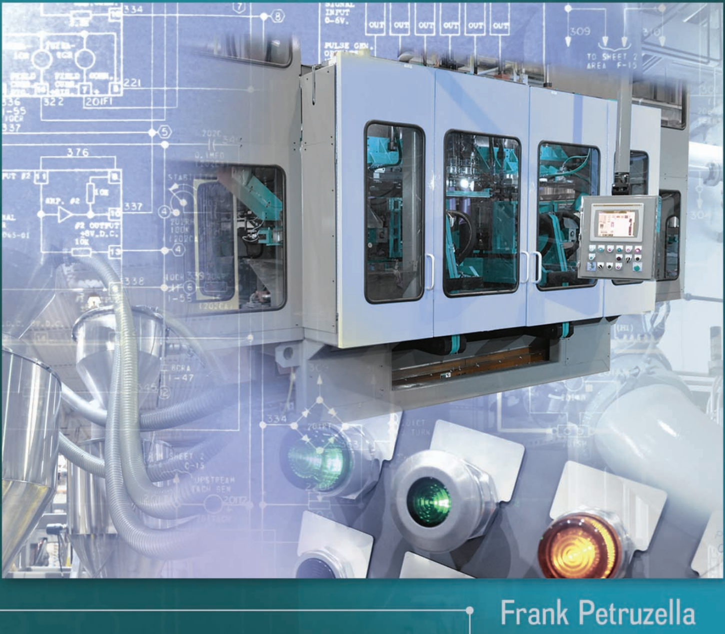
Programmable Logic Controllers

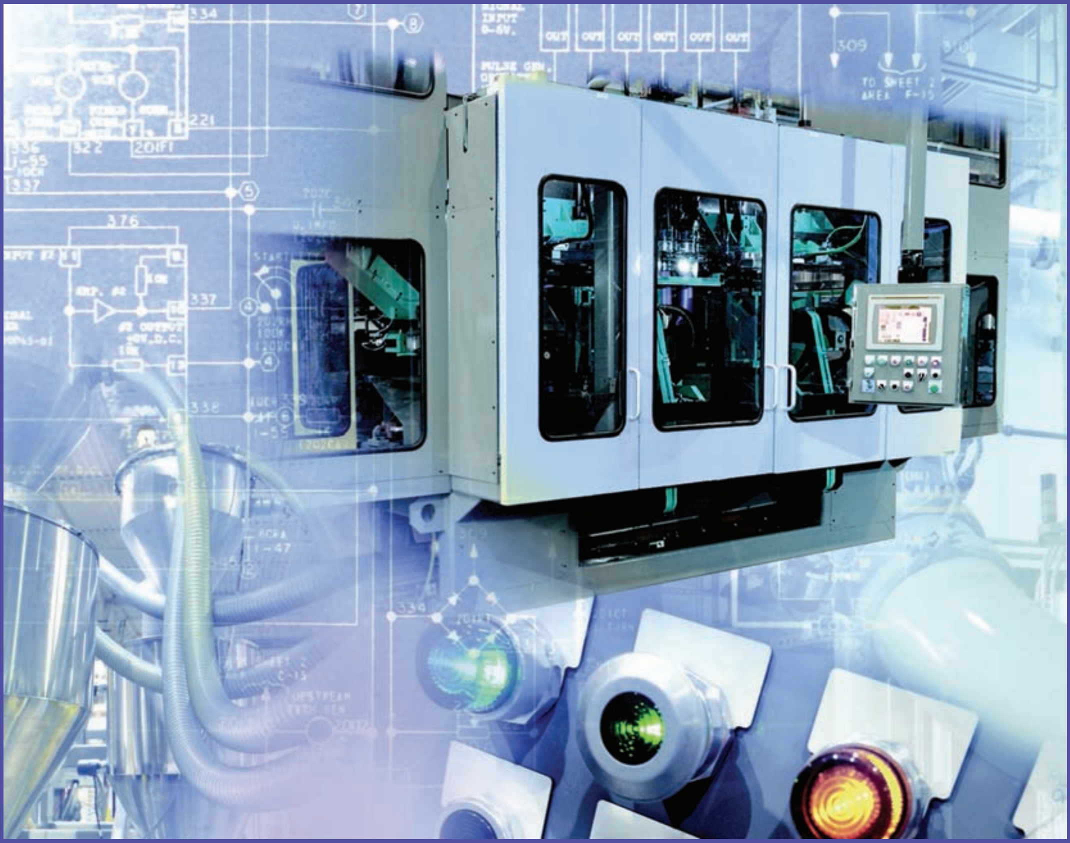
FIFTH EDITION



Frank Petruzella

Mc Graw Hill Education

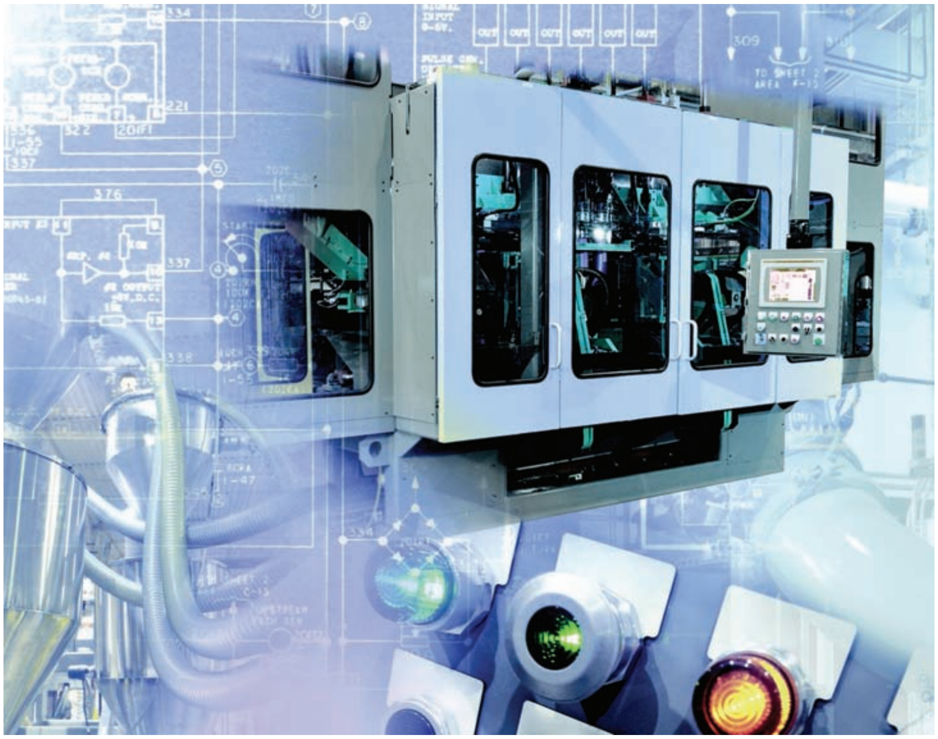
Programmable Logic Controllers



This page intentionally left blank

Programmable Logic Controllers

Fifth Edition



Frank D. Petruzella

Mc Graw Hill Education

PROGRAMMABLE LOGIC CONTROLLERS, FIFTH EDITION

Some ancillaries, including electronic and print components, may not be available to customers outside the United States.

This book is printed on acid-free paper.

1 2 3 4 5 6 7 8 9 0 RMN/RMN 1 0 9 8 7 6

ISBN 978-0-07-337384-3
MHID 0-07-337384-2

Senior Vice President, Products & Markets: *Kurt L. Strand*
Vice President, General Manager, Products & Markets: *Marty Lange*
Vice President, Content Design & Delivery: *Kimberly Meriwether David*
Managing Director: *Thomas Timp*
Global Brand Manager: *Raghu Srinivasan*
Director, Product Development: *Rose Koos*
Product Developer: *Vincent Bradshaw*
Marketing Manager: *Nick McFadden*
Digital Product Developer: *Amy Bumbaco, Ph.D.*
Director, Content Design & Delivery: *Linda Avenarius*
Executive Program Manager: *Faye M. Herrig*
Content Project Managers: *Jessica Portz, Tammy Juran, Sandra Schnee*
Buyer: *Laura M. Fuller*
Content Licensing Specialist: *Lorraine Buczek*
Compositor: *MPS Limited*
Printer: *R. R. Donnelley*

All credits appearing on page or at the end of the book are considered to be an extension of the copyright page.

Library of Congress Cataloging-in-Publication Data

Petruzella, Frank D., author.
Programmable logic controllers / Frank D. Petruzella.—Fifth edition.
pages cm
Includes index.
ISBN 978-0-07-337384-3 (alk. paper)—ISBN 0-07-337384-2 (alk. paper) 1. Programmable controllers. I. Title.
TJ223.P76P48 2017
629.8’95—dc23
2015035302

The Internet addresses listed in the text were accurate at the time of publication. The inclusion of a website does not indicate an endorsement by the authors or McGraw-Hill Education, and McGraw-Hill Education does not guarantee the accuracy of the information presented at these sites.

mheducation.com/highered

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 nontraditional devices to be attached directly to the PLC.

Now in its Fifth Edition, changes made to the content of the text have been made **solely** based on reviews from current instructors and include:

- material that should be added or deleted from chapters
- topics requiring more in-depth coverage
- increased integration of the ControlLogix platform of controllers
- chapter modifications require to meet current curriculum needs

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 a SLC-500 or ControlLogix 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 bulled 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.

Features new to the Fifth Edition include:

- **Key concepts** and terms are **highlighted** in bold the first time they appear.
- New/updated **photos** and **line art** for every **chapter**.
- **New topics** for every **chapter** as requested by reviewers.
- Addition review **questions** for new topics.
- Updated instructor **PowerPoint** lessons.
- More than **175** SLC-500 and ControlLogix program simulation **videos tied directly** to the programs studied in the text

  In addition, students who are using McGraw-Hill's Connect can watch simulated, step-by-step execution of numerous ladder logic programming examples. They're **guided** by an audio commentary that explains what to look for as the program is executed. The videos are part of the Student Resources section of Connect.

Chapter changes in this edition include:

Chapter 1

- Testing of field devices.
- Extended coverage of scan cycle sequence.
- Additional test bank questions.
- Program video simulations.
- New and modified line diagrams and photos.

Chapter 2

- ControlLogix Base and Alias addressing.
- Extended coverage of DC module Sinking and Sourcing.
- Analog module input sensor 2-, 3-, and 4-wire connections.
- Scaling of PLC analog inputs and outputs.
- Extended coverage of Human Machine Interfaces (HMIs)
- Additional chapter review questions.
- Additional test bank questions.
- Program video simulations.
- New and modified line diagrams and photos.

Chapter 3

- 16 bit 2's complement.
- Floating point arithmetic.
- Additional chapter problems.
- Additional test bank questions.
- Program video simulations.
- New and modified line diagrams and photos.

Chapter 4

- Modification to hardwired programming examples
- Additional test bank questions.
- Additional chapter review questions.
- Program video simulations.
- New and modified line diagrams and photos.

Chapter 5

- Electrical versus logical continuity.
- Evaluating XIO and XIC bit instructions.
- Rack-based versus tag-based addressing.
- Connecting with analog devices.
- Additional test bank questions.
- Additional chapter review questions.
- Program video simulations.
- New and modified line diagrams and photos.

Chapter 6

- Magnetic reed float switch.
- Resistance temperature detectors (RTDs).
- Electrical interlocking circuits.
- Process instrumentation.
- Additional test bank questions.
- Additional chapter review questions.
- Program video simulations.
- New and modified line diagrams and photos.

Chapter 7

- Extended coverage of timer instructions.
- ControlLogix timer instruction.
- Reciprocating timers.
- TON timer bit table.
- TOF timer bit table.
- Additional test bank questions.
- Program video simulations.
- New and modified line diagrams and photos.

Chapter 8

- ControlLogix counter instruction.
- Extended coverage of CTD instruction.
- Additional information on incremental encoders.
- New section on High-Speed Counter instruction.
- Additional test bank questions.
- Program video simulations.
- New and modified line diagrams and photos.

Chapter 9

- Extended coverage of MCR instruction.
- Extended coverage of Jump instruction.
- Extended coverage of Immediate Input and Output instructions.
- ControlLogix Immediate Output instruction.
- Additional test bank questions.
- Program video simulations.
- New and modified line diagrams and photos.

Chapter 10

- Extended coverage of the Masked Move instruction.
- New example of a copy instruction program.
- New example of a data compare program.
- ControlLogix Limit Comparison instruction and program.
- Additional test bank questions.
- Program video simulations.
- New and modified line diagrams and photos.

Chapter 11

- Extended coverage of basic math instruction.
- New example of a compute instruction program.
- New coverage Modulo (MOD) instruction.
- New scale analog input using the SCP instruction.
- New scale analog output using the SCP instruction.
- Additional test bank questions.
- Program video simulations.
- New and modified line diagrams and photos.

Chapter 12

- Extended coverage of Sequencer Output (SQO) instruction.
- ControlLogix Sequencer Output (SQO) instruction and program.
- ControlLogix shift registers instruction and program.
- ControlLogix FIFO instruction and program.
- Additional test bank questions.
- Program video simulations.
- New and modified line diagrams and photos.

Chapter 13

- Extended coverage of communications using RSLinx and RSWho.
- Additional test bank questions.
- Program video simulations.
- New and modified line diagrams and photos.

Chapter 14

- SERCOS standard communication for motion control.
- HART communication protocol.
- SCADA alarm monitoring.
- FactoryTalk services platform.
- Additional test bank questions.
- Program video simulations.
- New and modified line diagrams and photos.

Chapter 15

Part 1

- Extended coverage of tag types.
- Program video simulations.
- New and modified line diagrams and photos.

Part 2

- Reversing conveyor motor program and operation.
- Motor pilot light internal relay program and operation.
- Latch/unlatch car wash program and operation.
- One-shot program instructions used in conjunction with math operations.
- Program video simulations.
- New and modified line diagrams and photos.

Part 3

- Cascading TON timers for timed event-driven routines program and operation
- Program video simulations.
- New and modified line diagrams and photos.

Part 4

- Combining Counter and Timer Functions program and operation.
- Program video simulations.
- New and modified line diagrams and photos.

Part 5

- Monitoring the setting of a thumbwheel switch program and operation.
- PLC program for three-speed control of a conveyor system program and operation.
- Conveyor parts tracking program and operation.
- Program video simulations.
- New and modified line diagrams and photos.
- Part 6 Function block parameters tab.
- Program video simulations.
- New and modified line diagrams and photos.

Acknowledgments

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

Noureddine Bekhouche
Jacksonville State University

Mark Bohnet
Northwest Iowa Community College

Michael Buck
Dakota County Technical College

Wayne Buroker
Waukesha County Technical College

Jerry Clark
Northwest Mississippi Community College

Chris Haley
North Georgia Technical College

Garrett Hunter
Western Illinois University

Wael Ibrahim
ECPI University

Ahmed Kamal
Tennessee Tech University

Gholam H. Massiha
University of Louisiana at Lafayette

Randy Owens
Henderson Community College

James Schabowski
Waukesha County Technical College

Jenifer Shannon
Penn State University, Berks Campus

Accounties Lashan Smith
Tri-County Technical College

Kenneth E. Swayne
Pellissippi State Community College

John Veitch
SUNY Adirondack

William Walker
Truckee Meadows Community College

Robert Permenter
Albany Technical College

A special thanks to Don Pelster of Nashville State Community 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

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.

Programmable 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.

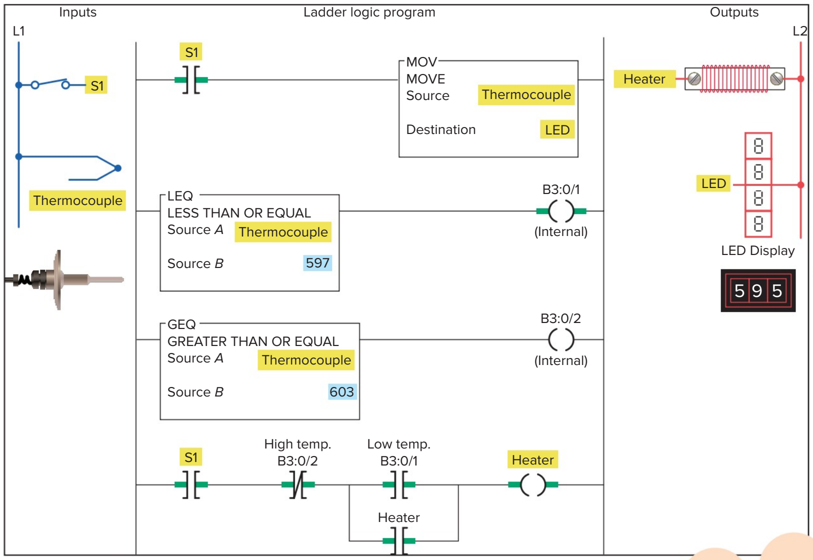
CHAPTER OBJECTIVES overview the chapter, letting students and instructors focus on the main points to better grasp concepts and retain information.

Chapter Objectives

After completing this chapter, you will be able to:

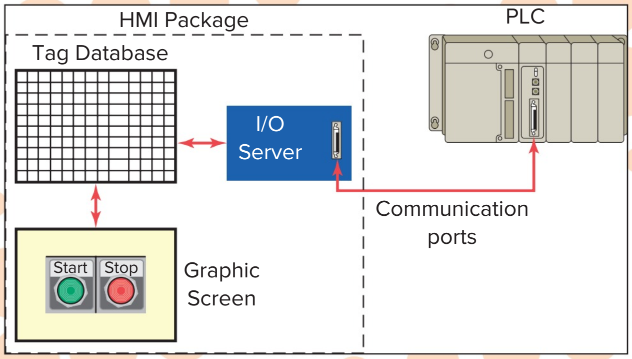
- Describe the operation of pneumatic on-delay and off-delay timers
- Describe PLC timer instruction and differentiate between a nonretentive and retentive timer
- Convert fundamental timer relay schematic diagrams to PLC ladder logic programs
- Analyze and interpret typical PLC timer ladder logic programs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

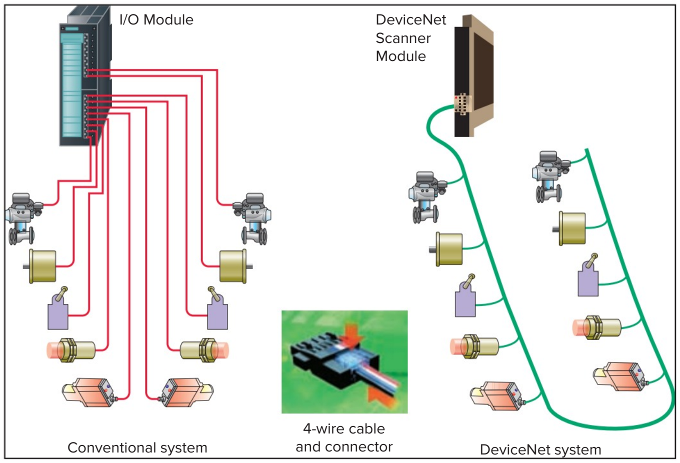


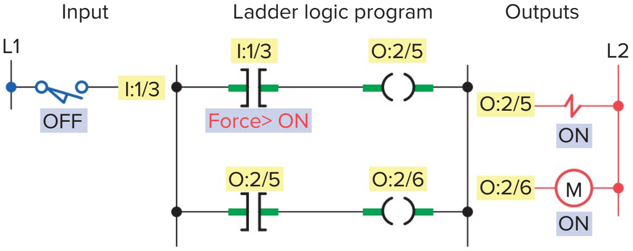
Here, drawings and photos of real-world input and output devices have been included

In Chapter 02, 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





- The processor ignores the actual state of input limit switch I:1/3.
- Although limit switch I:1/3 is off (0 or false) the processor considers it as being in the on (1 or true) state.
- The program scan records this, and the program is executed with this forced status.
- In other words, the program is executed as if the limit switch were actually closed.

BULLETED LISTS break down processes to helpfully summarize execution of tasks

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

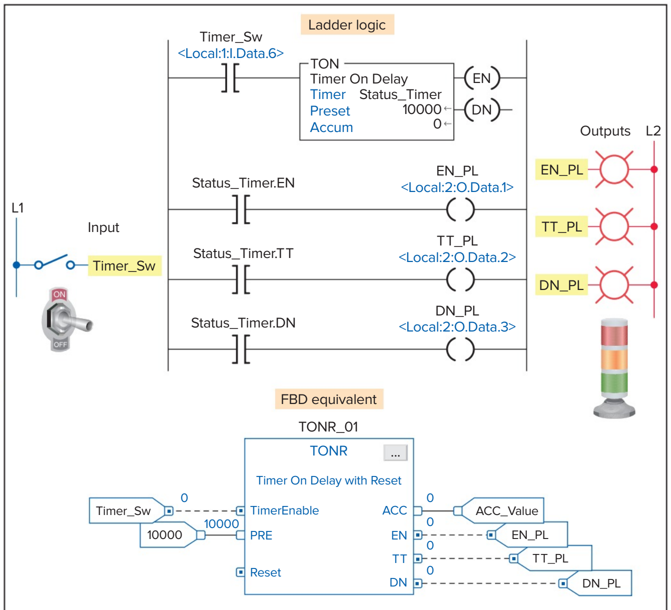
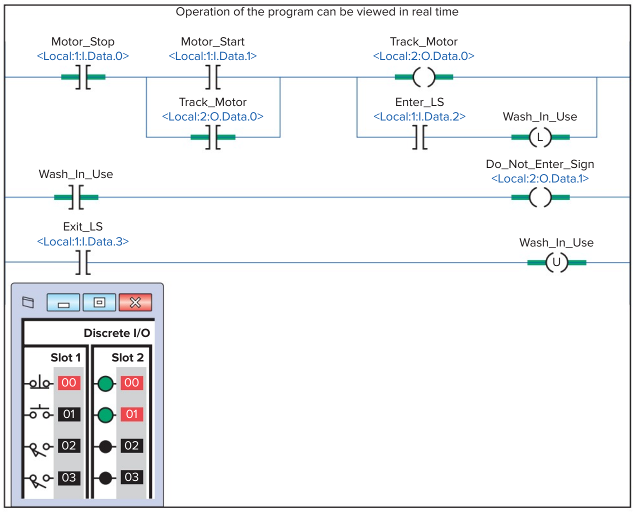
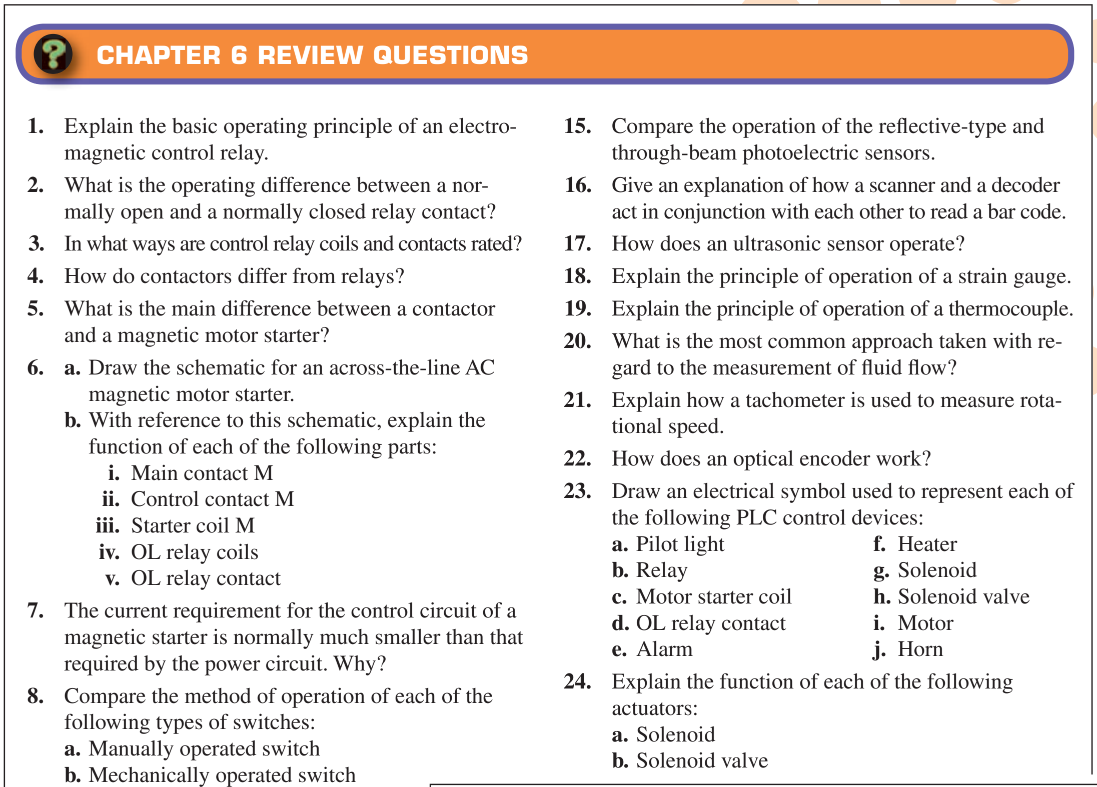


Figure 15-110 Comparison between ladder logic and the FBD equivalent for a 10 second TON and TONR timer.



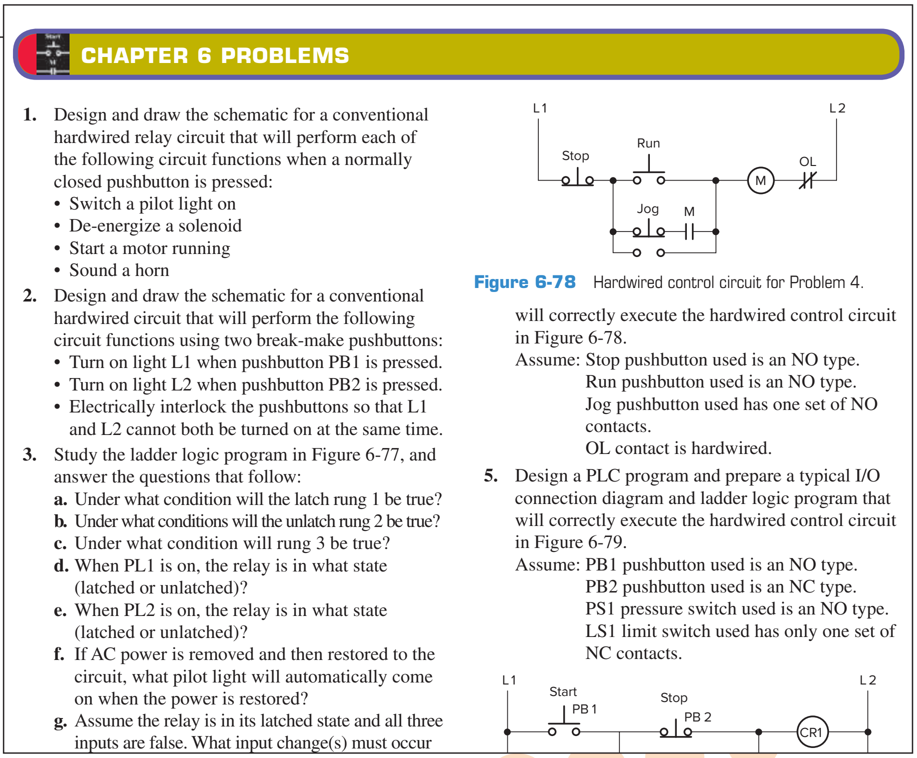
More than **175** SLC-500 and ControlLogix program simulation **videos tied directly** to the programs studied in the text

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 **f.** Heater
   b. Relay **g.** Solenoid
   c. Motor starter coil **h.** Solenoid valve
   d. OL relay contact **i.** Motor
   e. Alarm **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 6-77, 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 6-78 Hardwired control circuit for Problem 4.

will correctly execute the hardwired control circuit in Figure 6-78.
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.

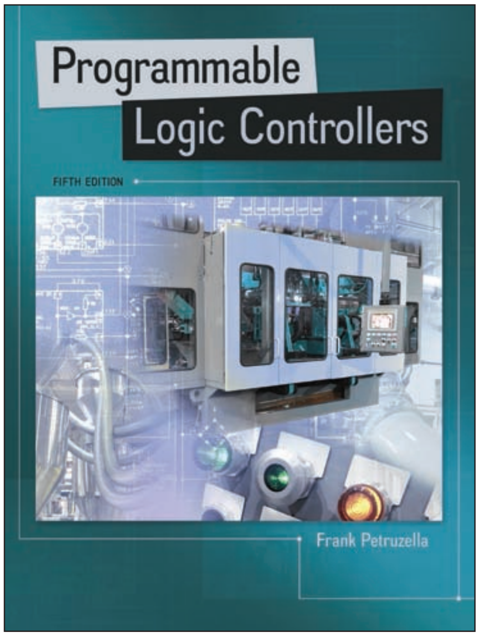
5. Design a PLC program and prepare a typical I/O connection diagram and ladder logic program that will correctly execute the hardwired control circuit in Figure 6-79.
   Assume: PB1 pushbutton used is an NO type.
   PB2 pushbutton used is an NC type.
   PS1 pressure switch used is an NO type.
   LS1 limit switch used has only one set of NC contacts.

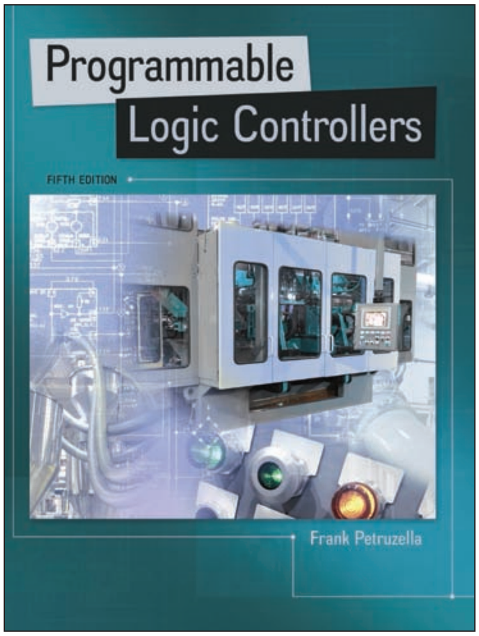
EXAMPLE PROBLEMS help bring home the applicability of chapter concepts

ANCILLARIES THAT WORK

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

- ExamView and EZ Test question test banks for each chapter.
- PowerPoint lessons with animations that help visualize the actual process.
- Activity Manual contains true/false, completion, matching, and multiple-choice tests for every chapter in the text. So that students get a better understanding of programmable logic controllers, the manual also includes a wide range of programming assignments and additional practice exercises.
- Answers to the questions and problems in the textbook, Activities Manual, and LogixPro Manual. Available on the Instructor Resources section of Connect.





In addition, for students, this edition also has available:

- *LogixPro PLC Lab Manual for use with Programmable Logic Controllers* Fifth Edition, with LogixPro PLC Simulator. This manual contains:
- McGraw-Hill's Connect and Smartbook.
- LogixPro simulations with audio and video for those using Connect.
  - Over 250 LogixPro student lab exercises sequenced to support material covered in the text.

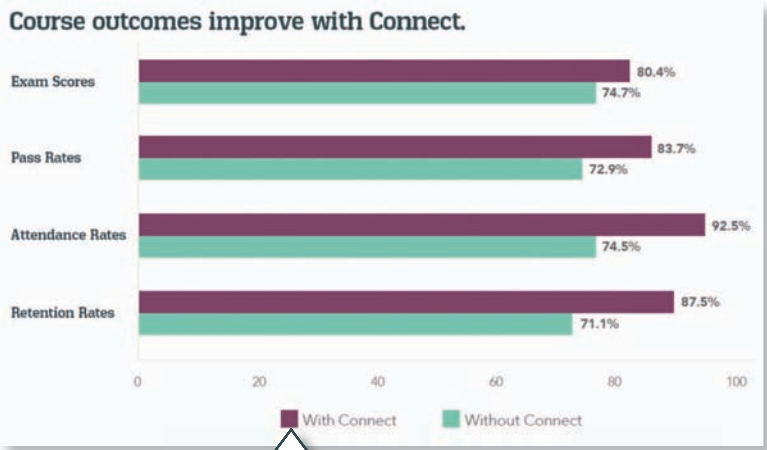
Mc Graw Hill Education connect®

Required=Results

McGraw-Hill Connect® Learn Without Limits

Connect is a teaching and learning platform that is proven to deliver better results for students and instructors.

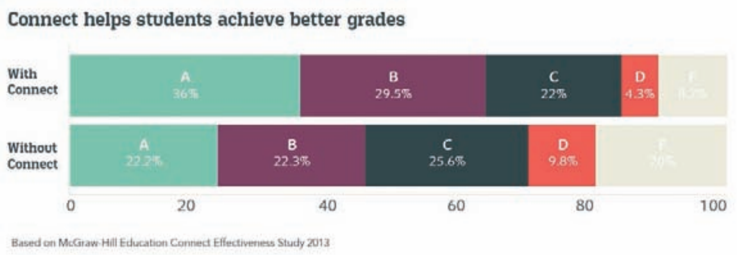
Connect empowers students by continually adapting to deliver precisely what they need, when they need it, and how they need it, so your class time is more engaging and effective.


Course outcomes improve with Connect.

Using **Connect** improves passing rates by **10.8%** and retention by **16.4%.**

88% of instructors who use **Connect** require it; instructor satisfaction **increases** by 38% when **Connect** is required.

Analytics


Connect helps students achieve better grades

Based on McGraw-Hill Education Connect Effectiveness Study 2013

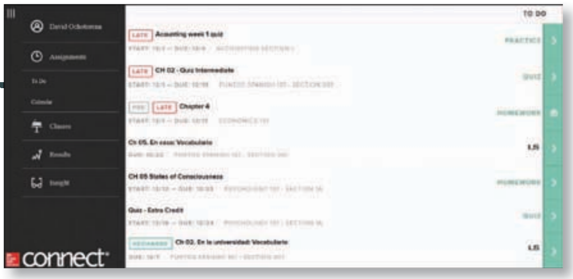
Connect Insight®

Connect Insight is Connect's new one-of-a-kind visual analytics dashboard—now available for both instructors and students—that provides at-a-glance information regarding student performance, which is immediately actionable. By presenting assignment, assessment, and topical performance results together with a time metric that is easily visible for aggregate or individual results, Connect Insight gives the user the ability to take a just-in-time approach to teaching and learning, which was never before available. Connect Insight presents data that empowers students and helps instructors improve class performance in a way that is efficient and effective.

Students can view their results for any **Connect** course.

Mobile

Connect's new, intuitive mobile interface gives students and instructors flexible and convenient, anytime–anywhere access to all components of the Connect platform.



Accounting week 1 quiz

CH 02 - Quiz Intermediate

Chapter 4

Ch 05. En casa: Vocabulario

CH 05 States of Consciousness

Quiz - Extra Credit

Ch 02. En la universidad: Vocabulario

Adaptive

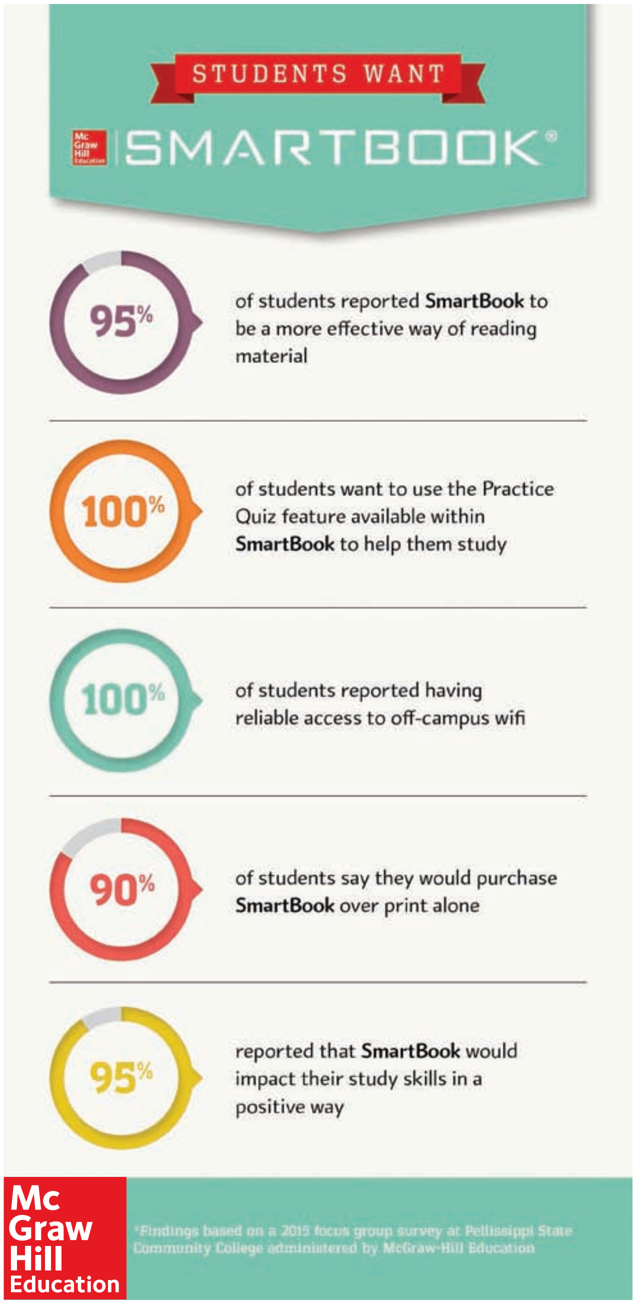
THE FIRST AND ONLY **ADAPTIVE READING EXPERIENCE** DESIGNED TO TRANSFORM THE WAY STUDENTS READ

> More students earn **A's** and **B's** when they use McGraw-Hill Education **Adaptive** products.

SmartBook®

Proven to help students improve grades and study more efficiently, SmartBook contains the same content within the print book, but actively tailors that content to the needs of the individual. SmartBook's adaptive technology provides precise, personalized instruction on what the student should do next, guiding the student to master and remember key concepts, targeting gaps in knowledge and offering customized feedback, and driving the student toward comprehension and retention of the subject matter. Available on smartphones and tablets, SmartBook puts learning at the student's fingertips—anywhere, anytime.

> Over **4 billion questions** have been answered, making McGraw-Hill Education products more intelligent, reliable, and precise.



95% of students reported **SmartBook** to be a more effective way of reading material

100% of students want to use the Practice Quiz feature available within **SmartBook** to help them study

100% of students reported having reliable access to off-campus wifi

90% of students say they would purchase **SmartBook** over print alone

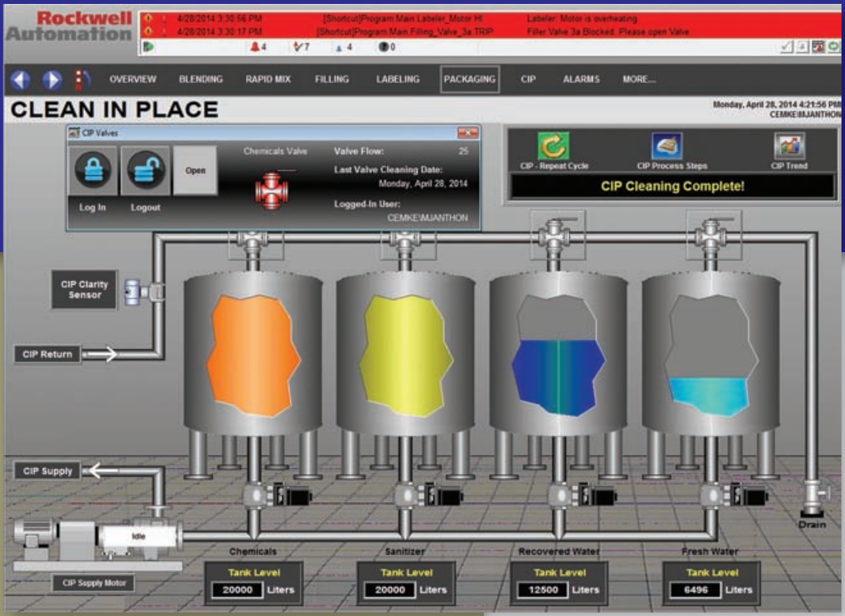
95% reported that **SmartBook** would impact their study skills in a positive way

*Findings based on a 2015 focus group survey at Pellissippi State Community College administered by McGraw-Hill Education

1

Programmable Logic Controllers (PLCs)

An Overview


4/28/2014 3:30:56 PM [Shortcut]Program Main Labeler_Motor HI Labeler: Motor is overheating
4/28/2014 3:30:17 PM [Shortcut]Program Main Filling_Valve_3a TRIP Filler Valve 3a Blocked. Please open Valve

Monday, April 28, 2014 4:21:56 PM
CEMKE\MJANTHON

Chemicals Valve
Valve Flow: 25
Last Valve Cleaning Date: Monday, April 28, 2014
Logged-In User: CEMKE\MJANTHON

CIP Cleaning Complete!

Chemicals Tank Level 20000 Liters
Sanitizer Tank Level 20000 Liters
Recovered Water Tank Level 12500 Liters
Fresh Water Tank Level 6496 Liters

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 (Figure 1-1) 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.

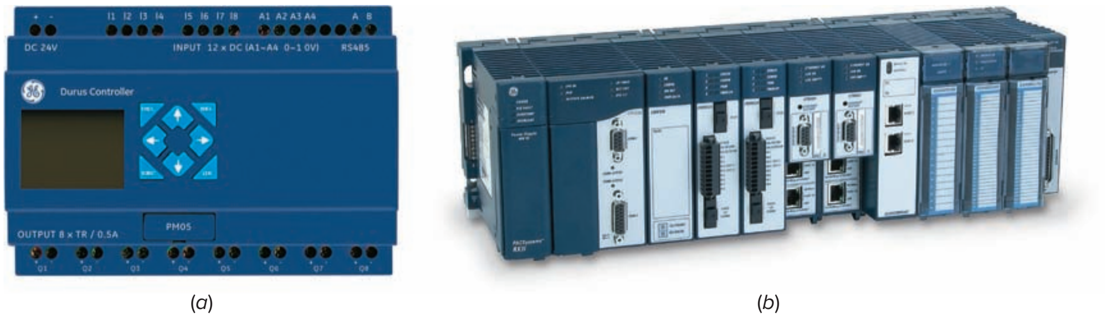
The PLC is, then, basically a digital computer designed for use in machine control. Unlike a personal computer, it has been designed to operate in the industrial environment and is equipped with special input/output interfaces and a control programming language. The common abbreviation used in industry for these devices, PC, can be confusing because it is also the abbreviation for "personal computer." Therefore, most manufacturers refer to their programmable controller as a PLC, which stands for "programmable logic controller."

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 1-2). 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:

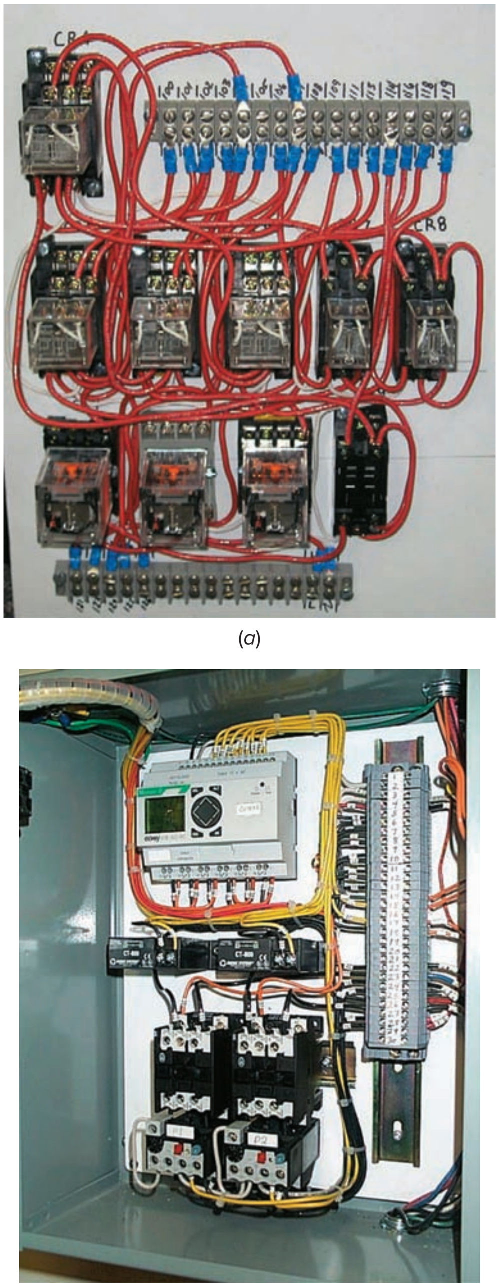
- ***Increased Reliability.*** Once a program has been written and tested, it can be easily downloaded to other PLCs. Since all the logic is contained in the PLC's memory, there is no chance of making a logic wiring error (Figure 1-3). The program takes the place of much of the external wiring that would 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 1-4). 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



(*a*)

(*b*)

Figure 1-1 Programmable logic controller.
Source: (*a–b*) Courtesy GE Intelligent Platforms.



(*a*)

(*b*)

Figure 1-2 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 Ramco Electric, Ltd.

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 (Figure 1-5).

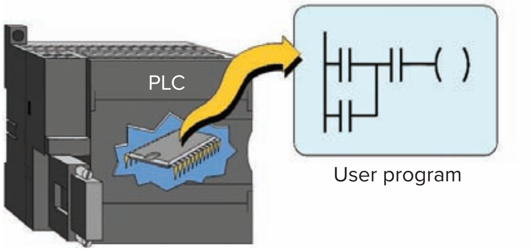


Figure 1-3 All the logic is contained in the PLC's memory.

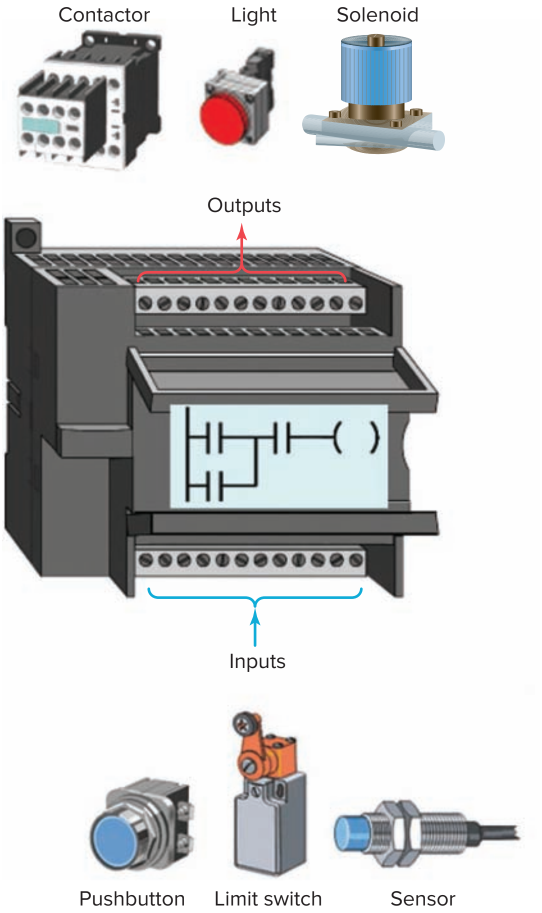


Figure 1-4 Relationships between the inputs and outputs are determined by the user program.

- ***Faster Response Time.*** PLCs are designed for high-speed and real-time applications (Figure 1-6). 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

Figure 1-5 PLC communication module.
Source: Photo courtesy Automation Direct, **www.automationdirect.com**.

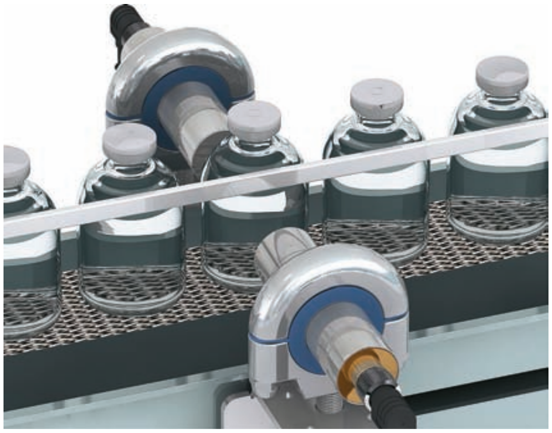


Figure 1-6 High-speed counting.
Source: Courtesy Banner Engineering Corp.

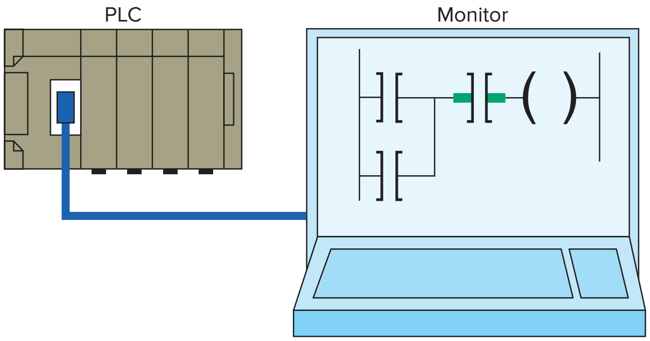


Figure 1-7 Control program can be displayed on a monitor in real time.

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 1-7)·

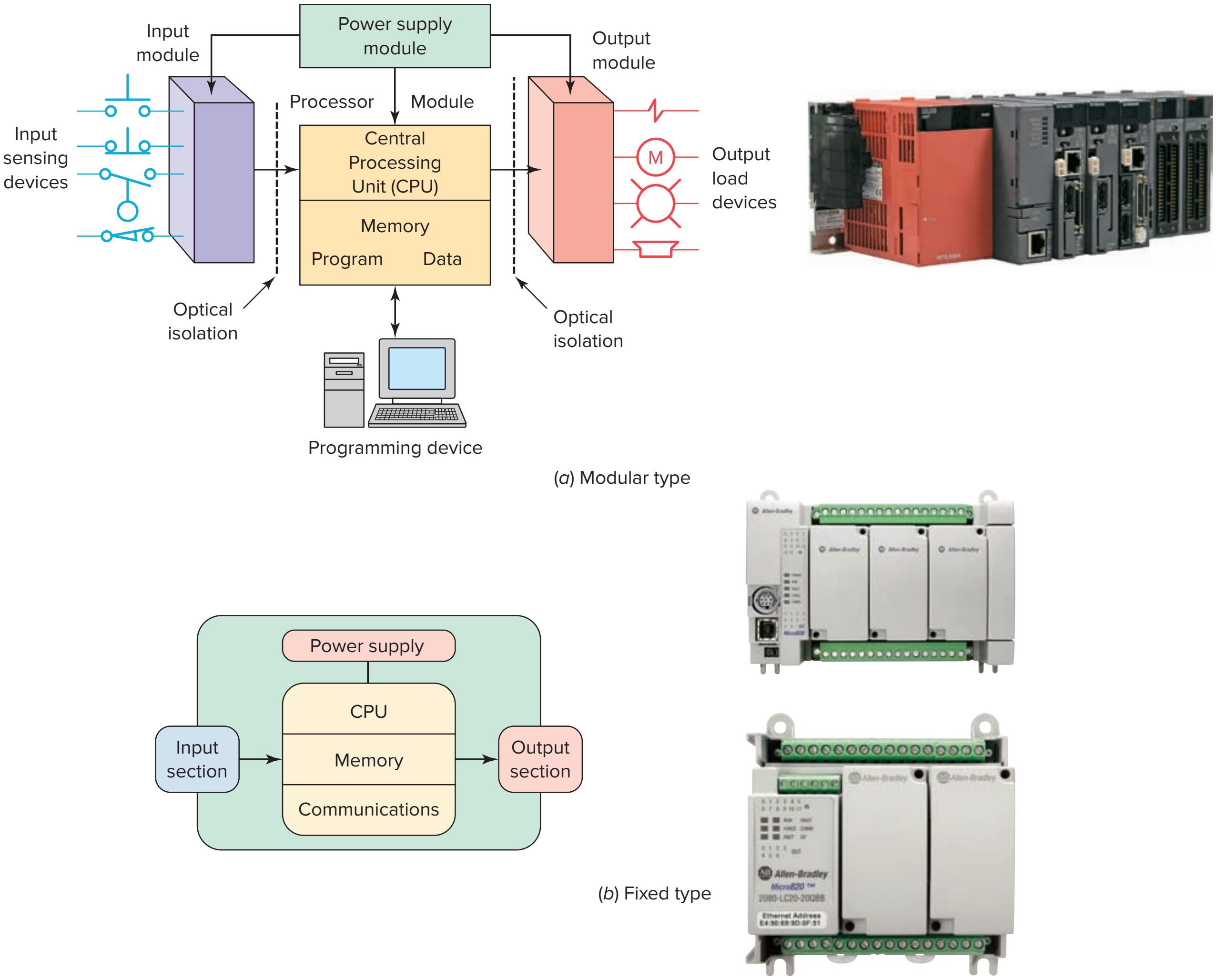
- **Easier to Test Field Devices.** A PLC control panel has the ability to check field devices at a common point. For example, a control system consisting of hundreds of input and output field devices may be contained within a very large manufacturing area. Thus, it would take a considerable amount of 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 1-8. 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 might be slight 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 1-9) 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.

Modular I/O (Figure 1-10) 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



(*a*) Modular type

(*b*) Fixed type

Figure 1-8 Typical parts of a programmable logic controller.
Source: (*a*) Courtesy Mitsubishi Automation; (*b*) Images Courtesy of Rockwell Automation, Inc.

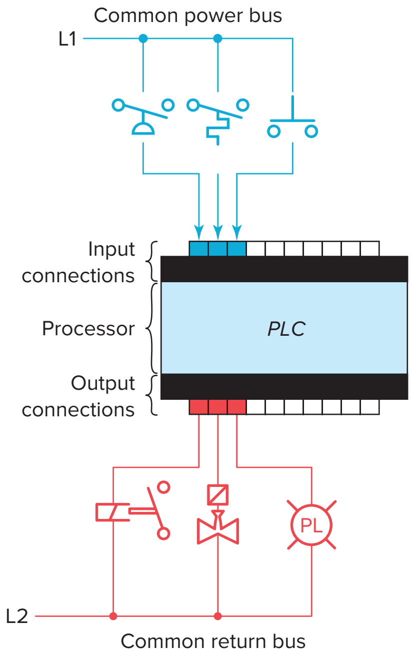


Figure 1-9 Fixed I/O configuration.

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. The PLC processor is also connected to the backplane and can communicate with all the modules in the rack.

The ***power supply*** supplies DC power to other modules that plug into the rack (Figure 1-11). For large PLC systems, this power supply does not normally supply power to the field devices. With larger 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 (Figure 1-12) 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.

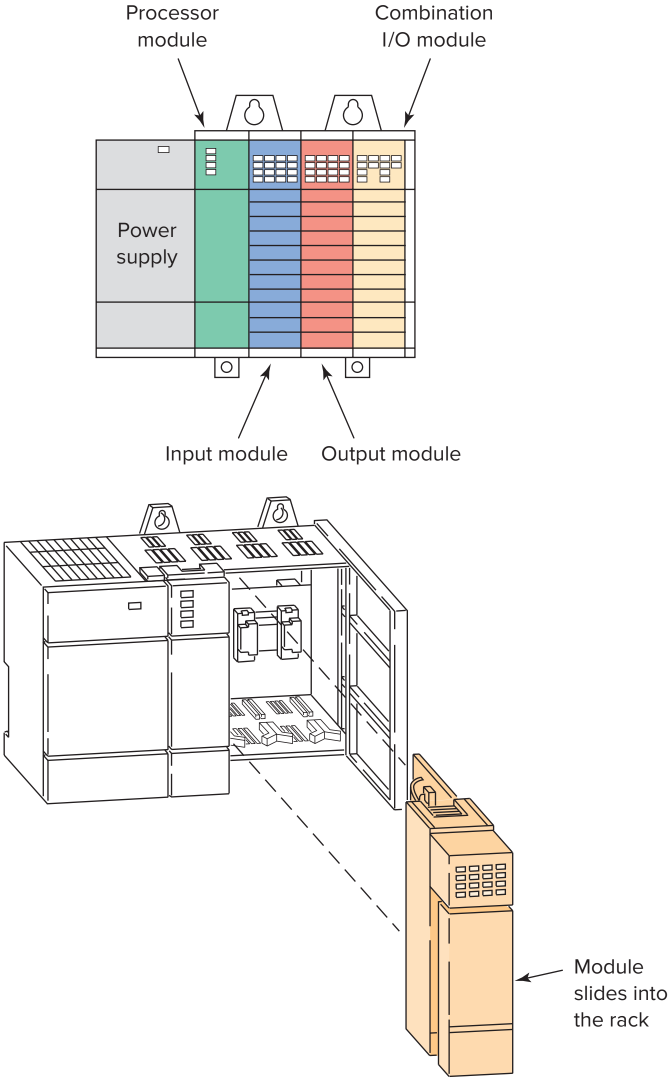


Figure 1-10 Modular I/O configuration.

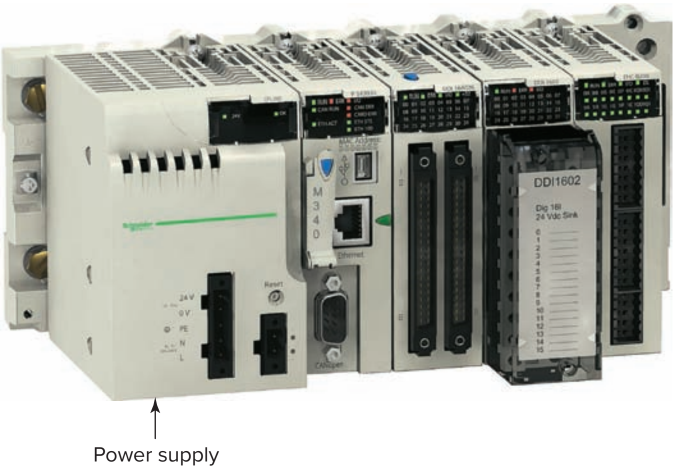


Figure 1-11 The power supply supplies DC power to other modules that plug into the rack.
Source: Photo of PLC Modicon M340 © Schneider Electric, 2010. **www.schneider-electric.com**.

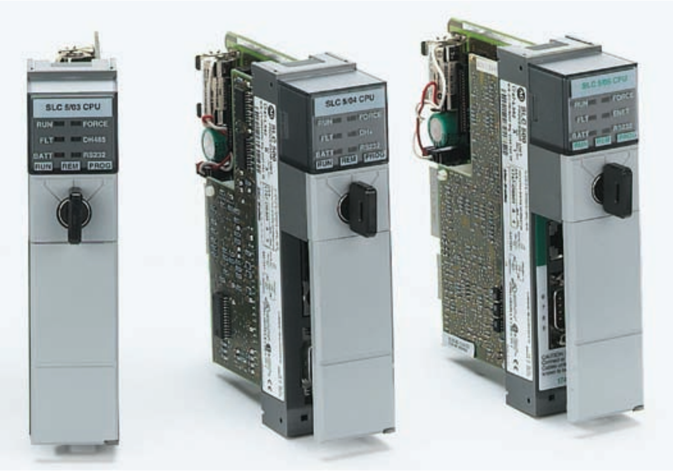


Figure 1-12 Typical PLC processor modules.
Source: Image Courtesy of Rockwell Automation, Inc.

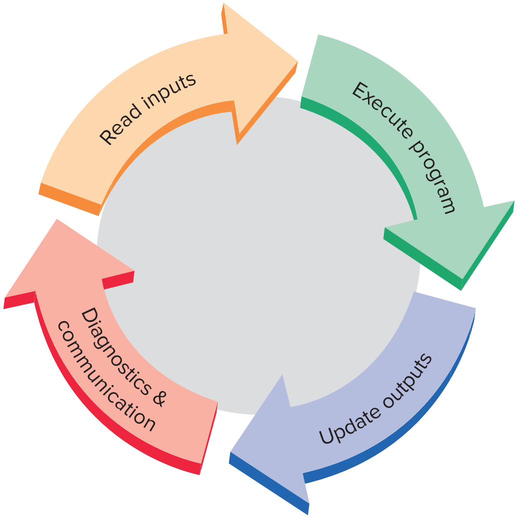


Figure 1-13 Typical PLC scan cycle.

The CPU controls all PLC activity and is designed so that the user can enter the desired program in relay ladder logic. The PLC program is executed as part of a repetitive process referred to as a scan (Figure 1-13). A typical PLC scan starts with the CPU reading the status of inputs. Then, the application program is executed. Once the 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 1-14). 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

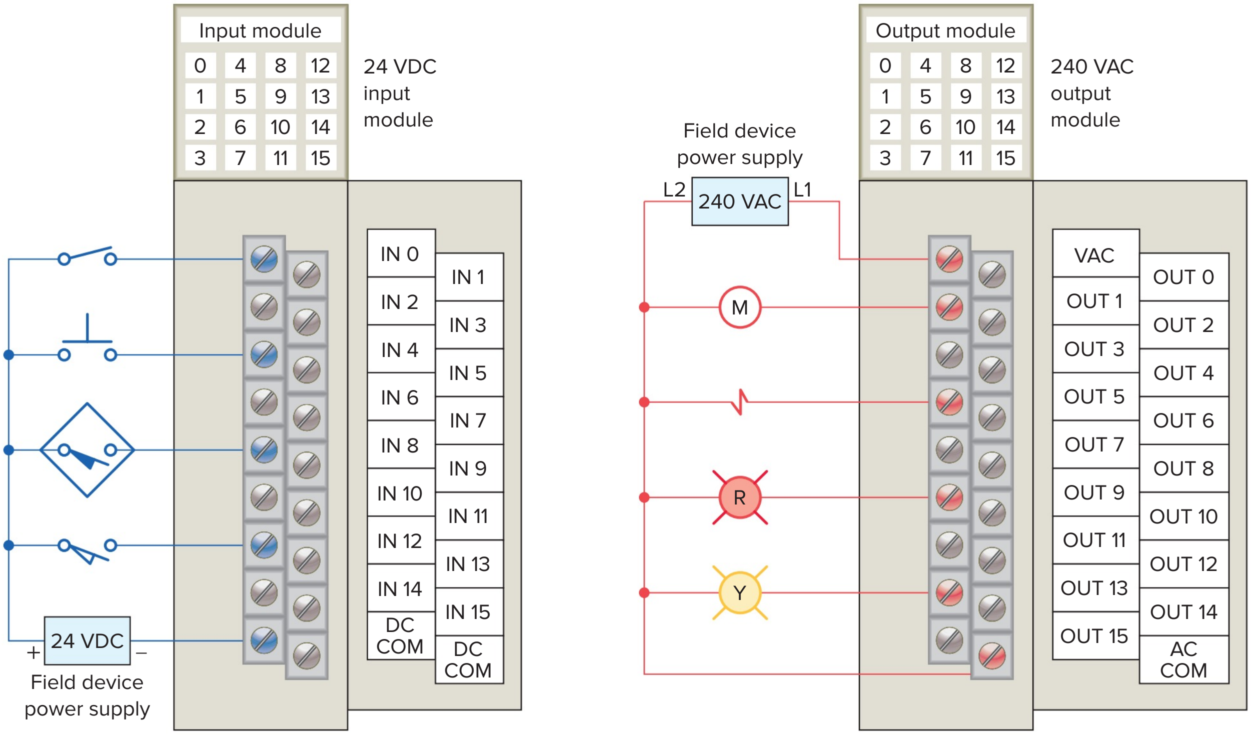


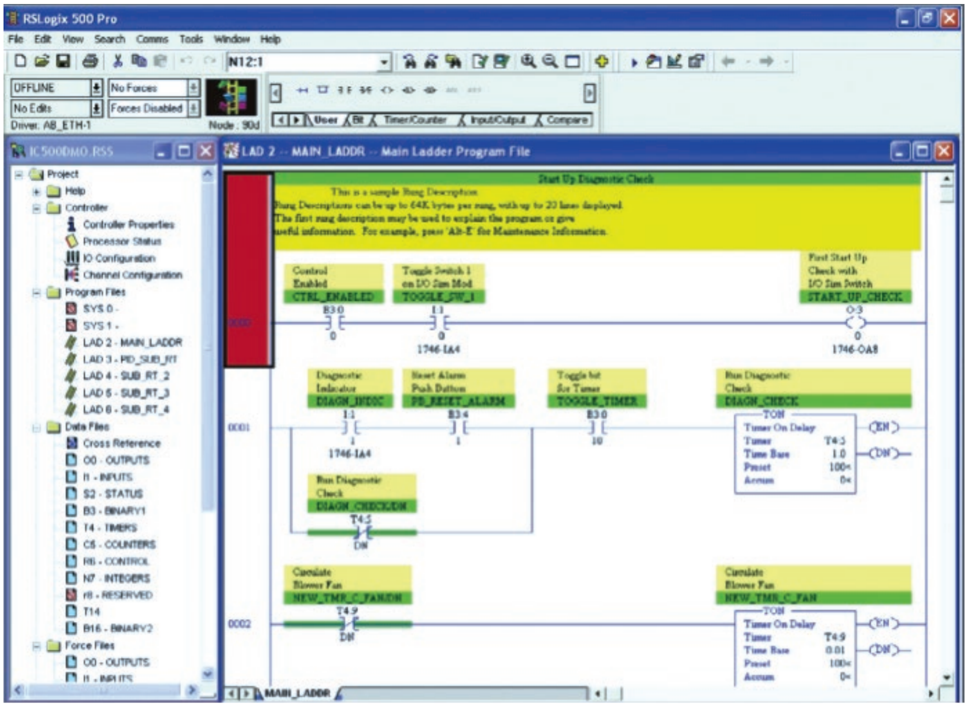
Figure 1-14 Typical PLC input/output (I/O) system connections.

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 duplicates the function of relays, timers, and counters.

A ***programming device*** is used to enter the desired program into the memory of the processor. The program can be entered using relay ladder logic, which is one of the most popular programming languages. Instead of words, ladder logic programming language uses graphic symbols that show their intended outcome. A program in ladder logic is similar to a schematic for a relay control circuit. It is a special language written to make it easy for people familiar with relay logic control to program the PLC. Hand-held programming devices are sometimes used to program small PLCs because they are inexpensive and easy to use. Once plugged into the PLC, they can be used to enter and monitor programs. Both compact hand-held units and laptop computers are frequently used on the factory floor for troubleshooting equipment, modifying programs, and transferring programs to multiple machines.

A personal computer (PC) is the most commonly used programming device. Most brands of PLCs have software available so that a PC can be used as the programming device. This software allows users to create, edit, document, store, and troubleshoot ladder logic programs (Figure 1-15). 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 personal computer communicates with the PLC processor via a serial or parallel data communications link, or Ethernet. If the programming unit is not in use, it may be unplugged and removed. Removing the programming unit will not affect the operation of the user program.

A ***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.



This is a sample Rung Description
Rung Descriptions can be up to 64K bytes per rung, with up to 20 lines displayed.
The first rung description may be used to explain the program or give
useful information. For example, press 'Alt-E' for Maintenance Information.

Figure 1-15 Typical PC software used to create a ladder logic program.
Source: Image Courtesy of Rockwell Automation, Inc.

1.3 Principles of Operation

To get an idea of how a PLC operates, consider the simple process control problem illustrated in Figure 1-16. Here a mixer motor is to be used to automatically stir the liquid in a vat when the temperature and pressure reach 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 1-17. The motor starter coil (M) is energized when both the pressure and temperature switches are closed or when the manual pushbutton is pressed.

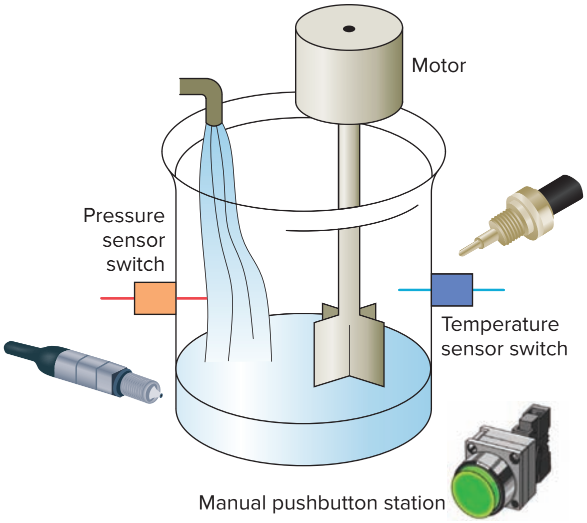


Figure 1-16 Mixer process control problem.

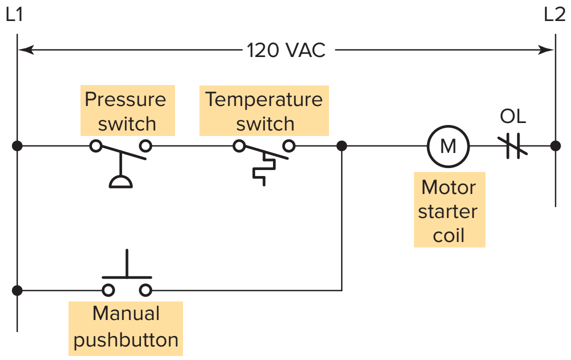


Figure 1-17 Process control relay ladder diagram.

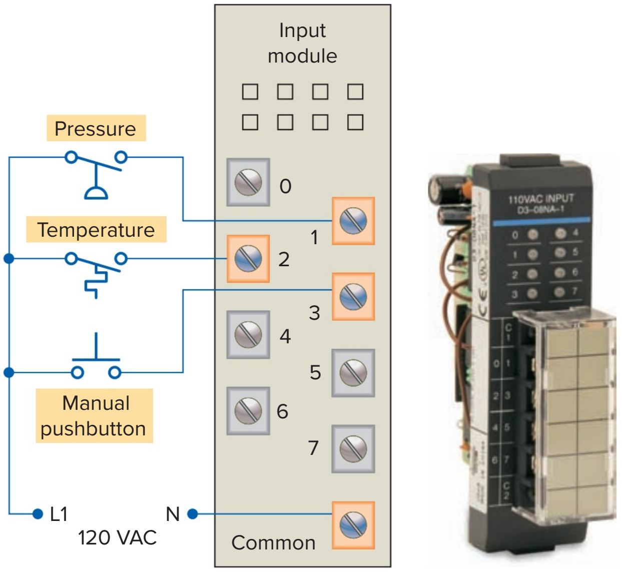


Figure 1-18 Typical wiring connections for a 120 VAC modular configured input module.

Source: Photo courtesy Automation Direct, **www.automationdirect.com**.

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. These devices would be hardwired to an appropriate input module according to the manufacturer's addressing location scheme. Typical wiring connections for a 120 VAC modular configured input module are shown in Figure 1-18.

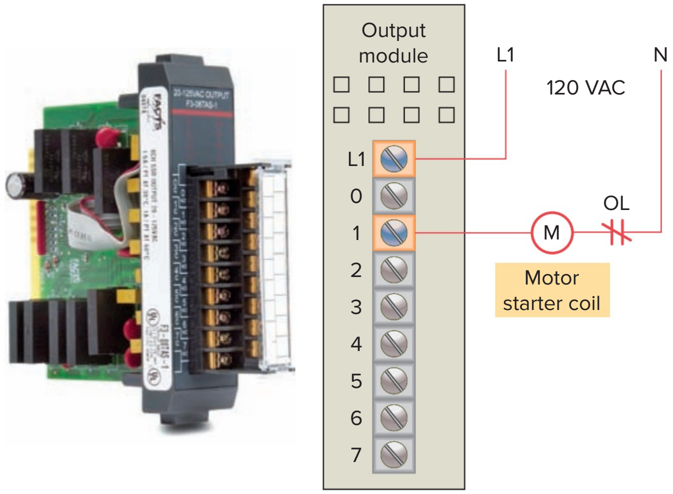


Figure 1-19 Typical wiring connections for a 120 VAC modular configured output module.

Source: Photo courtesy Automation Direct, **www.automationdirect.com**.

The same output field device (motor starter coil) would also be used. This device would be hardwired to an appropriate output module according to the manufacturer's addressing location scheme. Typical wiring connections for a 120 VAC modular configured output module are shown in Figure 1-19.

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 1-20. The format used is similar to the layout of

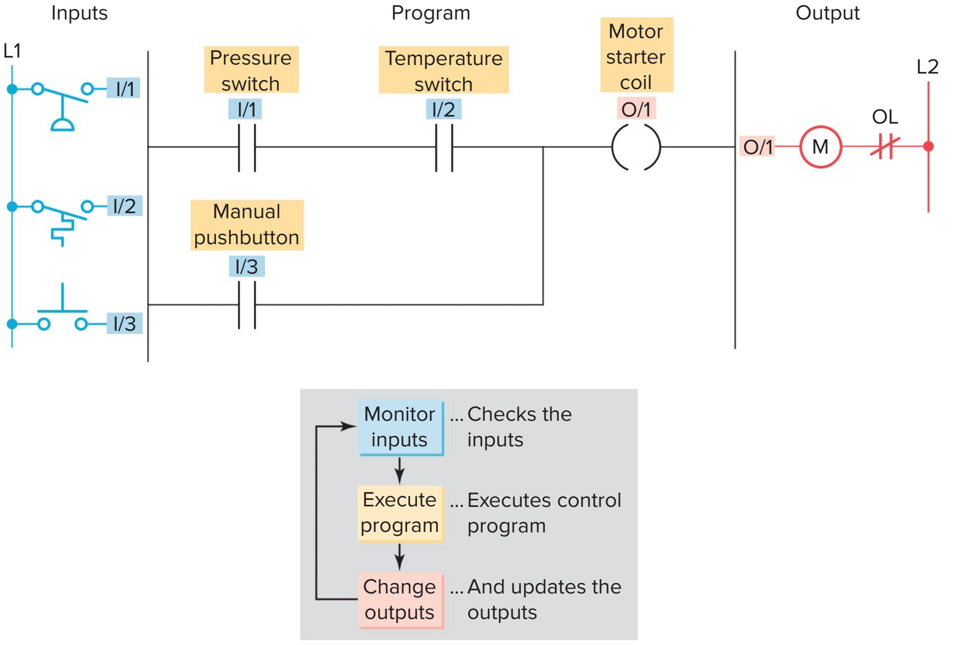


Figure 1-20 Process control PLC ladder logic program with typical addressing scheme.