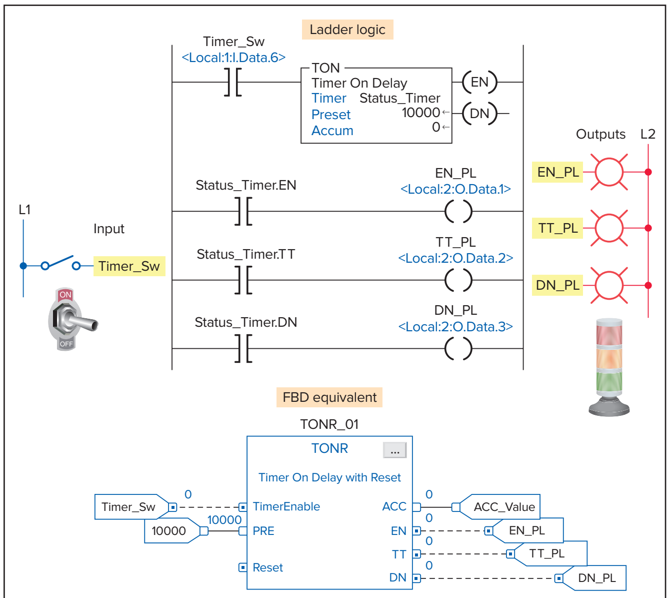
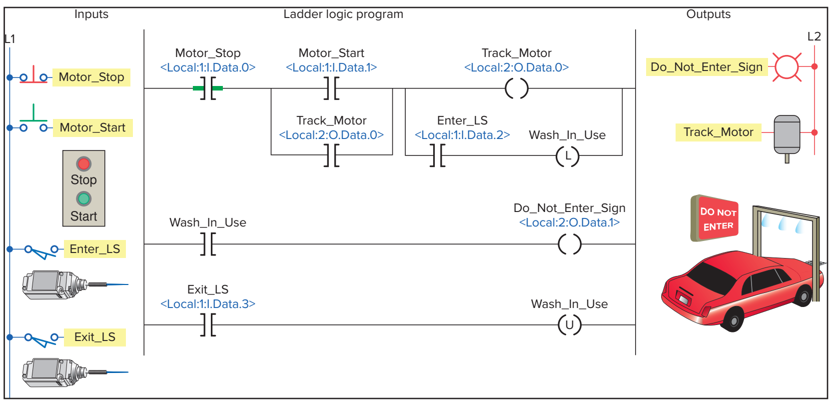


Figure 15-110 Comparison between ladder logic and the FBD equivalent for a 10 second TON and TONR timer.

Diagrams, such as this one illustrating an overview of the function block programming language, help students put the pieces together



More than **175** SLC-500 and ControlLogix program simulation **videos tied directly** to the programs studied in the text

END-OF-CHAPTER REVIEWS are structured to reinforce chapter objectives

CHAPTER 6 REVIEW QUESTIONS

1. Explain the basic operating principle of an electromagnetic control relay.
2. What is the operating difference between a normally open and a normally closed relay contact?
3. In what ways are control relay coils and contacts rated?
4. How do contactors differ from relays?
5. What is the main difference between a contactor and a magnetic motor starter?
6. **a.** Draw the schematic for an across-the-line AC magnetic motor starter.
   b. With reference to this schematic, explain the function of each of the following parts:
   i. Main contact M
   ii. Control contact M
   iii. Starter coil M
   iv. OL relay coils
   v. OL relay contact
7. The current requirement for the control circuit of a magnetic starter is normally much smaller than that required by the power circuit. Why?
8. Compare the method of operation of each of the following types of switches:
   a. Manually operated switch
   b. Mechanically operated switch
   c. Proximity switch
15. Compare the operation of the reflective-type and through-beam photoelectric sensors.
16. Give an explanation of how a scanner and a decoder act in conjunction with each other to read a bar code.
17. How does an ultrasonic sensor operate?
18. Explain the principle of operation of a strain gauge.
19. Explain the principle of operation of a thermocouple.
20. What is the most common approach taken with regard to the measurement of fluid flow?
21. Explain how a tachometer is used to measure rotational speed.
22. How does an optical encoder work?
23. Draw an electrical symbol used to represent each of the following PLC control devices:
   a. Pilot light
   b. Relay
   c. Motor starter coil
   d. OL relay contact
   e. Alarm
   f. Heater
   g. Solenoid
   h. Solenoid valve
   i. Motor
   j. Horn
24. Explain the function of each of the following actuators:
   a. Solenoid
   b. Solenoid valve

CHAPTER 6 PROBLEMS

1. Design and draw the schematic for a conventional hardwired relay circuit that will perform each of the following circuit functions when a normally closed pushbutton is pressed:
   - Switch a pilot light on
   - De-energize a solenoid
   - Start a motor running
   - Sound a horn
2. Design and draw the schematic for a conventional hardwired circuit that will perform the following circuit functions using two break-make pushbuttons:
   - Turn on light L1 when pushbutton PB1 is pressed.
   - Turn on light L2 when pushbutton PB2 is pressed.
   - Electrically interlock the pushbuttons so that L1 and L2 cannot both be turned on at the same time.
3. Study the ladder logic program in Figure 6-73, and answer the questions that follow:
   a. Under what condition will the latch rung 1 be true?
   b. Under what conditions will the unlatch rung 2 be true?
   c. Under what condition will rung 3 be true?
   d. When PL1 is on, the relay is in what state (latched or unlatched)?
   e. When PL2 is on, the relay is in what state (latched or unlatched)?
   f. If AC power is removed and then restored to the circuit, what pilot light will automatically come on when the power is restored?
   g. Assume the relay is in its latched state and all three inputs are false. What input change(s) must occur

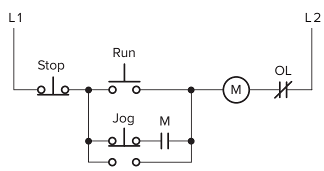
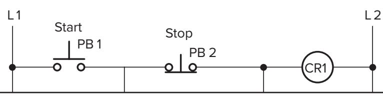


Figure 6-74 Hardwired control circuit for Problem 4.

will correctly execute the hardwired control circuit in Figure 6-74.
Assume: Stop pushbutton used is an NO type.
Run pushbutton used is an NO type.
Jog pushbutton used has one set of NO contacts.
OL contact is hardwired.

5. Design a PLC program and prepare a typical I/O connection diagram and ladder logic program that will correctly execute the hardwired control circuit in Figure 6-75.
   Assume: PB1 pushbutton used is an NO type.
   PB2 pushbutton used is an NC type.
   PS1 pressure switch used is an NO type.
   LS1 limit switch used has only one set of NC contacts.



EXAMPLE PROBLEMS help bring home the applicability of chapter concepts

ANCILLARIES THAT WORK

Expanded on and updated from the previous edition, this new edition includes an outstanding instructor support package available through Connect:

- EZ Test question test banks for each chapter.
- Lab Manual question banks for several chapters.
- Lecture PowerPoints.
- Answers to the questions and problems in the textbook, LogixPro Manual, and RSLogix 5000 Manual.

Proctorio

Remote Proctoring & Browser-Locking Capabilities

Remote proctoring and browser-locking capabilities, hosted by Proctorio within Connect, provide control of the assessment environment by enabling security options and verifying the identity of the student.

Seamlessly integrated within Connect, these services allow instructors to control students' assessment experience by restricting browser activity, recording students' activity, and verifying students are doing their own work.

Instant and detailed reporting gives instructors an at-a-glance view of potential academic integrity concerns, thereby avoiding personal bias and supporting evidence-based claims.

ReadAnywhere

Read or study when it's convenient for you with McGraw Hill's free ReadAnywhere app. Available for iOS or Android smartphones or tablets, ReadAnywhere gives users access to McGraw Hill tools including the eBook and SmartBook 2.0 or Adaptive Learning Assignments in Connect. Take notes, highlight, and complete assignments offline – all of your work will sync when you open the app with WiFi access. Log in with your McGraw Hill Connect username and password to start learning – anytime, anywhere!

Connect Course Builds and Edition to Edition Course Rebuilds

Adopters of McGraw Hill content and technology are invited to take advantage of our Connect course build and rebuild services. Instructors who are using Connect for the first time can share their syllabus with McGraw Hill, and McGraw Hill will assist with the initial build of SmartBook readings, assignments, and assessments that align with the instructor's course objectives. If an instructor is already using Connect, when a new edition is published, McGraw Hill is prepared to assist with the transition of the instructor's Connect course to the new edition. McGraw Hill will rebuild the instructor's current Connect course by recreating assignments, mimicking all questions, point values and policy settings, and adding new quiz and test questions to reflect updated content. When the new course is complete, McGraw Hill's Implementation Team is available to review the course with the instructor and highlight new features in Connect.

OLC-Aligned Courses

Implementing High-Quality Online Instruction and Assessment through Preconfigured Courseware

For select titles, in consultation with the Online Learning Consortium (OLC) and our certified Faculty Consultants, McGraw Hill has created pre-configured courseware using OLC's quality scorecard to align with best practices in online course delivery. This turnkey courseware contains a combination of formative assessments, summative assessments, homework, and application activities, and can easily be customized to meet an individual's needs and course outcomes. For more information, visit https://www.mheducation.com/highered/olc.

Tegrity: Lectures 24/7

Tegrity in Connect is a tool that makes class time available 24/7 by automatically capturing every lecture. With a simple one-click start-and-stop process, you capture all computer screens and corresponding audio in a format that is easy to search, frame by frame. Students can replay any part of any class with easy-to-use, browser-based viewing on a PC, Mac, or any mobile device.

Educators know that the more students can see, hear, and experience class resources, the better they learn. In fact, studies prove it. Tegrity's unique search feature helps students efficiently find what they need, when they need it, across an entire semester of class recordings. Help turn your students' study time into learning moments immediately supported by your lecture. With Tegrity, you also increase intent listening and class participation by easing students' concerns about note-taking. Using Tegrity in Connect will make it more likely you will see students' faces, not the tops of their heads.

Test Builder in Connect

Available within Connect, Test Builder is a cloud-based tool that enables instructors to format tests that can be printed, administered within a Learning Management System, or exported as a Word document of the test bank. Test Builder offers a modern, streamlined interface for easy content configuration that matches course needs, without requiring a download.

Test Builder allows you to:

- access all test bank content from a particular title.
- easily pinpoint the most relevant content through robust filtering options.
- manipulate the order of questions or scramble questions and/or answers.
- pin questions to a specific location within a test.
- determine your preferred treatment of algorithmic questions.
- choose the layout and spacing.
- add instructions and configure default settings.

Test Builder provides a secure interface for better protection of content and allows for just-in-time updates to flow directly into assessments.

Writing Assignment

Available within Connect and Connect Master, the Writing Assignment tool delivers a learning experience to help students improve their written communication skills and conceptual understanding. As an instructor you can assign, monitor, grade, and provide feedback on writing more efficiently and effectively.

Application-Based Activities in Connect

Application-Based Activities in Connect are highly interactive, assignable exercises that provide students with a safe space to apply the concepts they have learned to real-world, course-specific problems. Each Application-Based Activity involves the application of multiple concepts, allowing students to synthesize information and use critical thinking skills to solve realistic scenarios.

Create

Your Book, Your Way

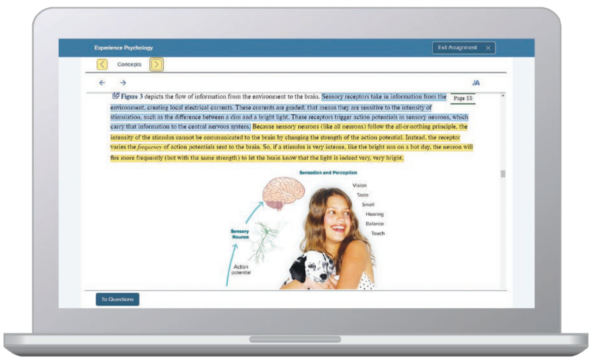
McGraw Hill's Content Collections Powered by Create® is a self-service website that enables instructors to create custom course materials—print and eBooks—by drawing upon McGraw Hill's comprehensive, cross-disciplinary content. Choose what you want from our high-quality textbooks, articles, and cases. Combine it with your own content quickly and easily, and tap into other rights-secured, third-party content such as readings, cases, and articles. Content can be arranged in a way that makes the most sense for your course and you can include the course name and information as well. Choose the best format for your course: color print, black-and-white print, or eBook. The eBook can be included in your Connect course and is available on the free ReadAnywhere app for smartphone or tablet access as well. When you are finished customizing, you will receive a free digital copy to review in just minutes! Visit McGraw Hill Create®—www.mcgrawhillcreate.com—today and begin building!

Instructors: Student Success Starts with You

Tools to enhance your unique voice

Want to build your own course? No problem. Prefer to use an OLC-aligned, prebuilt course? Easy. Want to make changes throughout the semester? Sure. And you'll save time with Connect's auto-grading too.

65%
Less Time Grading



Laptop: McGraw Hill; Woman/dog: George Doyle/Getty Images

Study made personal

Incorporate adaptive study resources like SmartBook® 2.0 into your course and help your students be better prepared in less time. Learn more about the powerful personalized learning experience available in SmartBook 2.0 at **www.mheducation.com/highered/connect/smartbook**

Affordable solutions, added value

Make technology work for you with LMS integration for single sign-on access, mobile access to the digital textbook, and reports to quickly show you how each of your students is doing. And with our Inclusive Access program you can provide all these tools at a discount to your students. Ask your McGraw Hill representative for more information.

Padlock: Jobalou/Getty Images

Solutions for your challenges

A product isn't a solution. Real solutions are affordable, reliable, and come with training and ongoing support when you need it and how you want it. Visit **www.supportateverystep.com** for videos and resources both you and your students can use throughout the semester.

Checkmark: Jobalou/Getty Images

SUPPORT AT every step

Students: Get Learning that Fits You

Effective tools for efficient studying

Connect is designed to help you be more productive with simple, flexible, intuitive tools that maximize your study time and meet your individual learning needs. Get learning that works for you with Connect.

Study anytime, anywhere

Download the free ReadAnywhere app and access your online eBook, SmartBook 2.0, or Adaptive Learning Assignments when it's convenient, even if you're offline. And since the app automatically syncs with your Connect account, all of your work is available every time you open it. Find out more at **www.mheducation.com/readanywhere**

"I really liked this app—it made it easy to study when you don't have your text-book in front of you."

- Jordan Cunningham,
Eastern Washington University

Calendar: owattaphotos/Getty Images

Everything you need in one place

Your Connect course has everything you need—whether reading on your digital eBook or completing assignments for class, Connect makes it easy to get your work done.

Learning for everyone

McGraw Hill works directly with Accessibility Services Departments and faculty to meet the learning needs of all students. Please contact your Accessibility Services Office and ask them to email accessibility@mheducation.com, or visit **www.mheducation.com/about/accessibility** for more information.

Top: Jenner Images/Getty Images, Left: Hero Images/Getty Images, Right: Hero Images/Getty Images

1

Programmable Logic Controllers (PLCs): An Overview

Image Courtesy of Rockwell Automation, Inc.

Chapter Objectives

After completing this chapter, you will be able to:

- Define what a programmable logic controller (PLC) is and list its advantages over relay systems
- Identify the main parts of a PLC and describe their functions
- Outline the basic sequence of operation for a PLC
- Identify the general classifications of PLCs

This chapter gives a brief history of the evolution of the programmable logic controller, or PLC. The reasons for changing from relay control systems to PLCs are discussed. You will learn the basic parts of a PLC, how a PLC is used to control a process, and the different kinds of PLCs and their applications. The ladder logic language, which was developed to simplify the task of programming PLCs, is introduced.

1.1 Programmable Logic Controllers

Programmable logic controllers are now the most widely used industrial process control technology. A **programmable logic controller (PLC)** is an industrial grade computer that is capable of being programmed to perform control functions. The programmable controller has eliminated much of the hardwiring associated with conventional relay control circuits. Other benefits include fast response, easy programming and installation, high control speed, network compatibility, troubleshooting and testing convenience, and high reliability.

The PLC is designed for multiple input and output arrangements, extended temperature ranges, immunity to electrical noise, and resistance to vibration and impact. Programs for the control and operation of manufacturing process equipment and machinery are typically stored in battery-backed or nonvolatile memory. A PLC is an example of a **real-time system** since the output of the system controlled by the PLC depends on the input conditions.

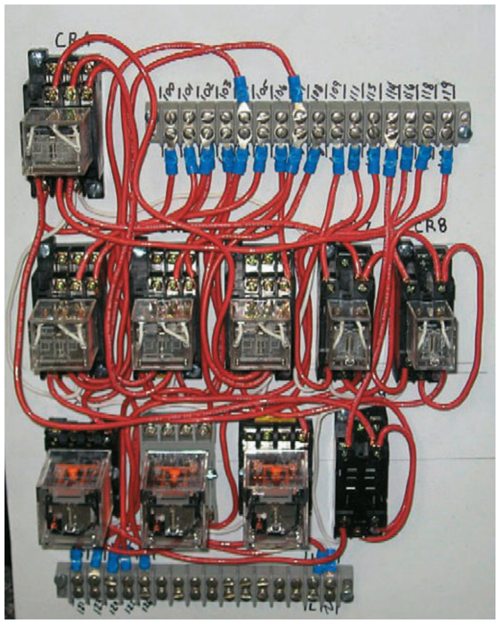
The PLC is, then, basically a digital computer designed for use in machine control. Unlike a personal computer, it has been designed to operate in the industrial environment and is equipped with special input/output interfaces and a control programming language.

Initially the PLC was used to replace **relay logic**, but its ever-increasing range of functions means that it is found in many and more complex applications. Because the structure of a PLC is based on the same principles as those employed in computer architecture, it is capable not only of performing relay switching tasks but also of performing other applications such as timing, counting, calculating, comparing, and the processing of analog signals.

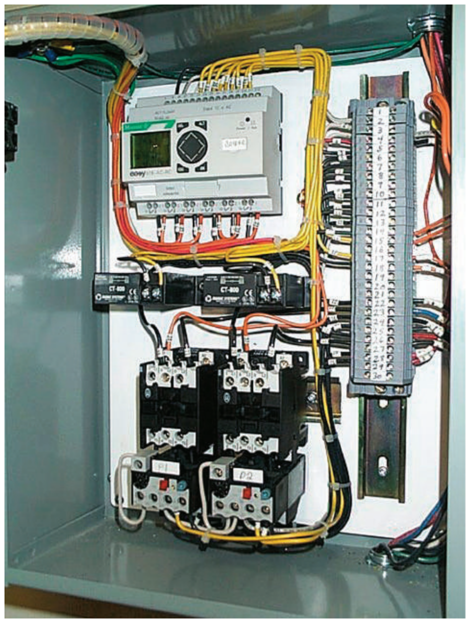
Programmable controllers offer several advantages over a conventional relay type of control. Relays have to be hardwired to perform a specific function. When the system requirements change, the relay wiring has to be changed or modified. In extreme cases, such as in the auto industry, complete control panels had to be replaced since it was not economically feasible to rewire the old panels with each model changeover. The programmable controller has eliminated much of the hardwiring associated with conventional relay control circuits (Figure 1-1). It is small and inexpensive compared to equivalent relay-based process control systems. Modern control systems still include relays, but these are rarely used for logic.

PLCs provide many other benefits including:

- ***Increased Reliability.*** Once a program has been written and tested, it can be easily downloaded to other PLCs. Since all the logic is contained in the PLC's memory, there is no chance of making a logic wiring error (Figure 1-2). The program takes the



(*a*)



(*b*)

Figure 1-1 Relay- and PLC-based control panels. (*a*) Relay-based control panel. (*b*) PLC-based control panel.
Source: (*a*) Courtesy Mid-Illini Technical Group, Inc.; (*b*) Photo courtesy of Ramco Electric Ltd., Toronto

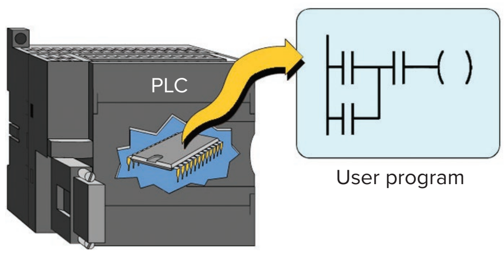


Figure 1-2 All the logic is contained in the PLC's memory.

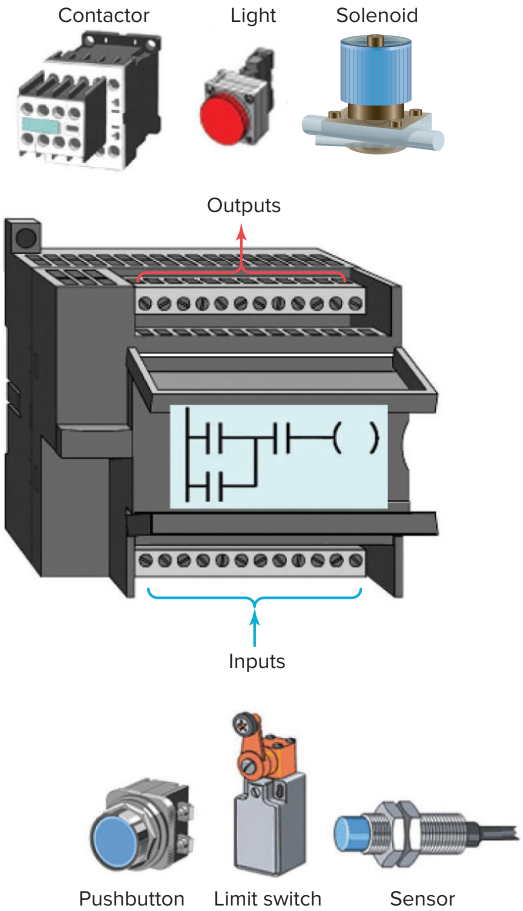


Figure 1-3 Relationships between the inputs and outputs are determined by the user program.

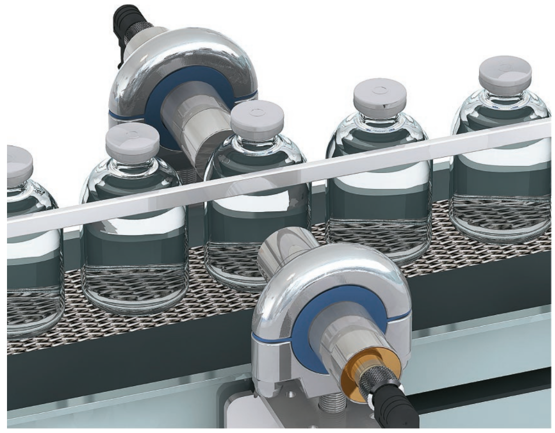


Figure 1-4 High-speed counting.
Source: Courtesy of Banner Engineering Corp.

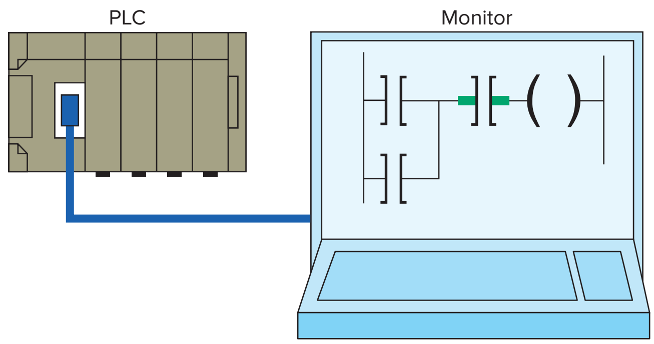


Figure 1-5 Control program can be displayed on a monitor in real time.

place of much of the external wiring that would normally be required for control of a process. Hardwiring, though still required to connect field devices, is less intensive. PLCs also offer the reliability associated with solid-state components.

- ***More Flexibility.*** It is easier to create and change a program in a PLC than to wire and rewire a circuit. With a PLC the relationships between the inputs and outputs are determined by the user program instead of the manner in which they are interconnected (Figure 1-3). Original equipment manufacturers can provide system updates by simply sending out a new program. End users can modify the program in the field, or if desired, security can be provided by hardware features such as key locks and by software passwords.
- ***Lower Cost.*** PLCs were originally designed to replace relay control logic, and the cost savings have been so significant that relay control is becoming obsolete except for power applications. Generally, if an application has more than about a half-dozen control relays, it will probably be less expensive to install a PLC.
- ***Communications Capability.*** A PLC can communicate with other controllers or computer equipment to perform such functions as supervisory control, data gathering, monitoring devices and process parameters, and download and upload of programs.
- ***Faster Response Time.*** PLCs are designed for high-speed and real-time applications (Figure 1-4). The programmable controller operates in real time, which means that an event taking place in the field will result in the execution of an operation or output. Machines that process thousands of items per second and objects that spend only a fraction of a second in front of a sensor require the PLC's quick-response capability.
- ***Easier to Troubleshoot.*** PLCs have resident diagnostics and override functions that allow users to easily trace and correct software and hardware problems. To find and fix problems, users can display the control program on a monitor and watch it in real time as it executes (Figure 1-5)·

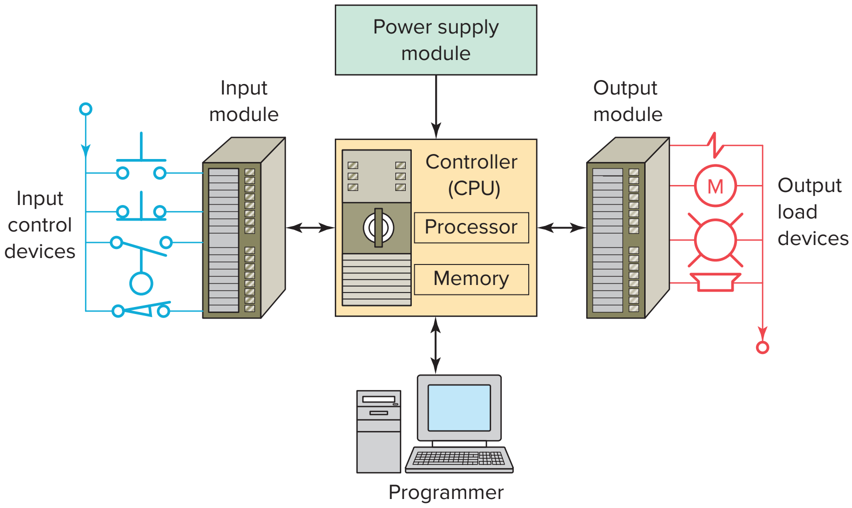


Figure 1-6 Typical parts of a programmable logic controller.

- **Easier to Test Field Devices.** A PLC control panel has the ability to check field devices at a common point. For example, a control system consisting of hundreds of input and output field devices may be contained within a very large manufacturing area. Thus, it would take a considerable amount of time to check each device at its location. By having each device wired back to a common point on a PLC module, each device could be checked for operation fairly quickly.

1.2 Parts of a PLC

A typical PLC can be divided into parts, as illustrated in Figure 1-6. These are the *central processing unit (CPU)*, the *input/output (I/O)* section, the *power supply,* and the *programming device.* The term *architecture* can refer to PLC hardware, to PLC software, or to a combination of both. An *open* architecture design allows the system to be connected easily to devices and programs made by other manufacturers. Open architectures use off-the-shelf components that conform to approved standards. A system with a *closed* architecture is one whose design is proprietary, making it more difficult to connect to other systems. Most PLC systems are in fact **proprietary**, so you must be sure that any generic hardware or software you may use is compatible with your particular PLC. Also, although the principal concepts are the same in all methods of programming, there are differences in addressing, memory allocation, retrieval, and data handling for different models. Consequently, PLC programs cannot be interchanged among different PLC manufacturers.

There are two ways in which I/Os (Inputs/Outputs) are incorporated into the PLC: fixed and modular. ***Fixed I/O*** (Figure 1-7) is typical of small PLCs that come in one package with no separate, removable units. The processor and I/O are packaged together, and the I/O terminals will have a fixed number of connections built in for inputs and outputs. The main advantage of this type of packaging is lower cost. The number of available I/O points varies and usually can be expanded by buying additional units of fixed I/O. One disadvantage of fixed I/O is its lack of flexibility; you are limited in what you can get in the quantities and types dictated by the packaging. Also, for some models, if any part in the unit fails, the whole unit has to be replaced.

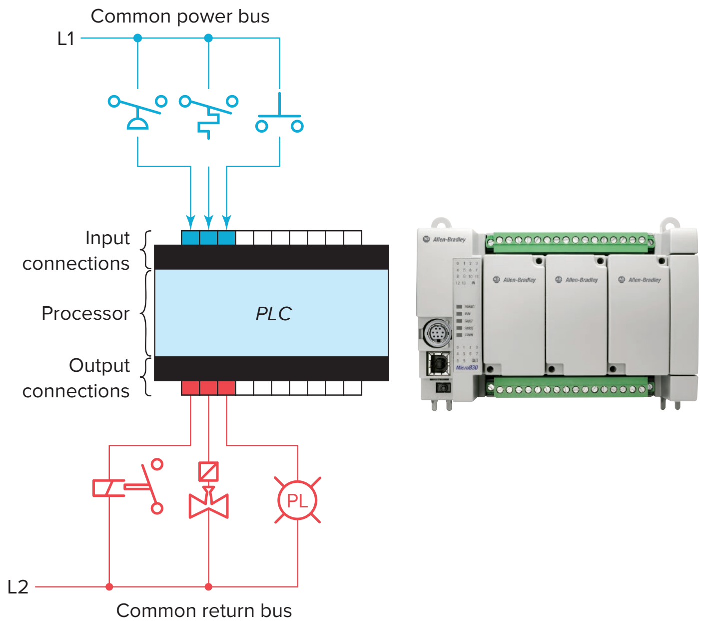


Figure 1-7 Fixed I/O configuration.
Source: Image Courtesy of Rockwell Automation, Inc

Modular I/O (Figure 1-8) is divided by compartments into which separate modules can be plugged. This feature greatly increases your options and the unit's flexibility. You can choose from the modules available from the manufacturer and mix them any way you desire. The basic modular controller consists of a rack, power supply, processor module (CPU), input/output (I/O modules), and an operator interface for programming and monitoring. The modules plug into a rack. When a module is slid into the rack, it makes an electrical connection with a series of contacts called the backplane, located at the rear of the rack. Communication between modules is accomplished by the backplane rail that each module plugs into.

The ***power supply*** supplies DC power through the backplane, to the processor and the other modules that plug into the rack (Figure 1-9). For large PLC systems, this power supply does not normally supply power to the field devices. With most systems, power to field devices is provided by external alternating current (AC) or direct current (DC) supplies. For some small micro PLC systems, the power supply may be used to power field devices.

The ***processor*** **(CPU)** is the "brain" of the PLC. A typical processor usually consists of a microprocessor for implementing the logic and controlling the communications among the modules. The processor requires memory for storing user program instructions, numerical values, and I/O devices status.

The CPU controls all PLC activity and is responsible for running the program. The PLC program is executed as part of a repetitive process referred to as a scan (Figure 1-10). A typical PLC scan starts with the CPU reading the status of inputs. Then, the program logic is executed. Once the

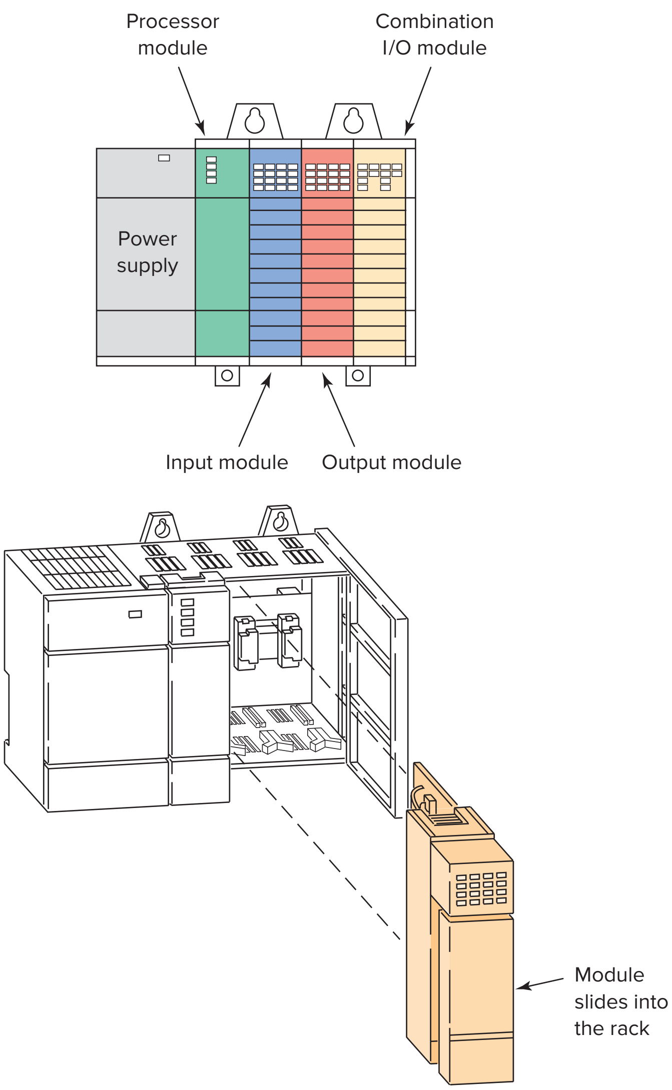


Figure 1-8 Modular I/O configuration.

Figure 1-9 The power supply supplies DC power to other modules that plug into the rack.
Source: Photo of PLC Modicon M340 © Schneider Electric, 2010. **www.schneider-electric.com**.

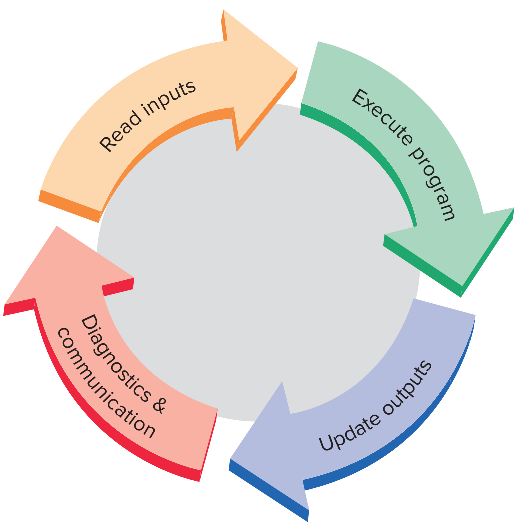


Figure 1-10 Typical PLC scan cycle.

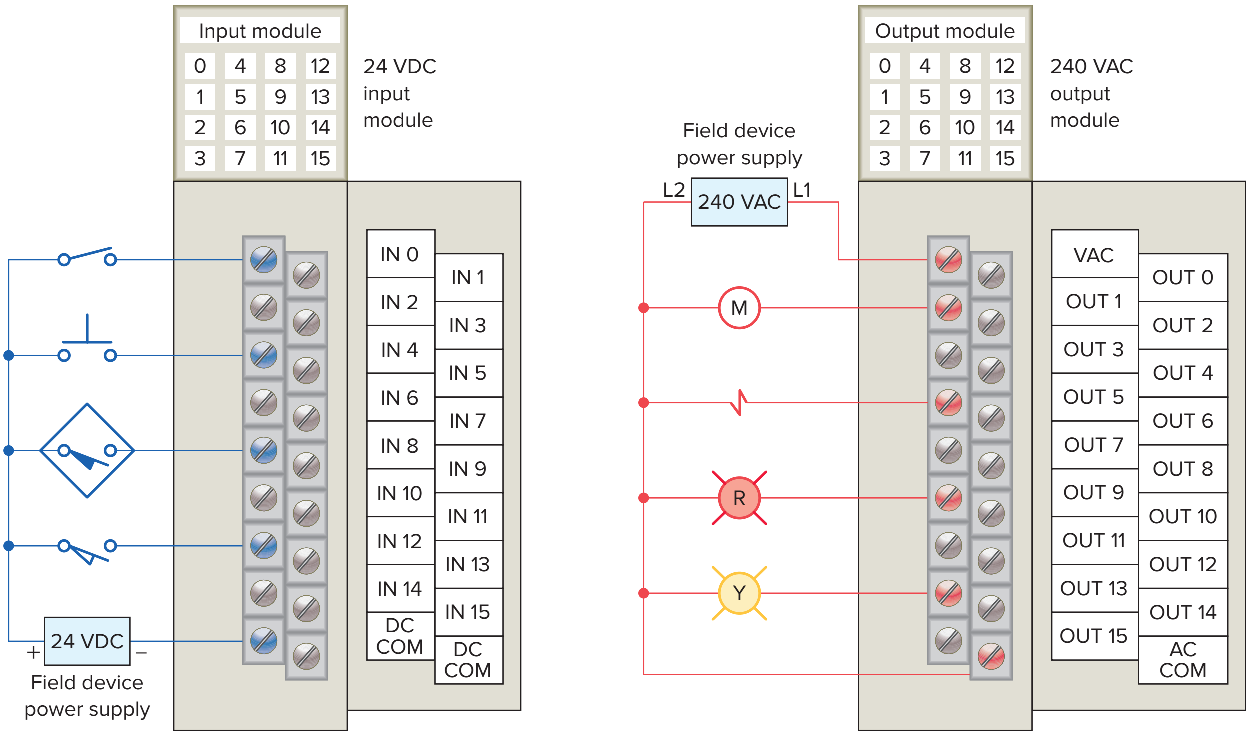


Figure 1-11 Typical PLC input/output (I/O) system connections.

program execution is completed, the status of all outputs is updated. Next, the CPU performs internal diagnostic and communication tasks. This process is repeated continuously as long as the PLC is in the run mode.

The ***I/O system*** forms the interface by which field devices are connected to the controller (Figure 1-11). The purpose of this interface is to condition the various signals received from or sent to external field devices. Input devices such as pushbuttons, limit switches, and sensors are hardwired to the input terminals. Output devices such as small motors, motor starters, solenoid valves, and indicator lights are hardwired to the output terminals. To electrically isolate the internal components from the input and output terminals, PLCs commonly employ an optical isolator, which uses light to couple the circuits together. The external devices are also referred to as "field" or "real-world" inputs and outputs. The terms ***field*** or ***real world*** are used to distinguish actual external devices that exist and must be physically wired from the internal user program that emulates the function of relays, timers, and counters.

A ***programming device*** is used to interface with the PLC in order to develop and transfer logic programs, download or upload data, or supply diagnostic functions to troubleshoot PLC systems. The device may be a dedicated handheld type or a personal computer running special application software. Removing the programming device will not affect the operation of the program.

A laptop computer is the most commonly used programming device. Most brands of PLCs have software available so that a laptop can be used as the programming device. This software allows users to create, edit, document, store, and troubleshoot ladder logic programs. The computer monitor is able to display more logic on the screen than can hand-held types, thus simplifying the interpretation of the program. The programming device communicates with the PLC processor via a serial or parallel data communications link, or Ethernet.

The logic ***program*** is a user-developed series of instructions that directs the PLC to execute actions. A *programming language* provides rules for combining the instructions so that they produce the desired actions. ***Relay ladder logic (RLL)*** is the standard programming language used with PLCs. Its origin is based on electromechanical relay control. The relay ladder logic program graphically represents rungs of contacts, coils, and special instruction blocks. RLL was originally designed for easy use and understanding for its users and has been modified to keep up with the increasing demands of industry's control needs.

1.3 Principles of Operation

To get an idea of how a PLC operates, consider the simple process control problem illustrated in Figure 1-12. Here a mixer motor is to be used to automatically stir the liquid in a vat when the temperature and pressure reach

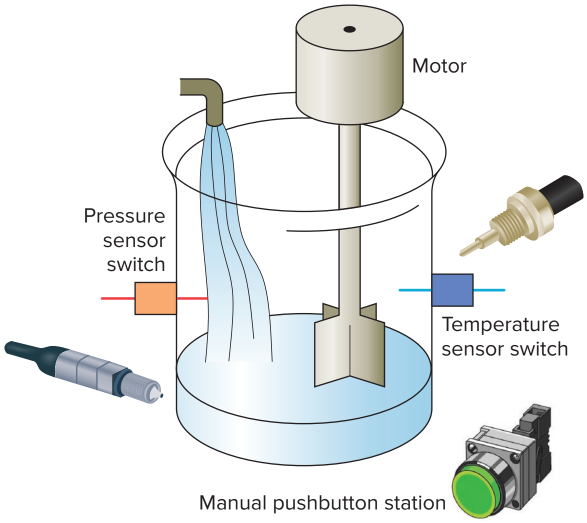


Figure 1-12 Mixer process control problem.

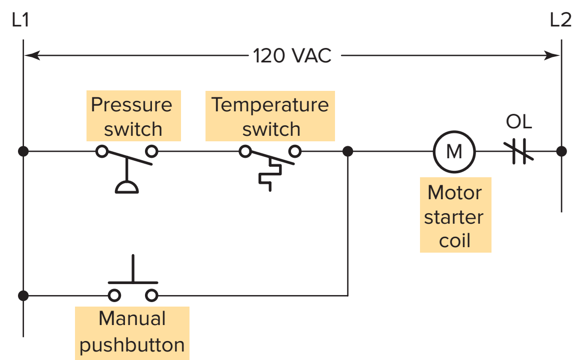


Figure 1-13 Process control relay ladder diagram.

preset values. In addition, direct manual operation of the motor is provided by means of a separate pushbutton station. The process is monitored with temperature and pressure sensor switches that close their respective contacts when conditions reach their preset values.

This control problem can be solved using the relay method for motor control shown in the relay ladder diagram of Figure 1-13. The motor starter coil (M) is energized when both the pressure and temperature switches are closed or when the manual pushbutton is pressed.

Now let's look at how a programmable logic controller might be used for this application. The same input field devices (pressure switch, temperature switch, and pushbutton) are used. Each of these devices is wired to a terminal on the 120 VAC input module as shown in Figure 1-14. The processor memory location addresses used are:

I/1 for the Pressure switch
I/2 for the Temperature switch
I/3 for the Manual pushbutton

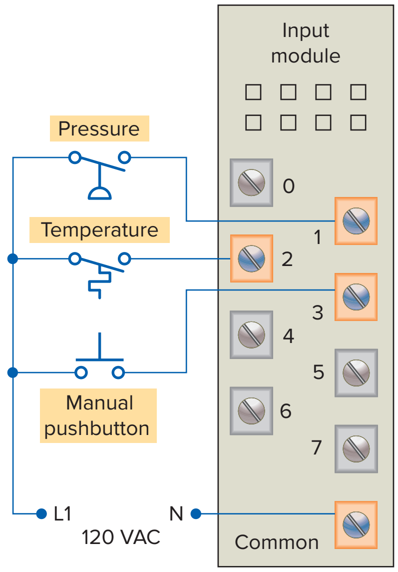


Figure 1-14 Typical wiring connections for a 120 VAC modular configured input module.

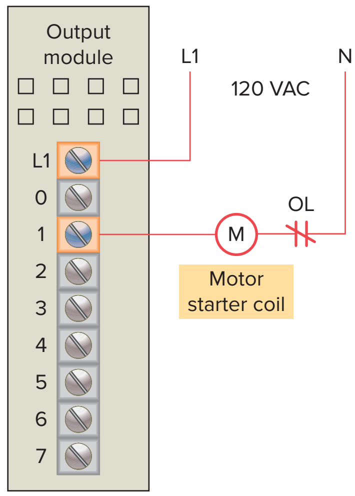


Figure 1-15 Typical wiring connections for a 120 VAC modular configured output module.

The same output field device (motor starter coil) would also be used. This device is wired to a terminal on the 120 VAC output module as shown in Figure 1-15. The processor memory location address used for the Motor starter coil is O/1.

Next, the PLC ladder logic program would be constructed and entered into the memory of the CPU. A typical ladder logic program for this process is shown in Figure 1-16. The format used is similar to the layout of the hardwired relay ladder circuit. The individual symbols represent instructions, whereas the numbers represent the instruction memory location addresses. To program the controller, you enter these instructions one by one into the processor memory from the programming

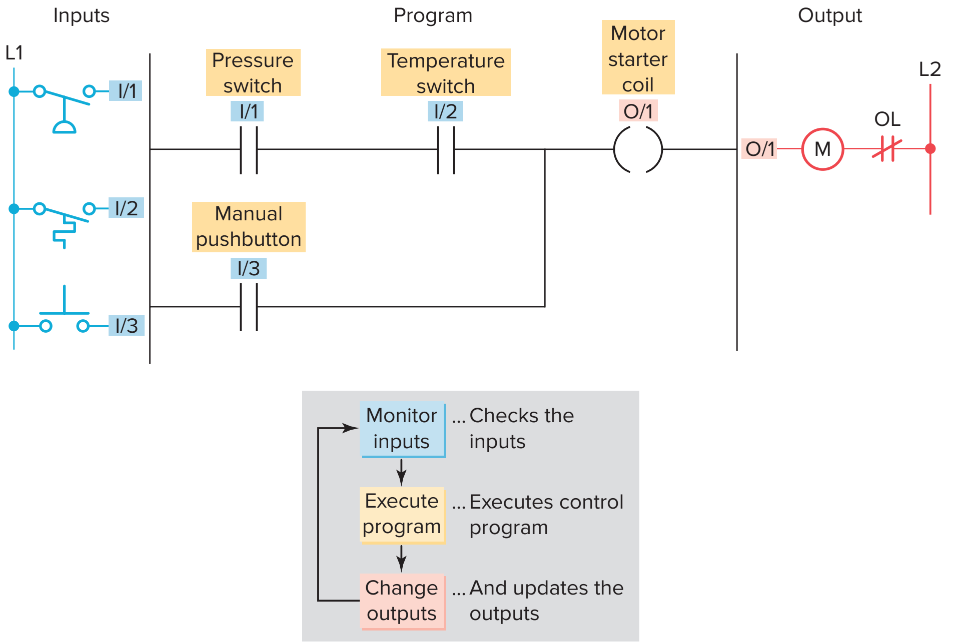


Figure 1-16 Process control PLC ladder logic program with typical addressing scheme.

device. Each input and output device is given an address, which lets the PLC know where it is physically connected. Note that the I/O address format will differ, depending on the PLC model and manufacturer. Instructions are stored in the user program portion of the processor memory. During the program scan the controller monitors the inputs, executes the control program, and changes the output accordingly.

For the program to operate, the controller is placed in the RUN mode, or operating cycle. During the program scan, the controller monitors the inputs, executes the control program, and changes the output accordingly. Each ⊣⊢ symbol (looks like a normally open contact) is an instruction. The ( ) symbol is considered to represent a coil that, when energized, will energize the device that is wired to the respective output. In the ladder logic program of Figure 1-16, the coil O/1 is energized when contacts I/1 and I/2 are closed or when contact I/3 is closed. Either of these conditions provides a continuous logic path from left to right across the rung that includes the coil.

A programmable logic controller operates in real time in that an event taking place in the field will result in an operation or output taking place. The RUN operation for the process control logic can be described by the following sequence of events:

- First, the pressure switch, temperature switch, and pushbutton inputs are examined and their status is recorded in the controller's memory.
- A closed contact is recorded in memory as logic 1 and an open contact as logic 0.
- Next the ladder diagram is evaluated, with each internal contact given an OPEN or CLOSED status according to its recorded 1 or 0 state.
- When the states of the input contacts provide logic continuity from left to right across the rung, the output coil memory location is given a logic 1 value and the output module interface contacts will close.
- When there is no logic continuity of the program rung, the output coil memory location is set to logic 0 and the output module interface contacts will be open.
- The completion of one cycle of this sequence by the controller is called a *scan.* The scan time, the time required for one full cycle, provides a measure of the speed of response of the PLC.
- Generally, the output memory location is updated during the scan but the actual output is not updated until the end of the program scan during the I/O scan.

Figure 1-17 shows the typical wiring required to implement the process control scheme using a fixed PLC controller. In this example, the Allen-Bradley Pico controller equipped with 8 inputs and 4 outputs is used to control and monitor the process. Installation can be summarized as follows:

- Fused power lines, of the specified voltage type and level, are connected to the controller's L1 and L2 terminals.
- The pressure switch, temperature switch, and pushbutton field input devices are hardwired between L1

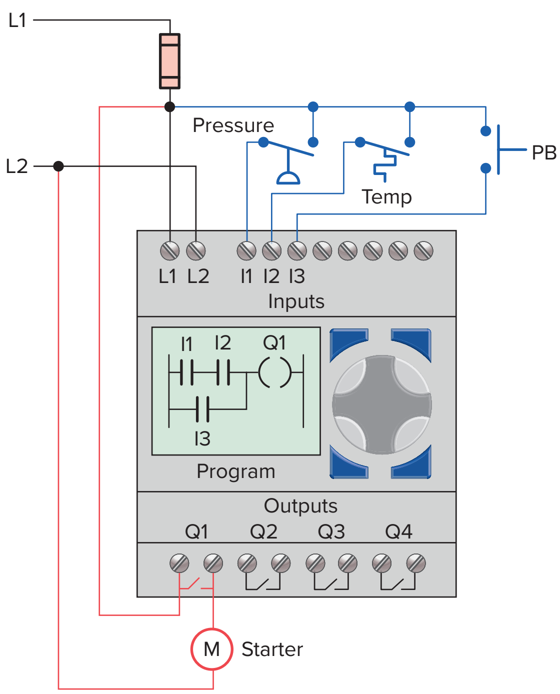


Figure 1-17 Typical wiring required to implement the process control scheme using a fixed PLC controller.

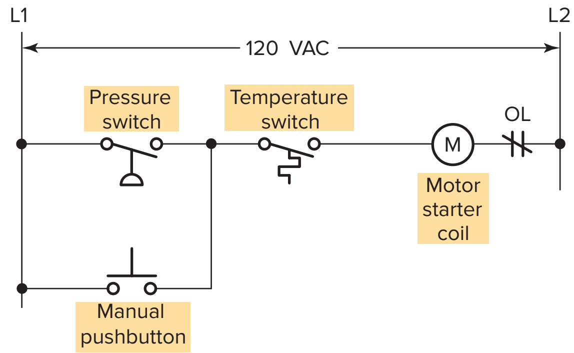


Figure 1-18 Relay ladder diagram for the modified process.

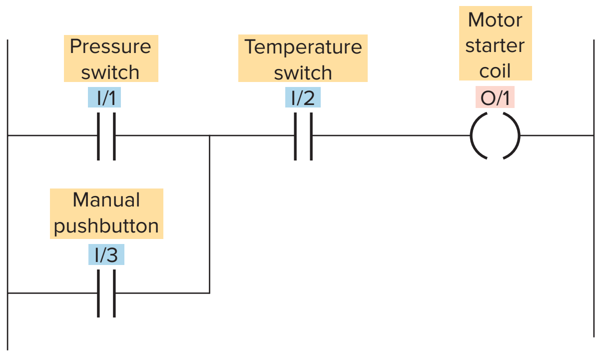


Figure 1-19 PLC ladder logic program for the modified process.

and controller input terminals I1, I2, and I3, respectively.

- The motor starter coil connects directly to L2 and in series with Q1 relay output contacts to L1.
- The ladder logic program is entered using the front keypad and LCD display.
- Pico programming software is also available that allows you to create as well as test your program using a personal computer.

1.4 Modifying the Operation

One of the important features of a PLC is the ease with which the program can be changed. For example, assume that the original process control circuit for the mixing operation must be modified as shown in the relay ladder diagram of Figure 1-18. The change requires that the manual pushbutton control be permitted to operate at any pressure, but not unless the specified temperature setting has been reached.

If a relay system were used, it would require some rewiring of the circuit shown in Figure 1-18 to achieve the desired change. However, if a PLC system were used, no rewiring would be necessary. The inputs and outputs are still the same. All that is required is to change the PLC ladder logic program as shown in Figure 1-19.

At times, a process may call for additional real input or output field devices to be added to the circuit. The

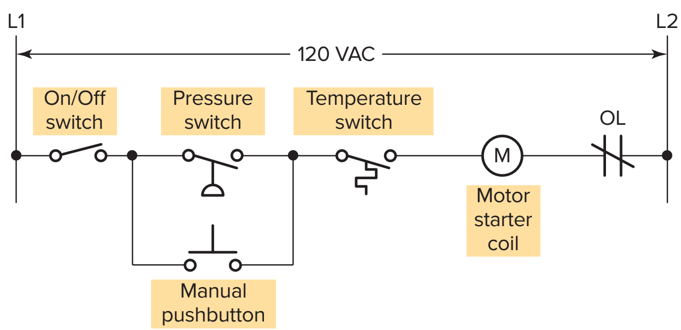


Figure 1-20 Modified relay ladder for the addition of an ON/OFF switch.

relay ladder diagram of Figure 1-20 shows the circuit further modified to include a process ON/OFF switch. To accomplish this, using hard-wired circuit control requires accessing both the pressure switch and manual pushbutton and rewiring the circuit so that the two are in parallel with each other and in series with the ON/OFF switch. The modification implemented using a PLC ladder logic program is shown in Figure 1-21. Note that the original wiring of the existing PLC input circuit remains the same. All that is required is the connection of the ON/OFF switch to I/4 input and the related change in the PLC program.

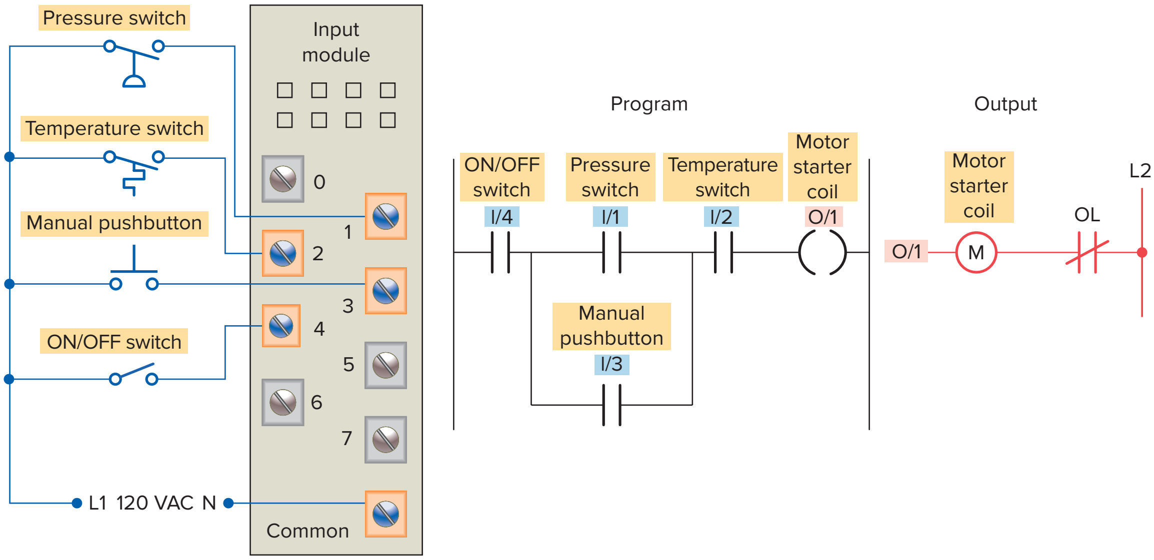


Figure 1-21 Modified PLC ladder logic program and ON/OFF switch module connection.

1.5 PLCs versus Computers

The architecture of a PLC is basically the same as that of a personal computer (PC). Characteristics that distinguish PLCs from PC include:

- ***Hardware.*** Unlike PCs, the PLC has no permanently attached keyboard, CD drive, or monitor.
- ***Operating Environment.*** PLCs are designed to operate in the industrial environment a wide range of ambient temperature, humidity, and electrical noise factors.
- ***Programing.*** The PLC is programmed in relay ladder logic or four other types of programming languages and comes with its program language built into its memory.
- ***Program Execution.*** PLCs execute a single program in a sequential order, while computers execute several programs or tasks simultaneously in any order.

Software associated with a PLC but written and run on a personal computer falls into the following two broad categories:

- PLC software that allows the user to program and document gives the user the tools to write a PLC program—using ladder logic or another programming language—and document or explain the program in as much detail as is necessary.
- PLC software that allows the user to monitor and control the process is called a ***human machine interface (HMI)***. It enables the user to view a process—or a graphical representation of a process—on a monitor, operate the machine, trend values, and receive alarm conditions (Figure 1-22). PLCs can be integrated with HMIs but the same software does not program both devices.

Most recently automation manufacturers have responded to the increased requirements of industrial control systems by blending the advantages of PLC-style control with that of PC-based systems. Such a device has been termed a *programmable automation controller,* or *PAC*. Programmable automation controllers combine PLC ruggedness with PC functionality. Using PACs, you can build advanced systems incorporating software capabilities such as advanced control, communication, data logging, and signal processing with rugged hardware performing logic, motion, process control, and vision. One main difference between PLC- and PAC-based programs is how the program is executed. A PLC mixes scan-based and event-driven program execution, whereas PAC software is typically event-driven. The scan-based execution of a PLC

Figure 1-22 Human Machine Interface (HMI).
Source: Image Courtesy of Rockwell Automation, Inc.

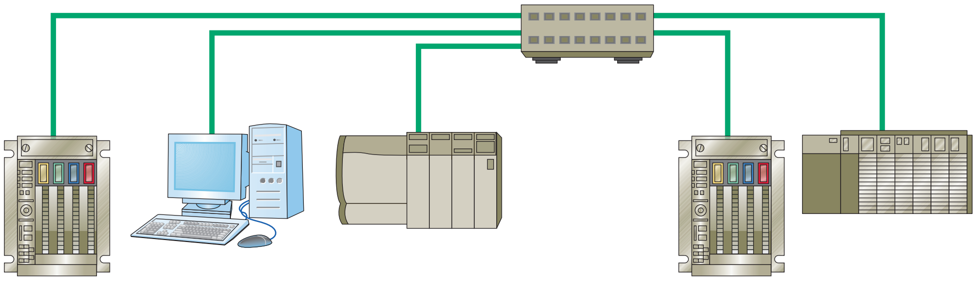


Figure 1-23 Control management PLC application.

program might take longer because the system needs to complete the higher priority actions in the cycle first.

1.6 PLC Size and Application

The criteria used in categorizing PLCs include functionality, number of inputs and outputs, cost, and physical size. Of these, the ***I/O count*** is the most important factor.

There are three major types of PLC application: single-ended, multitask, and control management. A ***single-ended*** or **stand-alone** PLC application involves one PLC controlling one process. This would be a stand-alone unit and would not be used for communicating with other computers or PLCs.

A ***multitask*** PLC application involves one PLC controlling several processes. Adequate I/O capacity is a significant factor in this type of installation. In addition, if the PLC would be a subsystem of a larger process and would have to communicate with a central PLC or computer, provisions for a data communications network are also required.

A ***control management*** PLC application involves one PLC controlling several others (Figure 1-23). This kind of application requires a large PLC processor designed to communicate with other PLCs and possibly with a computer. The control management PLC supervises several PLCs by downloading programs that tell the other PLCs what has to be done. It must be capable of connection to all PLCs so that by proper addressing it can communicate with any one it wishes to.

Memory is the part of a PLC that stores data, instructions, and the control program. Memory size is usually expressed in K values: 1 K, 6 K, 12 K, and so on. The measurement kilo, abbreviated K, normally refers to 1000 units. When dealing with computer or PLC memory, however, 1 K means 1024, because this measurement is based on the binary number system ($2^{10} = 1024$).

Although it is common for us to measure the memory capacity of PLCs in words, we need to know the number of bits in each word before memory size can be accurately compared. Modern computers usually have a word size of 16, 32, or 64 bits. The amount of memory required depends on the application. Factors affecting the memory size needed for a particular PLC installation include:

- Number of I/O points used
- Size of control program
- Data-collecting requirements
- Supervisory functions required
- Future expansion

The ***instruction set*** for a particular PLC lists the different types of instructions supported. Typically, this ranges from 15 instructions on smaller units up to 100 instructions on larger, more powerful units (see Table 1-1).

Table 1-1 Typical PLC Instructions

Instruction	Operation
XIC (Examine ON)	Examine a bit for an ON or 1 condition
XIO (Examine OFF)	Examine a bit for an OFF or 0 condition
OTE (Output Energize)	Turn ON a bit (nonretentive)
OTL (Output Latch)	Latch a bit (retentive)
OTU (Output Unlatch)	Unlatch a bit (retentive)
TOF (Timer Off-Delay)	Turn an output ON or OFF after its rung has been OFF for a preset time interval
TON (Timer On-Delay)	Turn an output ON or OFF after its rung has been ON for a preset time interval
CTD (Count Down)	Use a software counter to count down from a specified value
CTU (Count Up)	Use a software counter to count up to a specified value

CHAPTER 1 REVIEW QUESTIONS

1. What is a programmable logic controller (PLC)?
2. Identify four tasks in addition to relay switching operations that PLCs are capable of performing.
3. List six distinct advantages that PLCs offer over conventional relay-based control systems.
4. Explain the differences between open and proprietary PLC architecture.
5. State two ways in which I/O is incorporated into the PLC.
6. Describe how the I/O modules connect to the processor in a modular-type PLC configuration.
7. Explain the main function of each of the following major components of a PLC:
   a. Processor module (CPU)
   b. I/O modules
   c. Programming device
   d. Power supply module
8. What are the two most common types of PLC programming devices?
9. Explain the terms *program* and *programming language* as they apply to a PLC.
10. What is the standard programming language used with PLCs?
11. Answer the following with reference to the process control relay ladder diagram of Figure 1-13 of this chapter:
    a. When do the pressure switch contacts close?
    b. When do the temperature switch contacts close?
    c. How are the pressure and temperature switches connected with respect to each other?
    d. Describe the two conditions under which the motor starter coil will become energized.
    e. What is the approximate value of the voltage drop across each of the following when their contacts are open?
    (1) Pressure switch
    (2) Temperature switch
    (3) Manual pushbutton
12. The programmable controller operates in real time. What does this mean?
13. Answer the following with reference to the process control PLC ladder logic diagram of Figure 1-16 of this chapter:
    a. What do the individual symbols represent?
    b. What do the numbers represent?
    c. What field device is the number I/2 identified with?
    d. What field device is the number O/1 identified with?
    e. What two conditions will provide a continuous path from left to right across the rung?
    f. Describe the sequence of operation of the controller for one scan of the program.
14. Compare the method by which the process control operation is changed in a relay-based system to the method used for a PLC-based system.
15. Compare the PLC and PC with regard to:
    a. Physical hardware differences
    b. Operating environment
    c. Method of programming
    d. Execution of program
16. What two categories of software written and run on PCs are used in conjunction with PLCs?
17. What is a programmable automation controller (PAC)?
18. List four criteria by which PLCs are categorized.
19. Compare the single-ended, multitask, and control management types of PLC applications.
20. What is the memory capacity, expressed in bits, for a PLC that uses 16-bit words and has an 8 K word capacity?
21. List five factors affecting the memory size needed for a particular PLC installation.
22. What does the instruction set for a particular PLC refer to?

CHAPTER 1 PROBLEMS

1. Given two single-pole switches, write a program that will turn on an output when both switch *A* and switch *B* are closed.
2. Given two single-pole switches, write a program that will turn on an output when either switch *A* or switch *B* is closed.
3. Given four NO (Normally Open) pushbuttons (*A*-*B*-*C*-*D*), write a program that will turn a lamp on if pushbuttons *A* and *B* or *C* and *D* are closed.
4. Write a program for the relay ladder diagram shown in Figure 1-24.

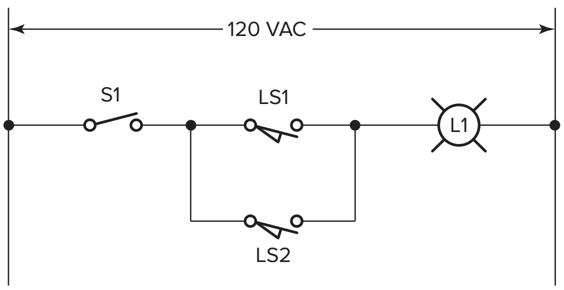


Figure 1-24 Circuit for Problem 4.

5. Write a program for the relay ladder diagram shown in Figure 1-25.

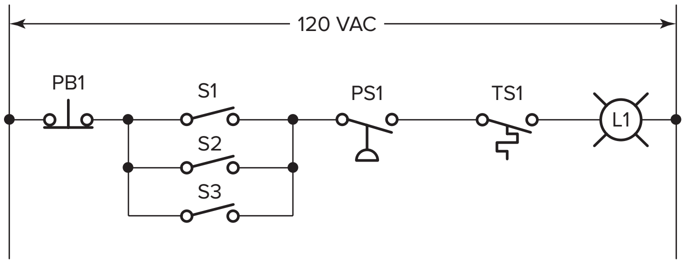


Figure 1-25 Circuit for Problem 5.

2

PLC Hardware Components

genkur/Shutterstock

This chapter exposes you to the details of PLC hardware and modules that make up a PLC control system. The chapter's illustrations show the various parts of a PLC as well as general connection paths. In this chapter we discuss the CPU and memory hardware components, including the various types of memory that are available, and we describe the hardware of the input/output section, including the difference between the discrete and analog types of modules.

Chapter Objectives

After completing this chapter, you will be able to:

- List and describe the function of the hardware components used in PLC systems
- Describe the basic circuitry and applications for discrete and analog I/O modules, and interpret typical I/O and CPU specifications
- Differentiate between tag-based and rack/slot-based addressing.
- Describe the general classes and types of PLC memory devices
- List and describe the different types of PLC peripheral support devices available

2.1 The I/O Section

The **input/output (I/O)** section of a PLC is the section to which all field devices are connected and provides the interface between them and the CPU. Input/output arrangements are built into a fixed PLC while modular types use external I/O modules that plug into the PLC.

Figure 2-1 illustrates a rack-based I/O section made up of individual I/O modules. Input interface modules accept signals from the machine or process devices and convert them into signals that can be used by the controller. Output interface modules convert controller signals into external signals used to control the machine or process. A typical PLC has room for several I/O modules, allowing it to be customized for a particular application by selecting the appropriate modules. Each slot in the rack is capable of accommodating any type of I/O module.

The I/O system provides an interface between the hardwired components in the field and the CPU. The input interface allows *status information* regarding processes to be communicated to the CPU, and thus allows the CPU to communicate *operating signals* through the output interface to the process devices under its control.

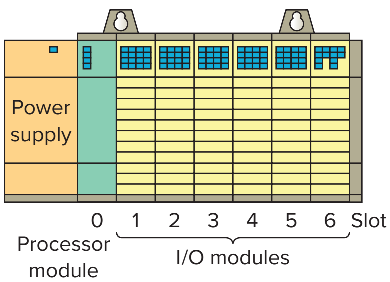


Figure 2-1 Rack-based I/O section.

One benefit of a PLC system is the ability to locate the I/O modules near the field devices. **Distributed I/O** refers to inputs and outputs located at a machine and controlled remotely from the PLC, as illustrated in Figure 2-2, in order to minimize the amount of wiring required. The processor receives signals from the remote input modules and sends signals back to their output modules via the communication module.

A rack is referred to as a ***remote*** rack when it is located away from the processor module. To communicate with the processor, the remote rack uses a special communications network. Each remote rack requires a unique station number to distinguish one from another. The remote racks are linked to the local rack through a ***communications module.*** Cables connect the modules with each other. If fiber optic cable is used between the CPU and I/O rack, it is possible to operate I/O points from distances greater than 20 miles with no voltage drop. Coaxial cable will allow remote I/O to be installed at distances greater than two miles. Fiber optic cable will not pick up noise caused by adjacent high power lines or equipment normally found in an industrial environment. Coaxial cable is more susceptible to this type of noise.

The PLC's memory system stores information about the status of all the inputs and outputs. To keep track of all this information, it uses a system called *addressing*. An **address** is a label or number that indicates where a certain piece of information is located in a PLC's memory. Just as your home address tells where you live in your city, a device's or a piece of data's address tells where

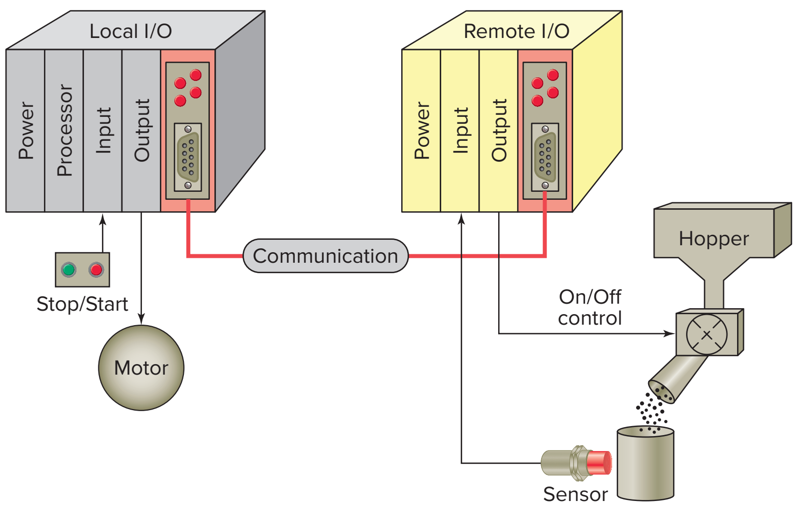


Figure 2-2 Remote I/O rack.

information about it resides in the PLC's memory. That way, if a PLC wants to find out information about a field device, it knows to look in its corresponding address location. Examples of addressing schemes include ***rack/slot-based***, versions of which are used in Allen-Bradley SLC 500 controllers, ***tag-based*** used in Allen-Bradley ControlLogix and CompactLogix controllers.

In general, rack/slot-based addressing elements include:

Type—The type determines if an input or output is being addressed.

Slot—The slot number is the physical location of the I/O module. This may be a combination of the rack number and the slot number when using expansion racks.

Word and Slot—The word and slot are used to identify the actual terminal connection in a particular I/O module. A discrete module usually uses only one word, and each connection corresponds to a different bit that makes up the word.

With a rack/slot address system the location of a module within a rack and the terminal number of a module to which an input or output device is connected will determine the device's address.

O:4/15	Output module in slot 4, terminal 15
I:3/8	Input module in slot 3, terminal 8
O:6.0	Output module, slot 6
I:5.0	Input module, slot 5

Every input and output device connected to a discrete I/O module is addressed to a specific ***bit*** in the PLC's memory. A bit is a binary digit that can be either 1 or 0. Analog I/O modules use a ***word*** addressing format, which allows the entire words to be addressed. The bit part of the address is usually not used; however, bits of the digital representation of the analog value can be addressed by the programmer if necessary. Figure 2-3 illustrates bit level and word level addressing.

The **RSLogix 5000** family of PLCs uses a tag-based memory structure with all data being assigned an alphanumeric name called a **tag.** Figure 2-4 illustrates the tag-based addressing format. Memory locations are defined by using base and alias tags. A **base tag** defines a memory location where data are stored. An **alias tag** is used to create an alternate name (alias) for a tag. The alias tag is often used to create a tag name to represent a real world input or output.

Figure 2-5 shows a comparison between rack/slot-based addressing and tag-based addressing. Input and output modules, when configured, automatically create their own tags like Local:1:I.Data.1. Tag names are descriptive to the data being stored in them. The alias tag lets you use names that are more meaningful for the application. In this example:

- Pressure_switch is used instead of I:1/1
- Temperature_switch is used instead of I:1/2
- Manual_pushbutton is used instead of I:1/3
- Mixer_motor is used instead of O:2/1

An I/O module is made up of a printed circuit board and a terminal assembly. The printed circuit board contains the electronic circuitry used to interface the circuit of the processor with that of the input or output device. Modules are designed to plug into a slot or connector in the I/O rack or directly into the processor. The terminal assembly, which is attached to the front edge of the printed circuit board, is used for making field-wiring connections. Modules contain terminals for each input and output connection, status lights for each of the inputs and outputs, and connections to the power supply used to power the inputs and outputs. Terminal and status light arrangements vary with different manufacturers.

Most PLC modules have plug-in wiring terminal strips. The terminal block is plugged into the actual module as illustrated in Figure 2-6. If there is a problem with a module, the entire strip is removed, a new module is inserted, and the terminal strip is plugged into the new module. Unless otherwise specified, never install or remove I/O modules or terminal blocks while the PLC is powered. A module inserted into the wrong slot could be damaged by improper voltages connected through the wiring arm. Most faceplates and I/O modules are keyed to prevent putting the wrong faceplate on the wrong module. In other words, an output module cannot be placed in the slot where an input module was originally located.

Input and output modules can be placed anywhere in a rack, but they are normally grouped together for ease of wiring. I/O modules can be 8, 16, 32, or 64 point cards. The number refers to the number of inputs or outputs available. The standard I/O module has eight inputs or outputs. A ***high-density*** module may have up to 64 inputs or outputs. The advantage with the high-density module is that it is possible to install up to 64 inputs or outputs in one slot for greater space savings. The only disadvantage is that the high-density output modules cannot handle as much current per output.

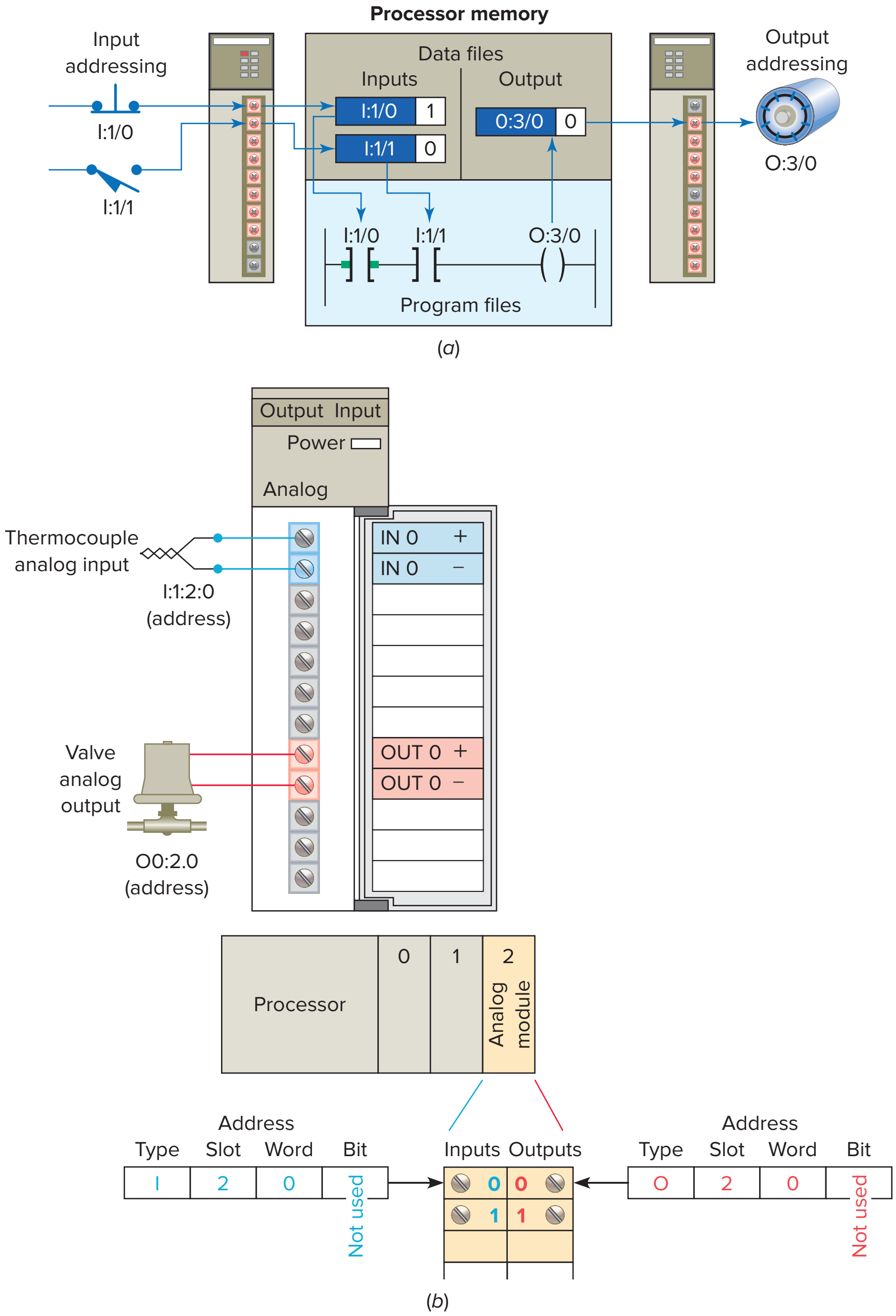
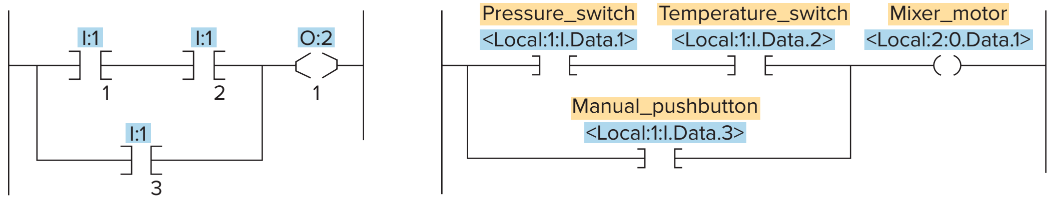


Figure 2-3 RS Logix 500 bit level and word level addressing. (*a*) Bit level addressing. (*b*) Word level addressing.

Figure 2-4 Allen-Bradley RSLogix 5000 tag-based addressing format.
Source: Image Courtesy of Rockwell Automation, Inc.



(*a*) RS Logix500 rack/slot-based addressing (*b*) Equivalent R 5000 tag-based addressing

Figure 2-5 Rack/slot-based versus tag-based addressing.

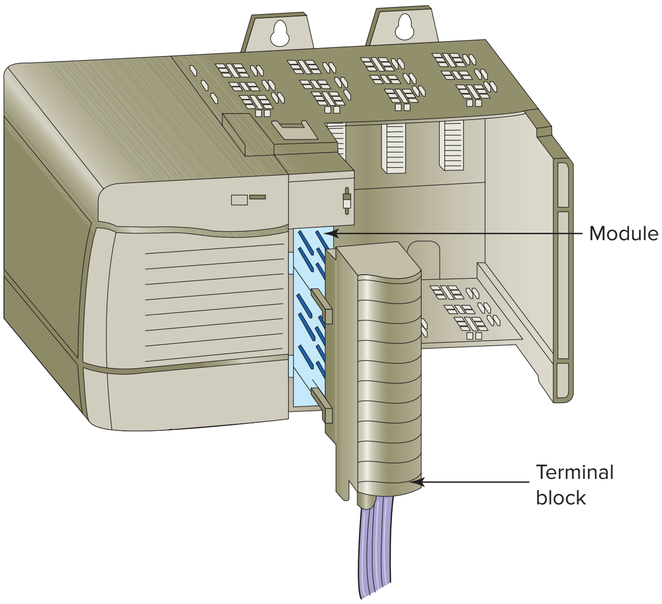


Figure 2-6 Plug-in terminal block.

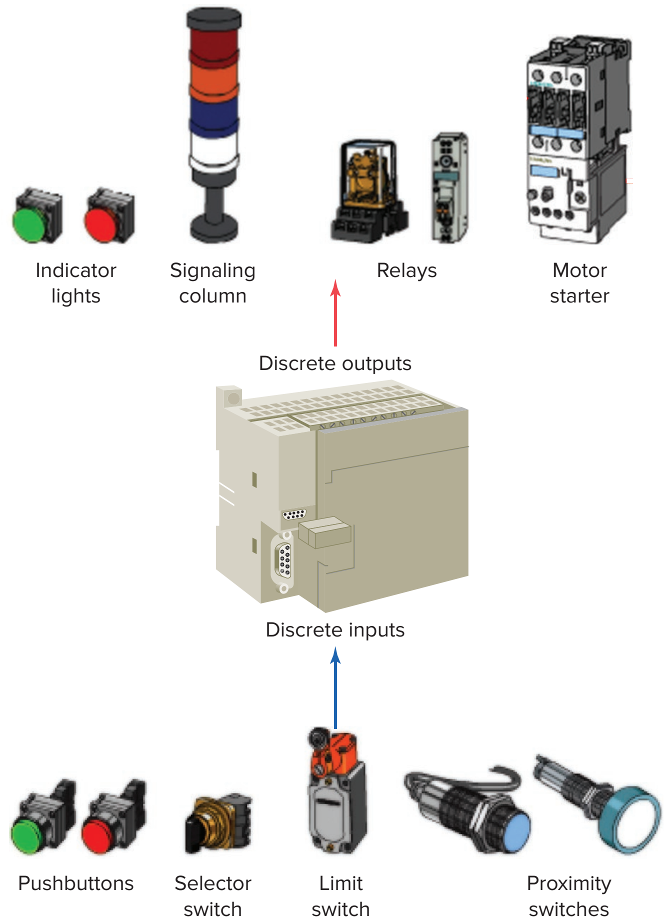


Figure 2-7 Discrete input and output devices.

2.2 Discrete I/O Modules

Discrete or **digital** type **input** modules (Figure 2-7) are designed to **monitor** ON/OFF devices such as selector switches, pushbuttons, and limit switches. Likewise, discrete output modules control devices such as lights, relays, solenoids, and motor starters that require simple ON/OFF switching. The classification of discrete I/O covers ***bit-oriented*** inputs and outputs. In this type of input or output, each bit represents a complete information element in itself and provides the status of some external contact or advises of the presence or absence of power in a process circuit.

Each discrete I/O module is powered by some *field-supplied* voltage source. Since these voltages can be of different magnitude or type, I/O modules are available at various AC and DC voltage ratings, as listed in Table 2-1.

The modules themselves receive their voltage and current for proper operation from the backplane of the rack enclosure into which they are inserted, as illustrated in Figure 2-8. **Backplane** power is provided by the PLC module power supply and is used to power the electronics that reside on the I/O module circuit board. The relatively higher

Table 2-1 Common Ratings for Discrete I/O Interface Modules

Input Interfaces	Output Interfaces
12 V **AC/DC** /24 V **AC/DC**	12–48 V **AC**
48 V **AC/DC**	120 V **AC**
120 V **AC/DC**	230 V **AC**
230 V **AC/DC**	120 V **DC**
5 V **DC** (TTL level)	230 V **DC**
	5 V **DC** (TTL level)
	24 V **DC**

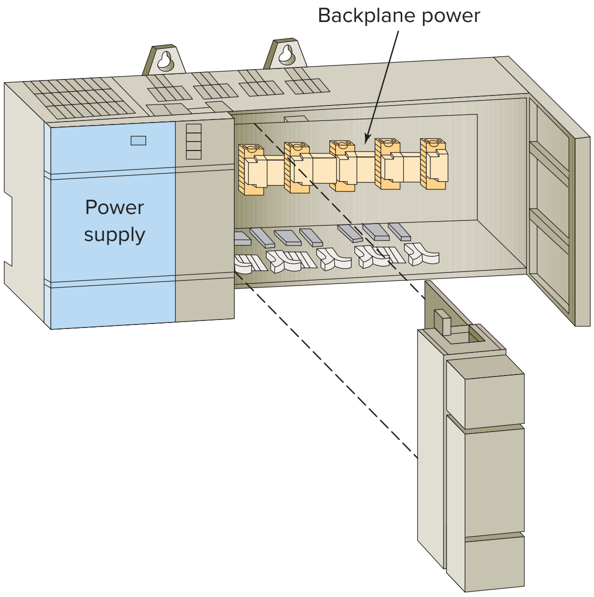


Figure 2-8 Modules receive their voltage and current from the backplane.

currents required by the loads of an output module are normally provided by user-supplied power. Module power supplies typically may be rated for 3 A, 4 A, 12 A, or 16 A depending on the type and number of modules used.

Figure 2-9 shows the block diagrams for one input of a typical alternating current (AC) *discrete input module*. The input circuit is composed of two basic sections: the ***power*** section and the ***logic*** section. An optical isolator is used to provide electrical isolation between the field wiring and the PLC backplane internal circuitry. The input LED turns on or off, indicating the status of the input device. Logic circuits process the digital signal to the processor. Internal PLC control circuitry typically operates at 5 VDC or less volts.

A simplified diagram for a single input of a discrete AC input module is shown in Figure 2-10. The operation of the circuit can be summarized as follows:

- The input noise filter consisting of the capacitor and resistors R1 and R2 removes false signals that are due to contact bounce or electrical interference.
- When the pushbutton is closed, 120 VAC is applied to the bridge rectifier input.
- This results in a low-level DC output voltage that is applied across the LED of the optical isolator.
- The Zener diode (Z_D) voltage rating sets the minimum threshold level of voltage that can be detected.
- When light from the LED strikes the phototransistor, it switches into conduction and the status of the pushbutton is communicated in logic to the processor.

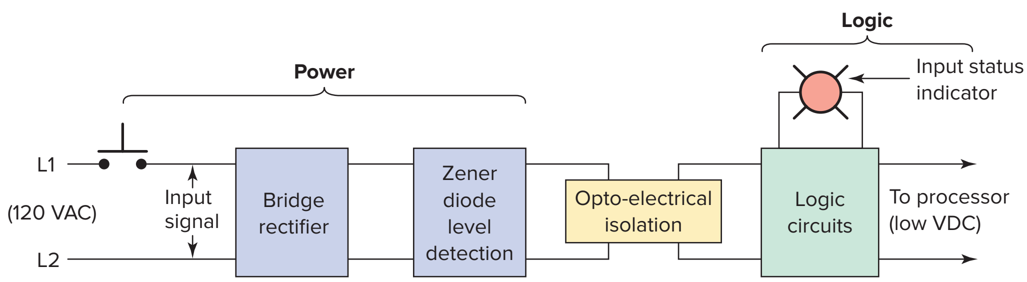


Figure 2-9 Discrete AC input module block diagram.

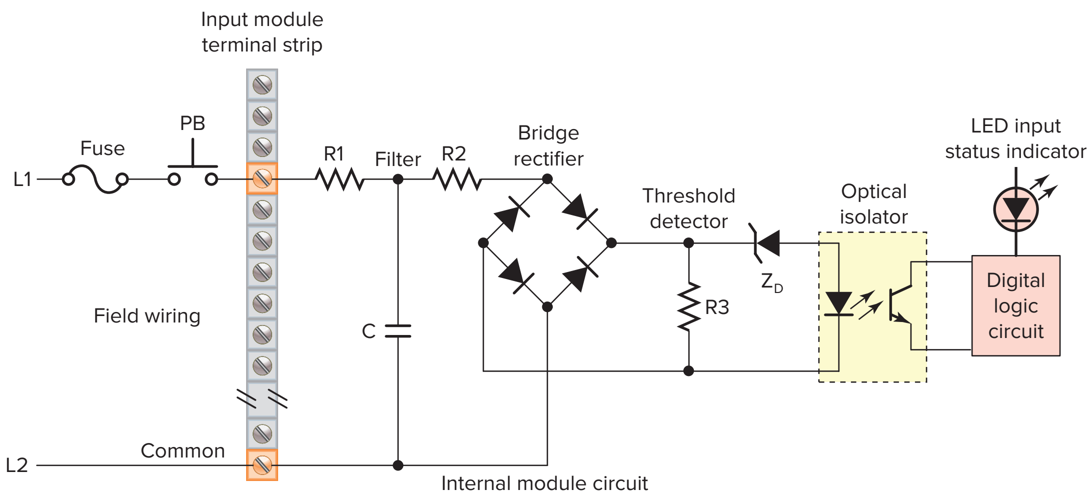


Figure 2-10 Simplified diagram for a single input of a discrete AC input module.

- The optical isolator not only separates the higher AC input voltage from the logic circuits but also prevents damage to the processor due to line voltage transients. In addition, this isolation also helps reduce the effects of electrical noise, common in the industrial environment, which can cause erratic operation of the processor.
- For fault diagnosis, an input state LED indicator is on when the input pushbutton is closed. This indicator may be wired on either side of the optical isolator.
- An AC/DC type of input module is used for both AC and DC inputs as the input polarity does not matter.
- A PLC input module will have either all inputs isolated from each other with no common input connections or groups of inputs that share a common connection.

Discrete input modules perform four tasks in the PLC control system. They:

- Sense when a signal is received from a field device.
- Convert the input signal to the correct voltage level for the particular PLC.
- Isolate the PLC from fluctuations in the input signal's voltage or current.
- Send a signal to the processor indicating which sensor originated the signal.

Figure 2-11 shows the block diagram for one output of a typical discrete output module. Like the input module, it is composed of two basic sections: the power section and the logic section, coupled by an isolation circuit. The output interface can be thought of as an electronic switch that turns the output load device on and off. Logic circuits determine the output status. An output LED indicates the status of the output signal.

A simplified diagram for a single output of a discrete AC output module is shown in Figure 2-12. The operation of the circuit can be summarized as follows:

- As part of its normal operation, the digital logic circuit of the processor sets the output status according to the program.
- When the processor calls for an output load to be energized, a voltage is applied across the LED of the opto-isolator.

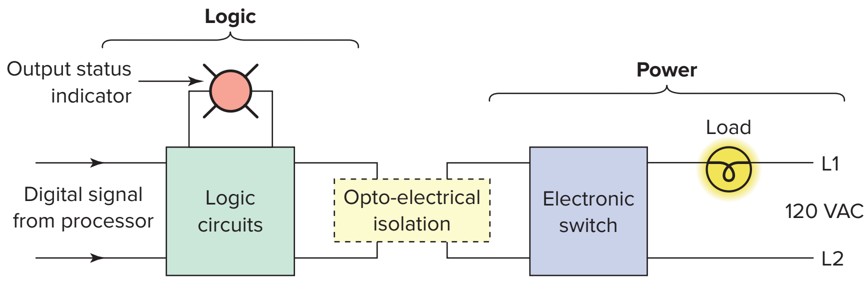


Figure 2-11 Discrete AC output module block diagram.

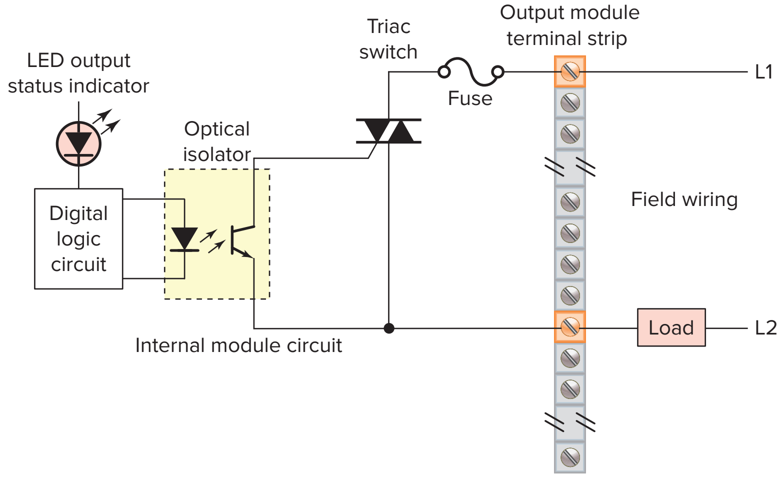


Figure 2-12 Simplified diagram for a single output of a discrete AC output module.

- The LED then emits light, which switches the phototransistor into conduction.
- This in turn triggers the triac AC semiconductor switch into conduction, allowing current to flow to the output load.
- Since the triac conducts in either direction, the output to the load is alternating current.
- The triac, rather than having ON and OFF status, actually has LOW and HIGH resistance levels, respectively. In its OFF state (HIGH resistance), a small leakage current of a few milliamperes still flows through the triac.
- As with input circuits, the output interface is usually provided with LEDs that indicate the status of each output.
- Fuses are normally required for the output module, and they are provided on a per circuit basis, thus allowing for each circuit to be protected and operated separately. Some modules also provide visual indicators for fuse status.
- The triac cannot be used to switch a DC load.
- For fault diagnosis, the LED output status indicator is on whenever the PLC is commanding that the output load be switched on.

Individual AC outputs are usually limited by the size of the triac to 1 A or 2 A. The maximum current load for any one module is also specified. To protect the output module circuits, specified current ratings should not be exceeded. For controlling larger loads, such as large motors, a standard control relay is connected to the output module. The contacts of the relay can then be used to control a larger load or motor starter, as shown in Figure 2-13. When a control relay is used in this manner, it is called an ***interposing* relay.**

Discrete output modules are used to turn field output devices either on or off. These modules can be used to control any two-state device, and they are available in AC and DC versions and in various voltage ranges and current ratings. Output modules can be purchased with *transistor, triac,* or

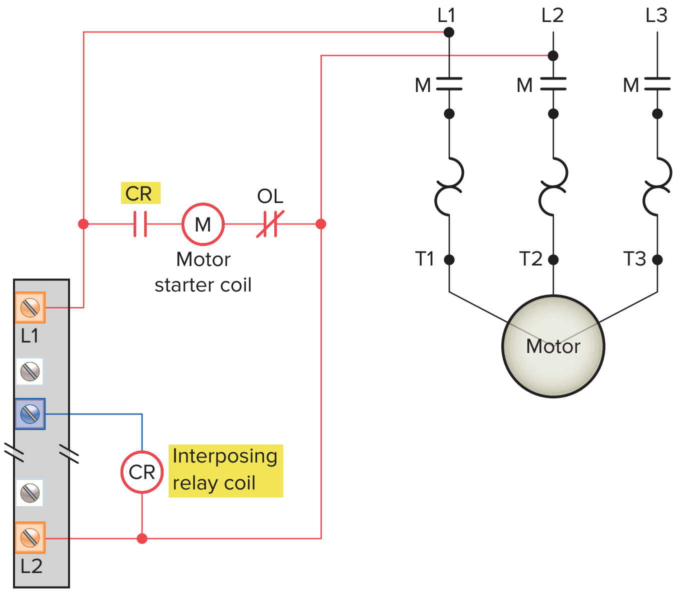


Figure 2-13 Interposing relay connection.

relay output as illustrated in Figure 2-14. Triac outputs can be used only for control of AC devices, whereas transistor outputs can be used only for control of DC devices. The discrete relay contact output module uses electromechanical as the switching element. These relay outputs can be used with AC or DC devices, but they have a much slower switching time compared to solid-state outputs.

Certain DC I/O modules specify whether the module is designed for interfacing with current-source (PNP) or current-sink (NPN) devices. If the module is a current-sourcing module, then the input or output device must be a current-sinking device. Conversely, if the module is specified as current-sinking, then the connected device must be current-sourcing. Some modules allow the user to select whether the module will act as current sinking or current sourcing, thereby allowing it to be set to whatever the field devices require. Sinking and sourcing terminology applies only to DC input and output circuits.

Allen-Bradley delineates between the various digital DC modules by sorting them into two categories:

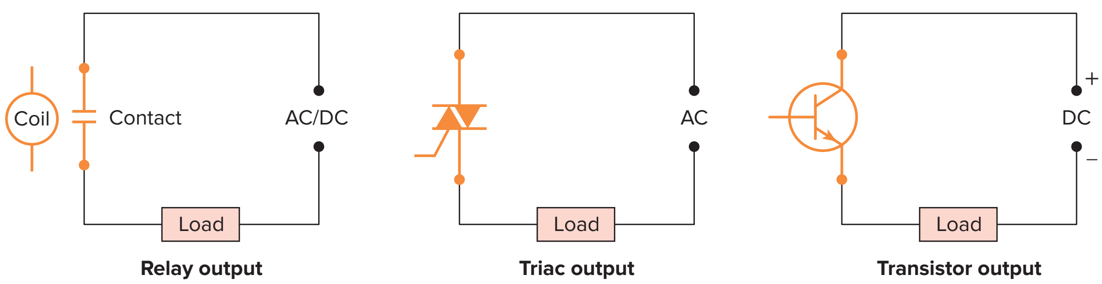


Figure 2-14 Relay, transistor, and triac switching elements.

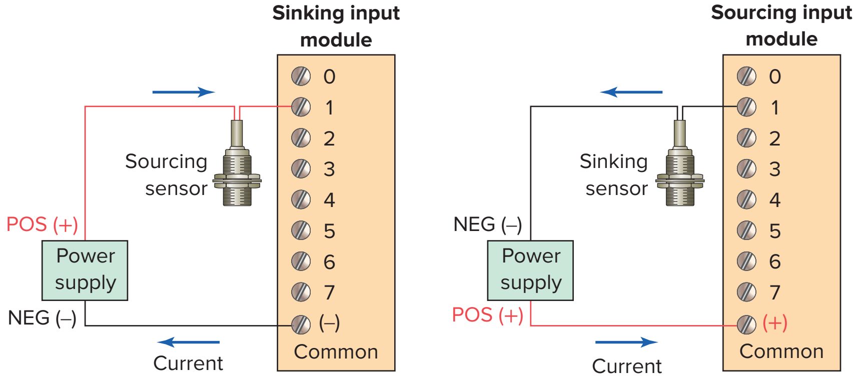


Figure 2-15 Sinking and sourcing inputs.

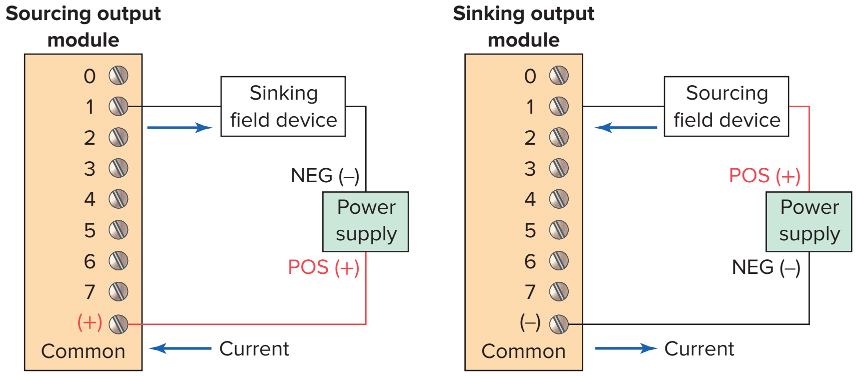


Figure 2-16 Sinking and sourcing outputs.

Sinking and **Sourcing.** These terms are used to describe a current signal flow relationship between field input and output devices. If a device provides current when it is ON, it is said to be sourcing current. Conversely, if a device receives current when it is ON, it is said to be sinking current.

Figures 2-15 and 2-16 show device connections for both sourcing and sinking configurations:

- Conventional current (+ to −) is assumed.
- In sinking devices, current flows into the device's terminal from the module (the module provides, or sources the current).
- In sourcing devices, current flows out of the device's terminal into the module (the module receives, or sinks, the current).
- A sourcing I/O device or I/O module will always have a connection directly to the positive side of the DC power supply.
- A sinking I/O device or I/O module will always have a connection directly to the negative side of the DC power supply.
- Input and output points that are sinking or sourcing only can conduct current in only one direction.

Therefore, it is possible to connect the external supply and field device to the I/O point with current trying to flow in the wrong direction, and the circuit **will not operate.**

2.3 Analog I/O Modules

Analog input and output modules are used whenever the control process requires the **continuously variable** type of control, in contrast to the discrete or digital ON/OFF types. Typical analog input detection devices include temperature sensors, potentiometers, and ultrasonic proximity sensors. Typical analog output control devices include control valves, meters, and stepper motors (Figure 2-17).

Discrete devices are inputs and outputs that have only two states: on and off. In comparison, **analog** devices represent physical quantities that can have an infinite number of values. Typical analog inputs and outputs vary from 0 to 20 mA, 4 to 20 mA, or 0 to 10 V. Figure 2-18 illustrates how PLC analog input and output modules are used in measuring and displaying the level of fluid in a tank. The analog input interface module contains the circuitry necessary to accept an analog voltage or current signal from the level transmitter field device. This input is converted from

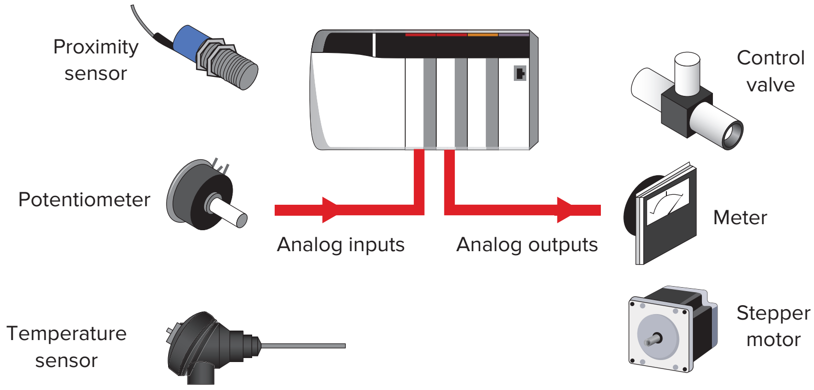


Figure 2-17 Analog input and output devices.

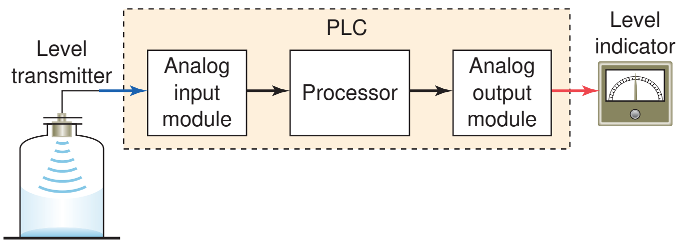


Figure 2-18 Analog input and output to a PLC.

an analog to a digital value for use by the processor. The circuitry of the analog output module accepts the digital value from the processor and converts it back to an analog signal that drives the field tank level meter.

Analog input modules normally have multiple input channels that allow 4, 8, or 16 devices to be interface to the PLC. The two basic types of analog input modules are ***voltage* sensing** and ***current* sensing.** Analog sensors measure a varying physical quantity over a specific range and generate a corresponding voltage or current signal. Common physical quantities measured by a PLC analog module include temperature, speed, level, flow, weight, pressure, and position. For example, a sensor may measure temperature over a range of 0 to 500°C, and output a corresponding voltage signal that varies between 0 and 50 mV.

Figure 2-19 illustrates an example of a voltage sensing input analog module used to measure temperature. The connection diagram applies to an Allen-Bradley MicroLogic 4-channel analog thermocouple input module. A varying DC voltage in the low millivolt range, proportional to the temperature being monitored, is produced by the thermocouple. This voltage is amplified and digitized by the analog input module and then sent to the processor on command from a program instruction. Because of the low voltage level of the input signal, a twisted shielded pair cable is used in wiring the circuit to reduce unwanted electrical noise signals that can be induced in the conductors from other wiring. When using an ungrounded thermocouple, the shield must be connected to ground at the module end. To obtain accurate readings from each of the channels, the temperature between the thermocouple wire and the input channel must be compensated for. A cold junction compensating (CJC) thermistor is integrated in the terminal block for this purpose.

Only digital signals can be handled by PLCs. The transition of an analog signal to digital values is accomplished by an

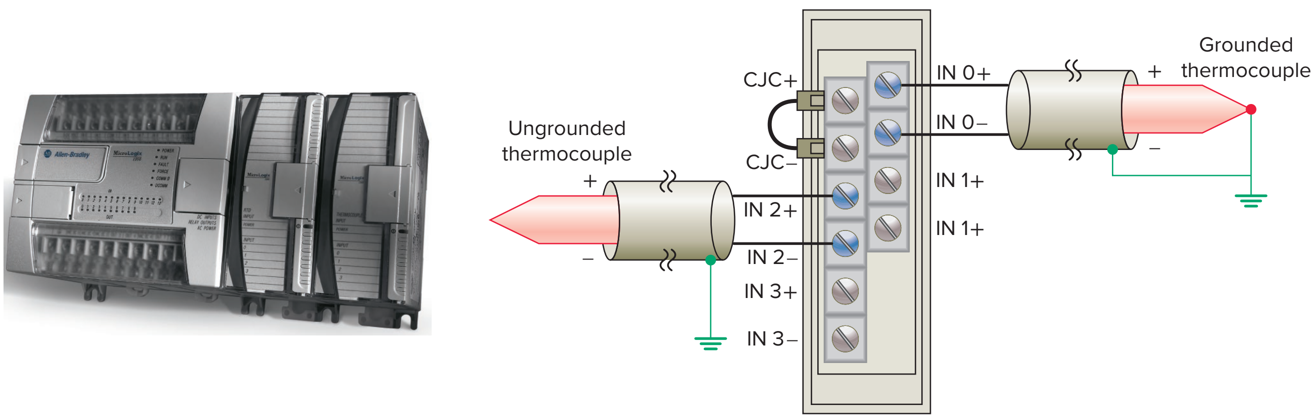


Figure 2-19 MicroLogix 4-channel analog thermocouple input module.
Source: Image Courtesy of Rockwell Automation, Inc.

analog-to-digital (A/D) converter, the main element of the analog input module. Analog voltage input modules are available in two types: unipolar and bipolar. ***Unipolar*** modules can accept an input signal that varies in the positive direction only. For example, if the field device outputs 0 to +10 V, then the unipolar modules would be used. Bipolar signals swing between a maximum negative value and a maximum positive value. For example, if the field device outputs −10 to +10 V a bipolar module would be used. The ***resolution*** of an analog input channel refers to the smallest change in input signal value that can be sensed and is based on the number of bits used in the digital representation. Analog input modules must produce a range of digital values between a maximum and minimum value to represent the analog signal over its entire span. Typical specifications are as follows:

Span of analog input	Bipolar	10 V	–10 to +10 V
		5 V	–5 to +5 V
	Unipolar	10 V	0 to +10 V
		5 V	0 to +5 V
Resolution			0.3 mV

When connecting voltage sensing inputs, close adherence to specified requirements regarding wire length is important to minimize signal degrading and the effects of electromagnetic noise interference induced along the connecting conductors. Current input signals, which are not as sensitive to noise as voltage signals, are typically not distance limited. Current sensing input modules typically accept analog data over the range of 4 to 20 mA, but can accommodate signal ranges of –20 to +20 mA. The loop power may be supplied by the sensor or may be provided by the analog output module as illustrated in Figure 2-20. Shielded twisted pair cable is normally recommended for connecting any type of analog input signal.

Field devices that provide an analog output as their signal may be connected to transmitters, or may contain their own internal transmitters to generate the analog signal to the module, as illustrated in Figure 2-21. A **transducer** converts a field device's variable (e.g., pressure, temperature etc.) into a very low-level electric signal (current or voltage) that can be amplified by a **transmitter** and then input into the analog module.

The method used to wire two-, three-, and four-wire sensors to an analog input module is illustrated in Figure 2-22. The module does not provide loop power for analog inputs. A separate power that matches the

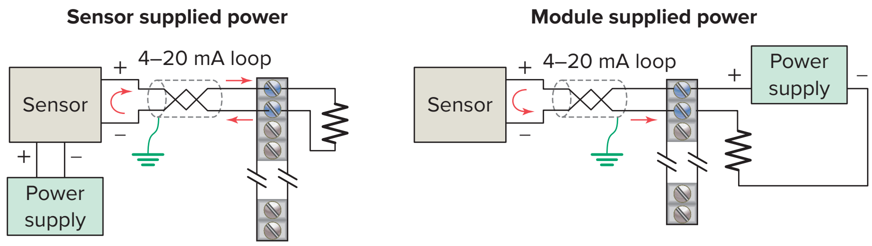


Figure 2-20 Sensor and analog module supplied power.

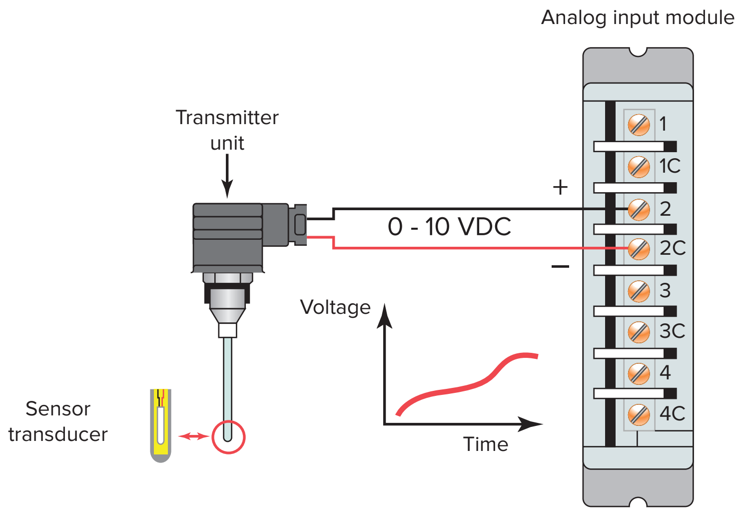


Figure 2-21 Transmitter input to an analog module.

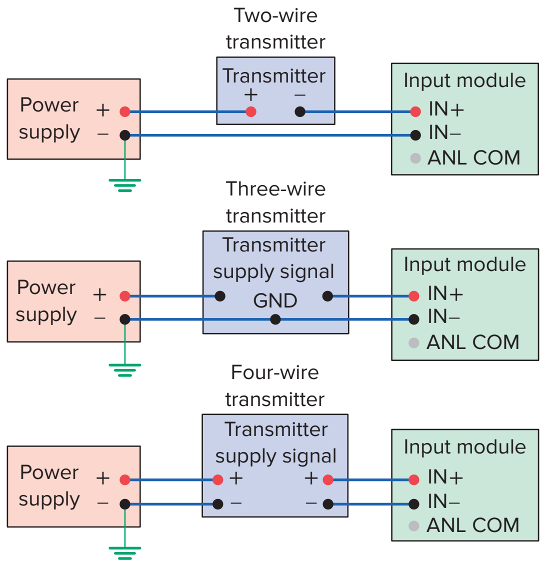


Figure 2-22 Wiring two-, three-, and four-wire sensors to an analog input module.

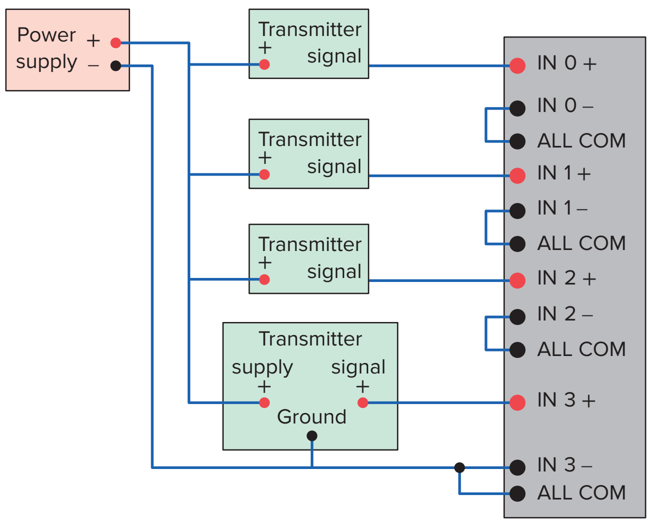


Figure 2-23 Wiring single-ended analog input devices.

transmitter specifications is used. All analog common (ANL COM) points are electrically connected together inside the module but not to earth ground. When wiring single-ended analog input devices to the analog input card, the number of total wires necessary can be limited by using the ANALOG COMMON terminal, as shown in Figure 2-23. Note that differential inputs are more immune to noise than single-ended inputs.

The ***analog output interface module*** receives from the processor digital data, which are converted into a proportional voltage or current to control an analog field device. The transition of a digital signal to analog values is

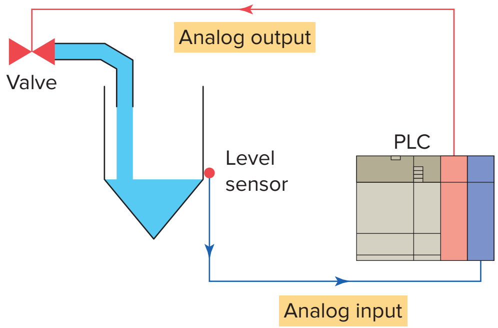


Figure 2-24 Typical analog I/O control system.

accomplished by a **digital-to-analog (D/A)** converter, the main element of the analog output module. An analog output signal is a continuous and changing signal that is varied under the control of the PLC program. Common devices controlled by a PLC analog output module include instruments, control valves, chart recorder, electronic drives, and other types of control devices that respond to analog signals. They employ standard analog output ranges such as ±5 V, ±10 V, 0 to 5 V, 0 to 10 V, 4 to 20 mA, or 0 to 20 mA.

Figure 2-24 illustrates the use of analog I/O modules in a typical PLC control system. In this application the PLC controls the amount of fluid placed in a holding tank. The level sensor detects the fluid level in the tank and transmits a signal to the analog input. The analog output from the PLC is used to control the flow by adjusting the percentage of the valve opening. The valve is initially open 100%. As the fluid level in the tank approaches the preset point, the processor modifies the output, which adjusts the valve to maintain a set point.

Transducers produce either voltage or current proportional to some engineering units such as temperature (°C or °F), pressure (lb/in^2), distance (cm), etc. **Scaling** refers to changing a quantity from one notation to another and involves:

Engineering units: The units a human uses and understands

Transducer units: Either a voltage or current

Binary, raw, or machine units: The units the processor requires

Scaling is used to produce an output value that has a linear relationship between the input and scaled values. It allows you to take an analog input from a sensor and convert it to the engineering units required by the application. Figure 2-25 illustrates a typical application involving temperature measurement.

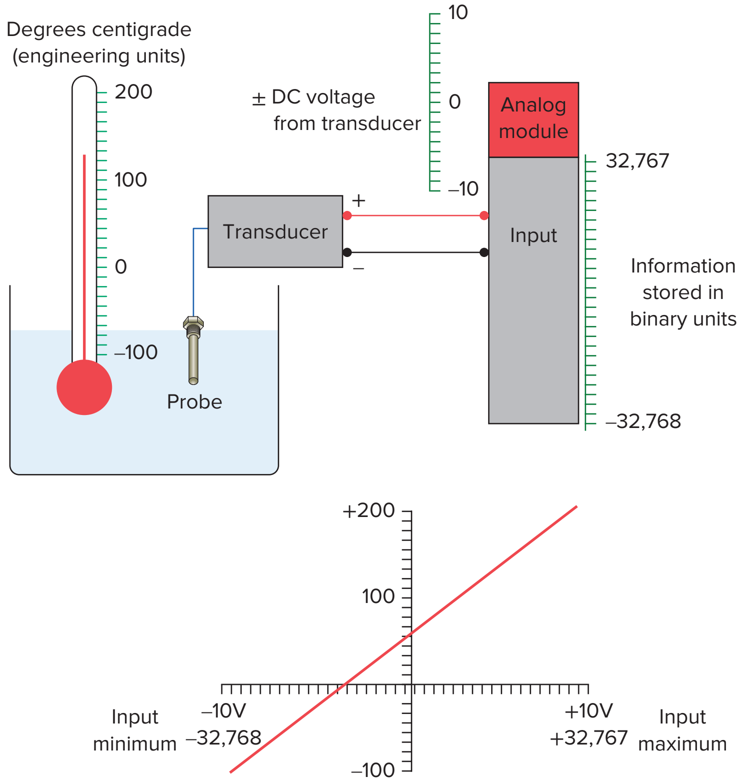


Figure 2-25 Measuring temperature.

2.4 Special I/O Modules

Many different types of I/O modules have been developed to meet special needs. These include:

HIGH-SPEED COUNTER MODULE

The high-speed counter module is used to provide an interface for applications requiring counter speeds that surpass the capability of the PLC ladder program. High-speed counter modules are used to count pulses (Figure 2-26) from sensors, encoders, and switches that operate at very high speeds. They have the electronics needed to count independently of the processor. A typical count rate available is 0 to 100 kHz, which means the module would be able to count 100,000 pulses per second.

ENCODER-COUNTER MODULE

An encoder-counter module allows the user to read the signal from an encoder (Figure 2-27) on a real-time basis and stores this information so it can be read later by the processor. An encoder is mainly used for feedback

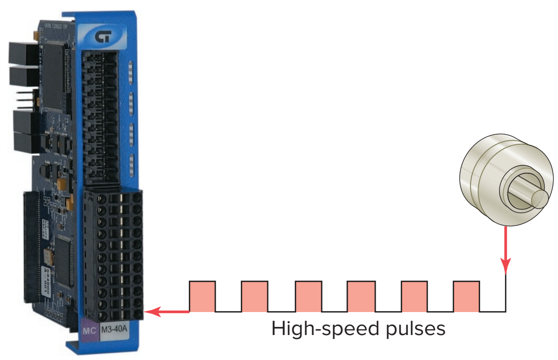


Figure 2-26 High-speed counter module.
Source: Courtesy of Control Technology Corporation

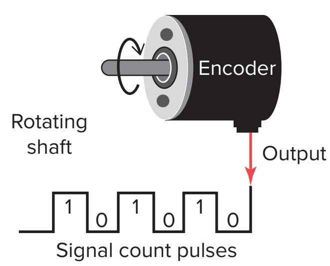


Figure 2-27 Rotary encoder.

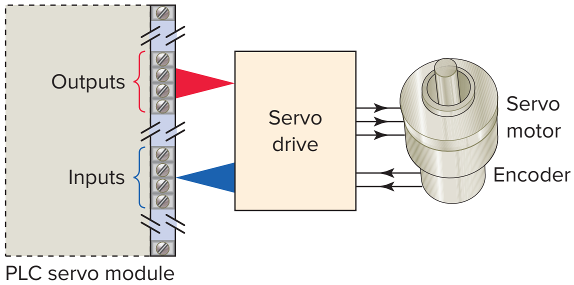


Figure 2-28 PLC servo module.

purpose to know the **position and distance** measurement in process applications. Encoders convert motion to electrical signal pulses. For example, if a rotary encoder is designed to produce 100 pulses per revolution, each pulse would indicate 3.6 degrees of rotation.

STEPPER-MOTOR MODULE

The stepper-motor module provides pulse trains to a stepper-motor translator, which enables control of a stepper motor. The commands for the module are determined by the control program in the PLC.

MOTION AND POSITION CONTROL MODULE

Motion and position control modules are used in applications involving accurate high-speed machining and packaging operations. Intelligent position and motion control modules permit PLCs to control stepper and servo motors. These systems require a drive, which contains the power electronics that translate the signals from the PLC module into signals required by the motor (Figure 2-28).

COMMUNICATION MODULES

Serial communications modules are used to establish point-to-point connections with other intelligent devices for the exchange of data. Such connections are normally established with computers, operator stations, process control systems, and other PLCs. Communication modules allow the user to connect the PLC to high-speed local networks that may be different from the network communication provided with the PLC.

2.5 I/O Specifications

Manufacturers' specifications provide information about how an interface device is correctly and safely used. These specifications place certain limitations not only on the I/O module but also on the field equipment that it can operate. Some PLC systems support *hot swappable* I/O modules designed to be changed with the power on and the PLC operating. The following is a list of some typical manufacturers' I/O specifications, along with a short description of what is specified.

Typical Discrete (Digital) I/O Module Specifications

NOMINAL INPUT VOLTAGE

This discrete input module voltage value specifies the magnitude (e.g., 5, 24, 230 V) and type (AC or DC) of user-supplied voltage that a module is designed to accept. Input modules are typically designed to operate correctly without damage within a range of plus or minus 10% of the input voltage rating. With DC input modules, the input voltage may also be expressed as an operating range (e.g., 24 to 60 V DC) over which the module will operate.

INPUT THRESHOLD VOLTAGES

This discrete input module specification specifies two values: a minimum ON-state voltage that is the minimum voltage at which logic 1 is recognized as absolutely ON; and a maximum OFF-state voltage which is the voltage at which logic 0 is recognized as absolutely OFF.

NOMINAL CURRENT PER INPUT

This value specifies the minimum input current that the discrete input devices must be capable of driving to operate the input circuit. This input current value, in conjunction with the input voltage, functions as a threshold to protect against detecting noise or leakage currents as valid signals.

AMBIENT TEMPERATURE RATING

This value specifies what the maximum temperature of the air surrounding the I/O modules should be for best operating conditions.

INPUT ON/OFF DELAY

Also known as *response time,* this value specifies the maximum time duration required by an input module's circuitry to recognize that a field device has switched ON (input ON-delay) or switched OFF (input OFF-delay). This delay is a result of filtering circuitry provided to protect against contact bounce and voltage transients. This input delay is typically in the 9 to 25 ms range.

OUTPUT VOLTAGE

This AC or DC value specifies the magnitude (e.g., 5 V, 115 V, 230 V) and type (AC or DC) of user-supplied voltage at which a discrete output module is designed to operate. The output field device that the module interfaces to the PLC must be matched to this specification. Output modules are typically designed to operate within a range of plus or minus 10% of the nominal output voltage rating.

OUTPUT CURRENT

These values specify the maximum current that a single output and the module as a whole can safely carry under load (at rated voltage). This rating is a function of the module's components and heat dissipation characteristics. A device drawing more than the rated output current results in overloading, causing the output fuse to blow. As an example, the specification may give each output a current limit of 1 A. The overall rating of the module current will normally be less than the total of the individuals. The overall rating might be 6 A because each of the eight devices would not normally draw their 1 A at the same time. Other names for the output current rating are *maximum continuous current* and *maximum load current.*

INRUSH CURRENT

An inrush current is a momentary surge of current that an AC or DC output circuit encounters when energizing inductive, capacitive, or filament loads. This value specifies the maximum inrush current and duration (e.g., 20 A for 0.1 s) for which an output circuit can exceed its maximum continuous current rating.

SHORT CIRCUIT PROTECTION

Short circuit protection is provided for AC and DC output modules by either fuses or some other current-limiting circuitry. This specification will designate whether the particular module's design has individual protection for each circuit or if fuse protection is provided for groups (e.g., 4 or 8) of outputs.

LEAKAGE CURRENT

This value specifies the amount of current still conducting through an output circuit even after the output has been turned off. Leakage current is a characteristic exhibited by solid-state switching devices such as transistors and triacs and is normally 1 to 2 mA. Leakage current is normally not large enough to falsely trigger an output device but must be taken into consideration when switching very low current sensitive devices.

ELECTRICAL ISOLATION

Recall that I/O module circuitry is electrically isolated to protect the low-level internal circuitry of the PLC from high voltages that can be encountered from field device connections. The specification for electrical isolation, typically 1500 or 2500 V AC, rates the module's capacity for sustaining an excessive voltage at its input or output terminals.

POINTS PER MODULE

This specification defines the number of field inputs or outputs that can be connected to a single module. Most commonly, a discrete module will have 8, 16, or 32 circuits. Modules with 32 or 64 input or output bits are referred to as *high-density* modules. Some modules provide more than one common terminal, which allows the user to use different voltage ranges on the same card as well as to distribute the current more effectively.

BACKPLANE CURRENT DRAW

This value indicates the amount of current the module requires from the backplane. The sum of the backplane current drawn for all modules in a chassis is used to select the appropriate chassis power supply rating.

Typical Analog I/O Module Specifications

CHANNELS PER MODULE

Whereas individual circuits on discrete I/O modules are referred to as points, circuits on analog I/O modules are referred to as channels. These modules normally have 4, 8, or 16 channels. Analog modules may allow for either single-ended or differential connections. *Single-ended* connections use a single ground terminal for all channels or for groups of channels. *Differential* connections use a separate positive and negative terminal for each channel. If the module normally allows 16 single-ended connections, it will generally allow only 8 differential connections. Single-ended connections are more susceptible to electrical noise.

INPUT CURRENT/VOLTAGE RANGE(S)

These are the voltage or current signal ranges that an analog input module is designed to accept. The input ranges must be matched accordingly to the varying current or voltage signals generated by the analog sensors.

OUTPUT CURRENT/VOLTAGE RANGE(S)

This specification defines the current or voltage signal ranges that a particular analog output module is designed to output under program control. The output ranges must be matched according to the varying voltage or current signals that will be required to drive the analog output devices.

INPUT PROTECTION

Analog input circuits are usually protected against accidentally connecting a voltage that exceeds the specified input voltage range.

RESOLUTION

The resolution of an analog I/O module specifies how accurately an analog value can be represented digitally. This specification determines the smallest measurable unit of current or voltage. The higher the resolution (typically specified in bits or mV), the more accurately an analog value can be represented.

INPUT IMPEDANCE AND CAPACITANCE

For analog I/Os, these values must be matched to the external device connected to the module. Typical ratings are in Megohm (MΩ) and picofarads (pF).

COMMON-MODE REJECTION

Noise is generally caused by electromagnetic interference, radio frequency interference, and ground loops. Common-mode noise rejection applies only to differential inputs and refers to an analog module's ability to prevent noise from interfering with data integrity on a single channel and from channel to channel on the module. Noise that is picked up equally in parallel wires is rejected because the difference is zero. Twisted pair wires are used to ensure that this type of noise is equal on both wires. Common-mode rejection is normally expressed in decibels or as a ratio.

2.6 The Central Processing Unit (CPU)

The central processing unit (CPU) is built into single-unit fixed PLCs while modular rack types typically use a plug-in module. CPU, controller, and processor are all terms used by different manufacturers to denote the same module that performs basically the same functions. Processors vary in processing speed and memory options. A processor module can be divided into two sections: the **CPU section** and the **memory section** (Figure 2-29). The CPU section executes the program and makes the decisions needed by the PLC to operate and communicate with other modules. The memory section electronically stores the PLC program along with other retrievable digital information.

The PLC power supply provides the necessary power (typically 5 VDC) to the processor and I/O modules plugged into the backplane of the rack (Figure 2-30).

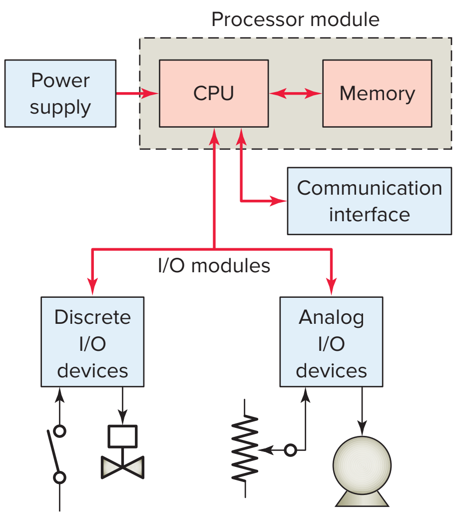


Figure 2-29 Sections of a PLC processor module.

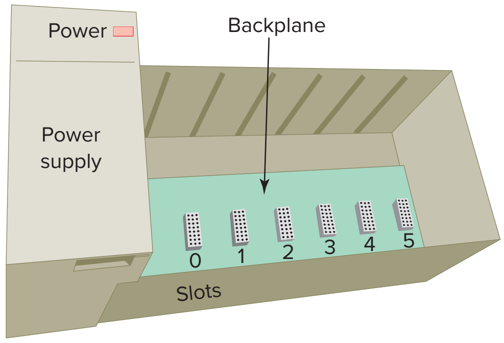


Figure 2-30 PLC power supply.

Power supplies are available for most voltage sources encountered. The power supply converts 24 VDC, 115 VAC or 230 VAC into the usable DC voltage required by the CPU, memory, and I/O electronic circuitry. PLC power supplies are normally designed to withstand momentary losses of power without affecting the operation of the PLC. *Hold-up time,* which is the length of time a PLC can tolerate a power loss, typically ranges from 10 ms to 3 s.

The CPU contains the similar type of microprocessor found in a personal computer. The difference is that the program used with the microprocessor is designed to facilitate industrial control rather than provide general-purpose computing. The CPU executes the operating system, manages memory, monitors inputs, evaluates the user logic (ladder program), and turns on the appropriate outputs.

Every PLC processor handles some type of communication. The **communications** done by the PLC CPU include:

- Communications to the programmer through a serial, Ethernet, or USB port on the processor module.
- Communications to the input and output (I/O) modules through the chassis backplane.
- Communications to other PLCs and other industrial automation devices through Ethernet and other network types.

The CPU of a PLC system may contain more than one processor. One advantage of using multiprocessing is that the overall operating speed is improved. Each processor has its own memory and programs, which operate simultaneously and independently. In such configurations the scan of each processor is parallel and independent thus reducing the total response time. Fault-tolerant PLC systems support dual processors for critical processes. These systems allow the user to configure the system with ***redundant*** (two) processors, which allows transfer of control to the second processor in the event of a processor fault.

Associated with the processor unit will be a number of status LED indicators to provide system diagnostic

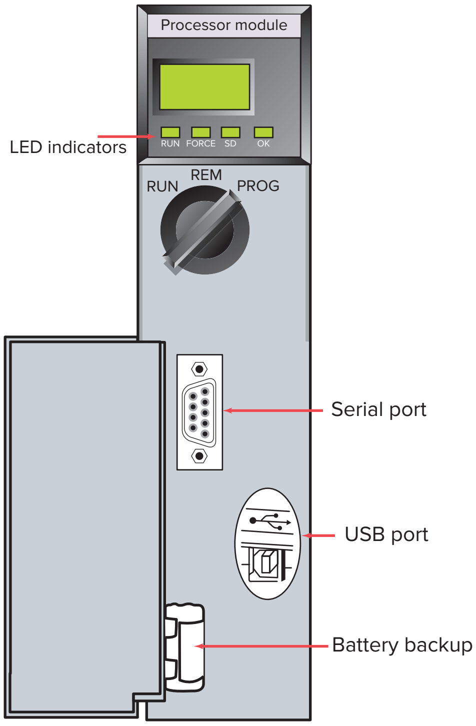


Figure 2-31 Typical processor module.
Source: Photo courtesy of Automation Direct, **www.automationdirect.com**

information to the operator (Figure 2-31). Also, a keyswitch may be provided that allows you to select one of the following three modes of operation: RUN, PROG, and REM.

RUN Position

- Places the processor in the Run mode
- Executes the ladder program and energizes output devices
- Prevents you from performing online program editing in this position
- Prevents you from using a programmer/operator interface device to change the processor mode

PROG Position

- Places the processor in the Program mode
- Prevents the processor from scanning or executing the ladder program, and the controller outputs are de-energized
- Allows you to perform program entry and editing
- Prevents you from using a programmer/operator interface device to change the processor mode

REM Position

- Places the processor in the Remote mode: either the REMote Run, REMote Program, or REMote Test mode
- Allows you to change the processor mode from a programmer/operator interface device
- Allows you to perform online program editing

The processor module also contains circuitry to communicate with the programming device. Somewhere on the module you will find a connector that allows the PLC to be connected to an external programming device. The decision-making capabilities of PLC processors go far beyond simple logic processing. The processor performs other functions such as timing, counting, latching, comparing, motion control and complex math functions.

PLC processors have changed constantly due to advancements in computer technology and greater demand from applications. Today, processors are faster and have additional instructions added as new models are introduced. Because PLCs are microprocessor based, they can be made to perform tasks that a computer can do. In addition to their control functions, PLCs can be networked to do supervisory control and data acquisition (SCADA).

Many electronic components found in processors and other types of PLC modules are sensitive to *electrostatic* voltages that can degrade their performance or damage them. The following static control procedures should be followed when handling and working with static-sensitive devices and modules:

- Ground yourself by touching a conductive surface before handling static-sensitive components.
- Wear a wrist strap that provides a path to bleed off any charge that may build up during work.
- Be careful not to touch the backplane connector or connector pins of the PLC system (always handle the circuit cards by the edge if possible).
- Be careful not to touch other circuit components in a module when you configure or replace its internal components.
- When not in use, store module in its static-shield bag.
- If available, use a static-safe work station.

2.7 Memory Design

Memory is the element that stores information, programs, and data in a PLC. The user memory of a PLC includes space for the user program as well as addressable memory locations for storage of data. Data are stored in memory locations by a process called *writing*. Data are retrieved from memory by what is referred to as *reading*.

The complexity of the program determines the amount of memory required. Memory elements store individual pieces of information called *bits* (for *binary digits*). The amount of memory capacity is specified in increments of 1000 or in “K” increments, where 1 K is 1024 bytes of memory storage (a byte is 8 bits).

The program is stored in the memory as 1s and 0s, which are typically assembled in the form of 16-bit or 32 bit

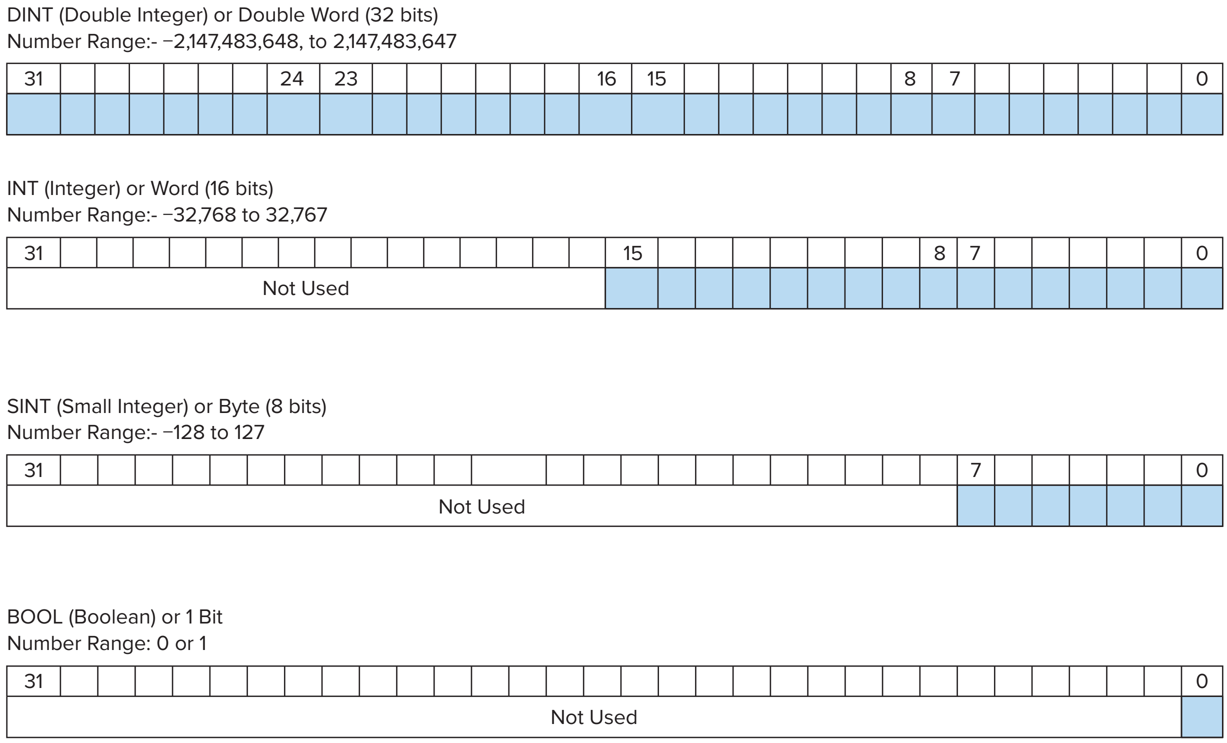


Figure 2-32 32 bit data types.

words. Memory sizes are commonly expressed in thousands of words that can be stored in the system; thus 2 K is a memory of 2000 words, and 64 K is a memory of 64,000 words. The memory size varies from as small as 1 K for small systems to 32 MB for very large systems. Memory capacity is an important prerequisite for determining whether a particular processor will handle the requirements of the specific application.

Memory *location* refers to an address in the CPU's memory where a binary word can be stored. The RSLogix 5000 memory is built on a 32-bit platform. The controller can process 32 bits of information at once and memory is grouped in 32-bit sections. Allen-Bradley refers to 32 bits of data as a double integer (**DINT**). Each bit in memory has its own unique address that can be used by instructions to read and/or write states. Some program instructions work with single bits of memory and are called **Boolean** instructions. Other instructions work with all 32 bits at on time and are classified as **word** instructions. Figure 2-32 shows how tag names are classified in terms of memory reserved. Memory *utilization* refers to the number of memory locations required to store each type of instruction. A rule of thumb for memory locations is one location per coil or contact. One K of memory would then allow a program containing 1000 coils and contacts to be stored in memory.

The memory of a PLC may be broken into sections that have specific functions. Sections of memory used to store the status of inputs and outputs are called input status files or tables and output status files or tables. These terms simply refer to a location where the status of an input or output device is stored. Each bit is either a 1 or 0, depending on whether the input is open or closed. A closed contact would have a binary 1 stored in its respective location in the input table, whereas an open contact would have a 0 stored. A lamp that is ON would have a 1 stored in its respective location in the output table, whereas a lamp that is OFF would have a 0 stored. Input and output image tables are constantly being revised by the CPU. Each time a memory location is examined, the table changes if the contact or coil has changed state.

PLCs execute memory-checking routines to be sure that the PLC memory has not been corrupted. This memory checking is undertaken for safety reasons. It helps ensure that the PLC will not execute if memory is corrupted.

2.8 Memory Types

Memory can be placed into two general categories: volatile and nonvolatile. Volatile memory will lose its stored information if all operating power is lost or removed. Volatile memory is easily altered and is quite suitable for most applications when supported by battery backup.

Nonvolatile memory has the ability to retain stored information when power is removed accidentally or intentionally. As the name implies, programmable logic controllers have programmable memory that allows users to develop and modify control programs. This memory is made nonvolatile so that if power is lost, the PLC holds its programming.

Read Only Memory (ROM) stores programs, and data cannot be changed after the memory chip has been manufactured. ROM is normally used to store the programs and data that define the capabilities of the PLC. ROM memory is nonvolatile, meaning that its contents will not be lost if power is lost. ROM is used by the PLC for the operating system. The operating system is burned into ROM by the PLC manufacturer and controls the system software that the user uses to program the PLC. When Allen Bradley burns the operating system into memory it is called PROM (programmable read-only memory).

Random Access Memory (RAM), sometimes referred to as *read-write (R/W) memory,* is designed so that information can be written into or read from the memory. RAM is used as a temporary storage area of data that may need to be quickly changed. RAM is volatile, meaning that the data stored in RAM will be lost if power is lost. A battery backup is required to avoid losing data in the event of a power loss (Figure 2-33). Most PLCs use CMOS-RAM technology for user memory. CMOS-RAM chips have very low current draw and can maintain memory with a lithium battery for an extended time, two to five years in many cases. Some processors have a capacitor that provides at least 30 minutes of battery backup when the battery is disconnected and power is OFF.

Erasable Programmable Read-Only Memory (EPROM) provides some level of security against unauthorized or unwanted changes in a program. EPROMs are designed so that data stored in them can be read, but not easily altered without special equipment. For example, UV EPROMs (ultraviolet erasable programmable read-only memory) can only be erased with an ultraviolet light. EPROM memory is used to back up, store, or transfer PLC programs.

Electrically erasable programmable read-only memory (EEPROM) is a nonvolatile memory that offers the same programming flexibility as does RAM. The EEPROM can be electrically overwritten with new data instead of being erased with ultraviolet light. Because the EEPROM is nonvolatile memory, it does not require battery backup. It provides permanent storage of the program and can be changed easily using standard programming devices. Typically, an EEPROM memory module is used to store, back up, or transfer PLC programs (Figure 2-34).

Flash EEPROMs are similar to EEPROMs in that they can only be used for backup storage. The main difference comes in the flash memory: they are extremely fast at saving and retrieving files. In addition, they do not need to be physically removed from the processor for reprogramming; this can be done using the circuitry within the processor module in which they reside. Flash memory is also sometimes built into the processor module (Figure 2-35), where it automatically backs up parts of RAM. If power fails while a PLC with flash memory is running, the PLC will resume running without having lost any working data after power is restored.

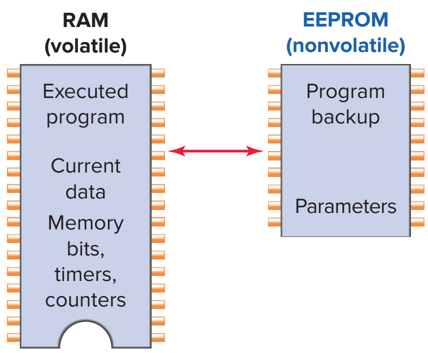


Figure 2-34 EEPROM memory module is used to store, back up, or transfer PLC programs.

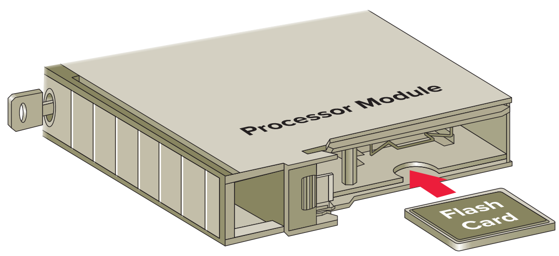


Figure 2-35 Flash memory card installed in a socket on the processor.

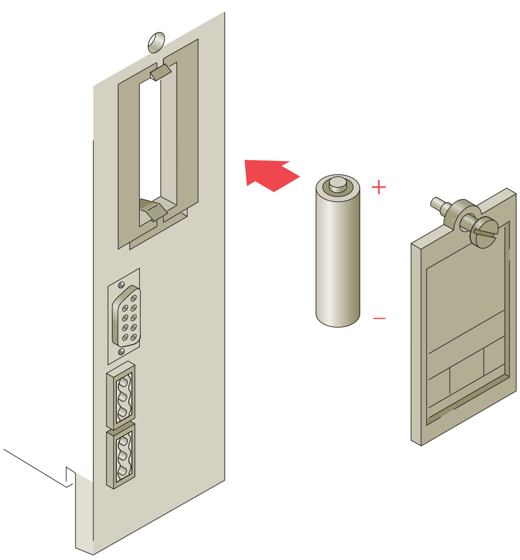


Figure 2-33 Battery used to back up processor RAM.

2.9 Programming Terminal Devices

A programming terminal device is needed to enter, modify, and troubleshoot the PLC program. PLC manufacturers use various types of programming devices. The simplest type is the hand-held type programmer shown in Figure 2-36. This proprietary programming device has a connecting cable so that it can be plugged into a PLC's programming port. Hand-held programmers have limited display capabilities. Some units will display only the last instruction that has been programmed, whereas other units will display from two to four rungs of ladder logic.

The most popular method of PLC programming is to use a personal computer (PC) in conjunction with the manufacturer's programming software (Figure 2-37). Typical capabilities of the programming software include online and offline program editing, online program monitoring, program documentation, diagnosing malfunctions in the PLC, and troubleshooting the controlled system. Hard-copy reports generated in the software can be printed on the computer's printer. Most software packages will not allow you to develop programs on another manufacturer's PLC. In some cases, a single manufacturer will have multiple PLC families, each requiring its own software to program.

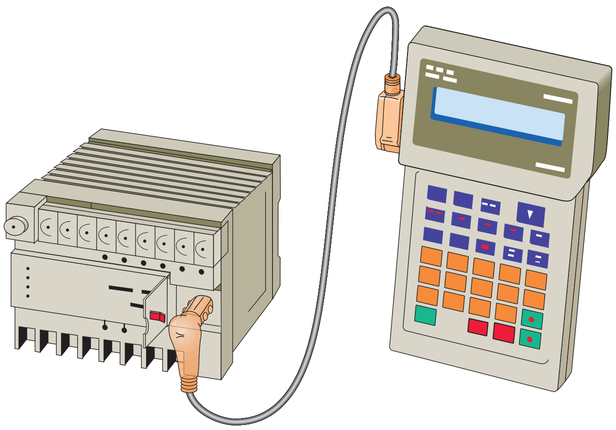


Figure 2-36 Hand-held programming terminal.

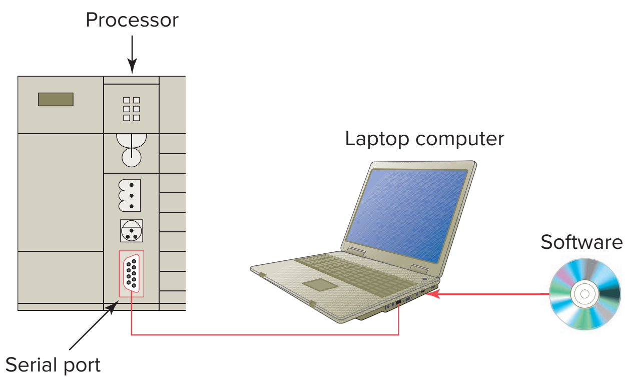


Figure 2-37 Personal computer used as the programming device.

2.10 Recording and Retrieving Data

Printers are used to provide hard-copy printouts of the processor's memory in ladder program format. Lengthy ladder programs cannot be shown completely on a screen. Typically, a screen shows a maximum of five rungs at a time. A printout can show programs of any length and analyze the complete program.

The PLC can have only one program in memory at a time. To change the program in the PLC, it is necessary either to enter a new program directly from the keyboard or to download one from the computer hard drive. Some CPUs support the use of a memory cartridge that provides portable EEPROM storage for the user program. The cartridge can be used to copy a program from one PLC to another similar type PLC.

2.11 Human Machine Interfaces (HMIs)

In the past, the typical user interface to a control system consisted of a panel with switches, pushbuttons, pilot lights, gauges, analog meters, and the like. With the advent of digital control systems, larger hard-wired panels have been replaced by a computer screen with process graphics and operator commands entered via a keyboard (Figure 2-38).

Human machine interfaces give the ability to the operator and to management to view the operation in real time. Through personal computer–based setup software, you can configure display screens to:

- Replace hardwired pushbuttons and pilot lights with realistic-looking icons. The machine operator need only touch the display panel to activate the pushbuttons.

Figure 2-38 Human Machine Interface (HMI).
Source: genkur/Shutterstock

- Show operations in graphic format for easier viewing.
- Allow the operator to change timer and counter presets by touching the numeric keypad graphic on the touch screen.
- Show alarms, complete with time of occurrence and location.
- Display variables as they change over time.

Human Machine Interfaces (HMIs) are also referred to as User Interface, Operator Panel, or Terminal and provide a means of *controlling*, *monitoring, managing*, and/or visualizing device processes. They can be located on the machine or in centralized control rooms. The general structure of an HMI package is shown in Figure 2-39. The tag database variables are programmed to interact with the graphic screen objects and communicate with the PLC through the I/O server. If required the screen light can be programmed to change from Green to Red in response whenever the process transitions from Stop to Start.

The design of the HMI application plays a critical role in determining the operator's ability to effectively manage the operation, particularly in response to abnormal situations. The major tasks in the development of an HMI application are:

- *Set up the communication with the PLC*. This involves configuring all necessary software and hardware components.
- *Create the tag database*. Most HMI packages provide a way to import tags from the PLC programming software.
- *Insert the graphical objects on the screen*. Graphics are drawn or imported from a library of common objects.
- *Animate the objects*. There are two basic types of animation: user input and display. User input types allow an operator to change tag values. A display animation allows a value to be displayed and also allows an object to change shape, position, and color.

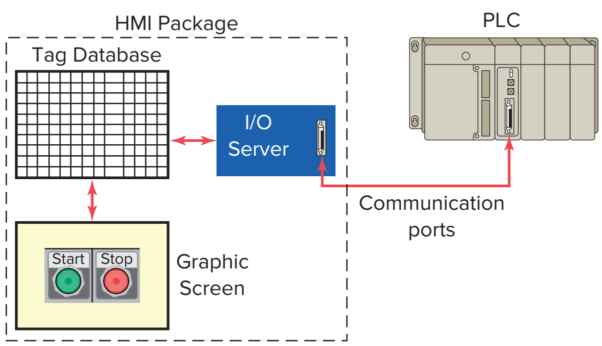


Figure 2-39 General structure of an HMI package.

Many different types of HMI hardware and software features are available. These include:

HMI MONITOR AND ENCLOSURE

HMI operator panels typically contain monochrome or 256 color display screens. These systems often communicate directly with the PLC to read or write memory locations.

- A monochrome monitor uses one color for the background and another to display text or images on the screen.
- Color displays enable clearer process representation and in general brighten up their systems. The color convention for status and alarms should follow the same convention as their hardwired equivalents, namely:
  - *Red*—for alarm, danger, and stop
  - *Yellow*—for caution and risk of danger
  - *Green*—for ready, running, and safe condition
- Screen resolution is expressed as width × height, with the units in pixels.
- Screen memory is expressed in Megabytes (MB).
- The environmental certification refers to the type of electrical enclosure used to protect their contents from troublesome operating conditions such as dust, liquids, and extreme variations in temperature.
- The screen is usually touch-sensitive. The touch-sensitive screen allows for more devices and data to be displayed in a smaller area. Detailed information about an object can be accessed by touching the object.

Serial or Ethernet connections are most commonly used when connecting a human machine interface to a PLC. **Serial-based** connection can handle fair amounts of data over distances of up to 100 feet. Ethernet-based connections can handle very large amounts of data more quickly over moderate distances up to 300 feet.

ALARMS

Alarms are messages which indicate that a fault condition is present (Figure 2-40). An alarm summary can present a complete list of timestamped active alarms. Typically an alarm can exist in the following states:

- *Inactive*—The condition being monitored does not have any faults present, and there is no associated alarm message waiting to be acknowledged.
- *Active*—A fault condition is present, and the alarm message has not been acknowledged by the operator.
- *Acknowledged*—The fault condition is present, and the operator has acknowledged the alarm message.

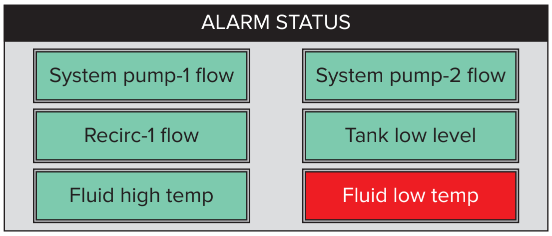


Figure 2-40 Typical alarm status screen.

- *OK*—The fault condition is no longer present, but the operator has not acknowledged the alarm message yet.

EVENT HISTORY

An event history presents a time-stamped list of all significant events that have occurred in the process. Many problems within the plant or equipment may occur when no one is monitoring the system, and intermittent problems may be difficult to diagnose without a history of previous issues.

TREND

Values of important process variables, such as flow, temperature, and production rate, over a period of time are shown by this type of display. This type of display provides the ability to chart the progress of the process in real time, providing the same function as a strip chart recorder. For example, suppose you are monitoring pressure of a Pounds per Square Inch Gauge (psig) as shown in Figure 2-41. According to the table, you can see that it's OK right now, but that's all you know. This trend shows the pressure oscillating around a known good level. We may want to check on the cause of oscillation, but there appears to be no immediate problem.

GRAPHICS LIBRARY

The graphics library contained within an HMI development package provides buttons, lights, switches, sliders,

Current Pressure	Alarm Level	Shutdown Level
235.2 psig	250 psig	300 psig

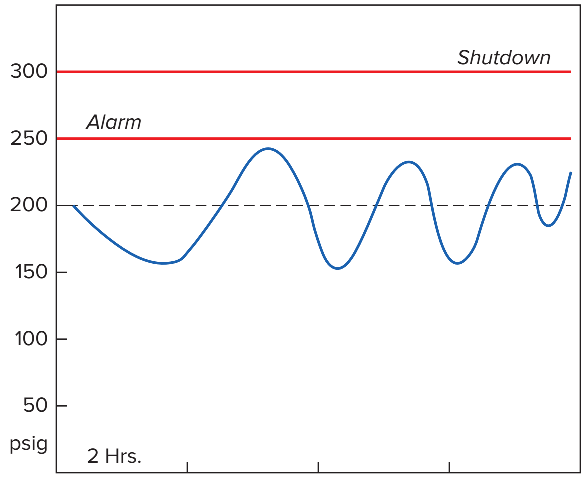


Figure 2-41 Trend monitoring of a pressure gauge.

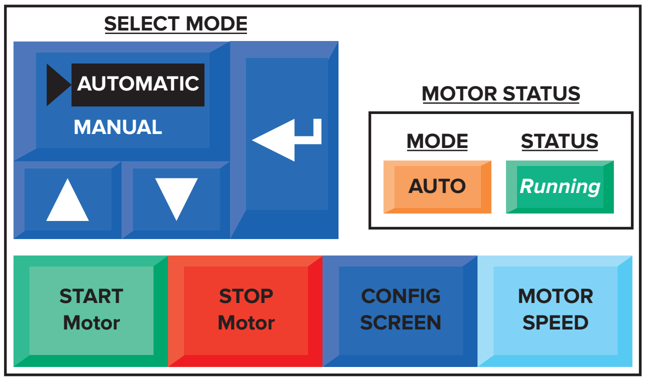


Figure 2-42 Typical motor control graphics.

meters, fills, and other graphic objects (Figure 2-42). It saves design time by providing graphics and faceplates for numerous industrial control devices that would otherwise have to be created manually. Librarian applications may include easy-to-use features for resizing, changing color scheme, and orientation of objects, as well as building your own graphics into the library.

CHAPTER 2 REVIEW QUESTIONS

1. What is the function of a PLC input interface module?
2. What is the function of a PLC output interface module?
3. With reference to a PLC rack:
   a. What is a *remote rack?*
   b. Why are remote racks used?
4. How does the processor identify the location of a specific input or output device?
5. List the three basic elements of rack/slot-based addressing.
6. Compare bit level and word level addressing.
7. In what way does tag-based addressing differ from rack/slot-based addressing?
8. What do PC-based control systems use to interface with field devices?
9. What type of I/O modules have both inputs and outputs connected to them?
10. In addition to field devices, what other connections are made to a PLC module?
11. Most PLC modules use plug-in wiring terminal strips. Why?
12. What are the advantage and the disadvantage of using high-density modules?
13. With reference to PLC discrete input modules:
    a. What types of field input devices are suitable for use with them?
    b. List three examples of discrete input devices.
14. With reference to PLC discrete output modules:
    a. What types of field output devices are suitable for use with them?
    b. List three examples of discrete output devices.
15. Explain the function of the backplane of a PLC rack.
16. What is the function of the optical isolator circuit used in discrete I/O module circuits?
17. Name the two distinct sections of an I/O module.
18. List four tasks performed by a discrete input module.
19. What electronic element can be used as the switching device for a 120 VAC discrete output interface module?
20. With reference to discrete output module current ratings:
    a. What is the maximum current rating for a typical 120 VAC output module?
    b. Explain one method of handling outputs with larger current requirements.
21. What electronic element can be used as the switching device for DC discrete output modules?
22. A discrete relay type output module can be used to switch either AC or DC load devices. Why?
23. With reference to sourcing and sinking I/O modules:
    a. What current relationship are the terms *sourcing* and *sinking* used to describe?
    b. If an I/O module is specified as a current-sinking type, then which type of field device (sinking or sourcing) it is electrically compatible with?
24. Compare discrete and analog I/O modules with respect to the type of input or output devices with which they can be used.
25. Explain the function of the analog-to-digital (A/D) converter circuit used in analog input modules.
26. Explain the function of the digital-to-analog (D/A) converter circuit used in analog output modules.
27. Name the two general sensing classifications for analog input modules.
28. List five common physical quantities measured by a PLC analog input module.
29. What type of cable is used when connecting a thermocouple to a voltage sensing analog input module? Why?
30. Explain the difference between a unipolar and bipolar analog input module.
31. The resolution of an analog input channel is specified as 0.3 mV. What does this tell you?
32. In what two ways can the loop power for current sensing input modules be supplied?
33. List three field devices that are commonly controlled by a PLC analog output module.
34. State one application for each of the following special I/O modules:
    a. High-speed counter module
    b. Encoder-counter module
    c. Stepper-motor module
35. List one application for each of the following intelligent I/O modules:
    a. Motion and position control module
    b. Communication module

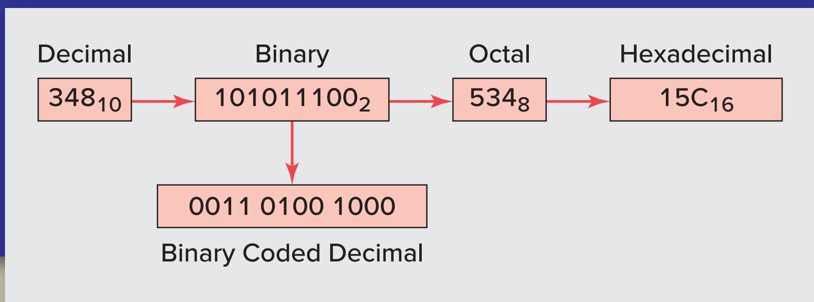
36. Write a short explanation for each of the following discrete I/O module specifications:
    a. Nominal input voltage
    b. Input threshold voltages
    c. Nominal current per input
    d. Ambient temperature rating
    e. Input ON/OFF delay
    f. Output voltage
    g. Output current
    h. Inrush current
    i. Short circuit protection
    j. Leakage current
    k. Electrical isolation
    l. Points per module
    m. Backplane current draw
37. Write a short explanation for each of the following analog I/O module specifications:
    a. Channels per module
    b. Input current/voltage range(s)
    c. Output current/voltage range(s)
    d. Input protection
    e. Resolution
    f. Input impedance and capacitance
    g. Common-mode rejection
38. Compare the function of the CPU and memory sections of a PLC processor.
39. With reference to the PLC chassis power supply:
    a. What conversion of power takes place within the power supply circuit?
    b. Explain the term *hold-up time* as it applies to the power supply.
40. Explain the purpose of a redundant PLC processor.
41. Describe three typical modes of operation that can be selected by the keyswitch of a processor.
42. State five other functions, in addition to simple logic processing, that PLC processors are capable of performing.
43. List five important procedures to follow when handling static-sensitive PLC components.
44. Define each of the following terms as they apply to the memory element of a PLC:
    a. writing
    b. reading
    c. bits
    d. location
    e. utilization
45. With reference to the I/O image tables:
    a. What information is stored in PLC input and output tables?
    b. What is the input status of a closed switch stored as?
    c. What is the input status of an open switch stored as?
    d. What is the status of an output that is ON stored as?
    e. What is the status of an output that is OFF stored as?
46. Why do PLCs execute memory-checking routines?
47. Compare the memory storage characteristics of volatile and nonvolatile memory elements.
48. What information is normally stored in the ROM memory of a PLC?
49. What information is normally stored in the RAM memory of a PLC?
50. What information is normally stored in an EEPROM memory module?
51. What are the advantages of a processor that utilizes a flash memory card?
52. List three functions of a PLC programming terminal device.
53. Give one advantage and one limitation to the use of hand-held programming devices.
54. What is required for a personal computer to be used as a PLC programming terminal?
55. List four important capabilities of PLC programming software.
56. How many programs can a PLC have stored in memory at any one time?
57. Outline four functions that an HMI interface screen can be configured to perform.
58. List the four major tasks in the development of an HMI application.
59. What information does an HMI trend display convey?
60. Define the term scaling as it applies to PLC inputs and outputs.
61. What is the function of a transducer?
62. In a tag based PLC memory structure, what is the function of a base tag and an alias tag?

CHAPTER 2 PROBLEMS

1. A discrete 120 VAC output module is to be used to control a 230 VDC solenoid valve. Draw a diagram showing how this could be accomplished using an interposing relay.
2. Assume a thermocouple, which supplies the input to an analog input module, generates a linear voltage of from 20 to 50 mV when the temperature changes from 750 to 1250°F. How much voltage will be generated when the temperature of the thermocouple is at 1000°F?
3. With reference to I/O module specifications:
   a. If the ON-delay time of a given discrete input module is specified as 12 ms, how much is this expressed in seconds?
   b. If the output leakage current of a discrete output module is specified as 950 μA, how much is this expressed in amperes?
   c. If the ambient temperature rating for an I/O module is specified as 60°C, how much is this expressed in degrees Fahrenheit?
4. Assume the triac of an AC discrete output module fails in the shorted state. How would this affect the device connected to this output?
5. A personal computer is to be used to program several different PLCs from different manufacturers. What would be required?
6. Create RSLogix 5000 base and alias tags for each of the following memory locations:
   a. A limit switch connected to terminal 3 of a 16-point input module located in slot 1 of the local chassis.
   b. A solenoid connected to terminal 8 of a 16-point output module located in slot 2 of the local chassis.

3

Number Systems and Codes



Chapter Objectives

After completing this chapter, you will be able to:

- Define the decimal, binary, octal, and hexadecimal numbering systems and be able to convert from one numbering or coding system to another
- Explain the BCD, Gray, and ASCII code systems
- Define the terms *bit, byte, word, least significant bit (LSB),* and *most significant bit (MSB)* as they apply to binary memory locations
- Add, subtract, multiply, and divide binary numbers

Using PLCs requires us to become familiar with other number systems besides decimal. Some PLC models and individual PLC functions use other numbering systems. This chapter deals with some of these numbering systems, including binary, octal, hexadecimal, BCD, Gray, and ASCII codes. The basics of each system, as well as conversion from one system to another, are explained.