Allen-Bradley

Classic 1785 PLC-5 Programmable Controllers

(1785-LT, -LT2, -LT3, -LT4)

User Manual

Important User Information

Because of the variety of uses for the products described in this publication, those responsible for the application and use of this control equipment must satisfy themselves that all necessary steps have been taken to assure that each application and use meets all performance and safety requirements, including any applicable laws, regulations, codes, and standards.

The illustrations, charts, sample programs, and layout examples shown in this guide are intended solely for purposes of example. Since there are many variables and requirements associated with any particular installation, Allen-Bradley does not assume responsibility or liability (to include intellectual property liability) for actual use based on the examples shown in this publication.

Allen-Bradley publication SGI-1.1, Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control (available from your local Allen-Bradley office), describes some important differences between solid-state equipment and electromechanical devices that should be taken into consideration when applying products such as those described in this publication.

Throughout this manual we use notes to make you aware of safety considerations:

ATTENTION: Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss.

Attention statements help you to:

- identify a hazard
- avoid the hazard
- recognize the consequences

Important: Identifies information that is critical for successful application and understanding of the product.

Summary of Changes

This manual has been revised to cover only Classic PLC-5 programmable controllers: PLC-5/10, -5/12, -5/15, and -5/25.

It has also been revised to include the accompanying design worksheets that were formerly available as a separate publication: 1785-5.2. This separate publication is no longer available; see Appendix B for these worksheets.

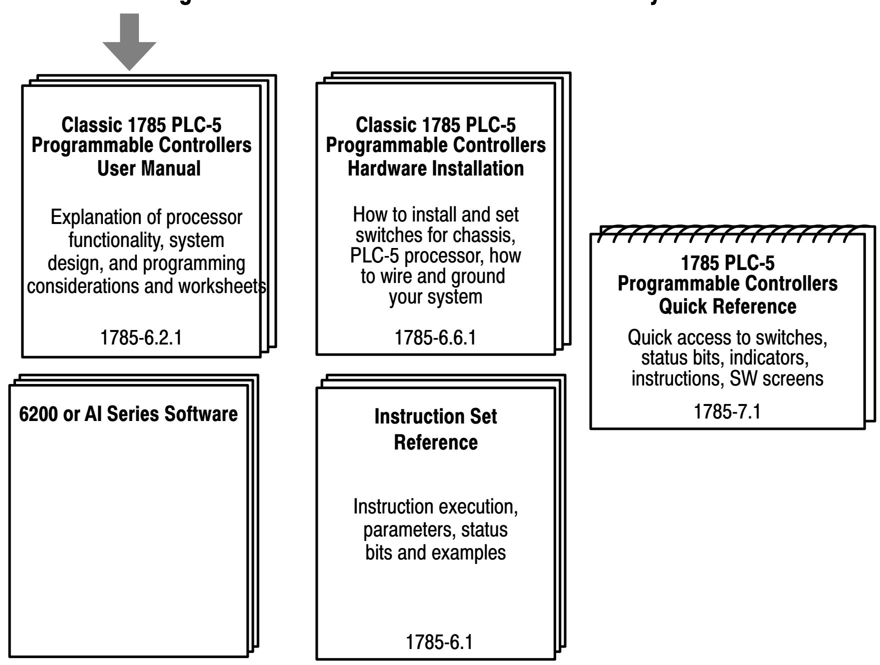
For information about Enhanced and Ethernet PLC-5 processors, see the Enhanced and Ethernet PLC-5 Programmable Controllers User Manual, publication 1785-6.5.12.

Classic PLC-5 Programmable Controllers

How to Use Your Documentation

Your Classic PLC-5 Programmable Controllers documentation is organized into manuals according to the tasks you perform. This organization lets you easily find the information you want without reading through information that is not related to your current task. The arrow in Figure 1 points to the book you are currently using.

Figure 1
Classic PLC-5 Programmable Controllers Documentation Library



For more information on 1785 PLC-5 programmable controllers or the above publications, contact your local Allen-Bradley sales office, distributor, or system integrator.

Purpose of this Manual

This manual is intended to help you design a Classic PLC-5 programmable controller system. Use this manual to assist you in:

- selecting the proper hardware components for your system
- determining the important features of classic PLC-5 processors and how to use those features
- planning your classic PLC-5 system layout

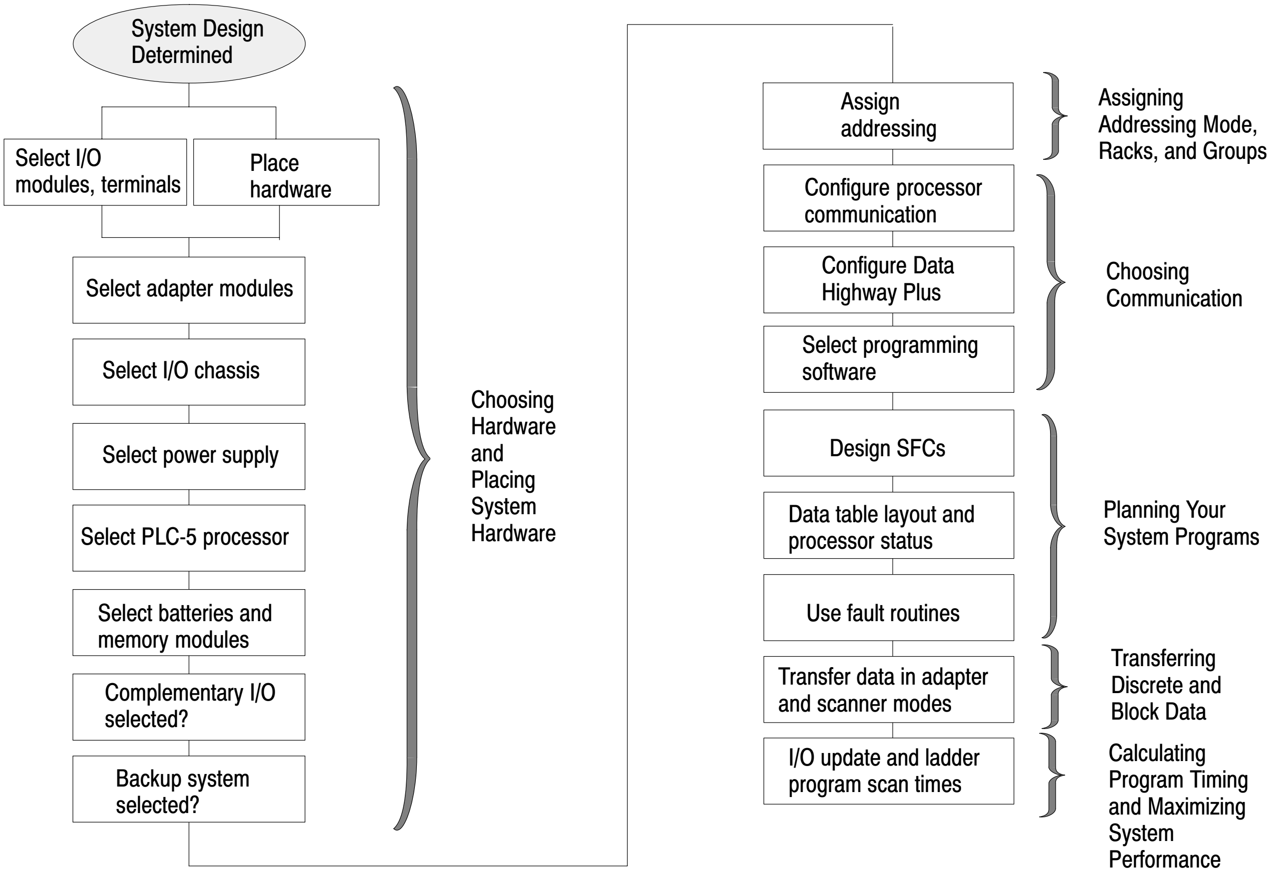
Manual Organization

This manual has ten chapters and two appendices. The following table lists each chapter or appendix with its corresponding title and a brief overview of the topics covered in it.

Chapter / Appendix	Title	Topics Covered
1	Understanding Your System	Provides an overview of Classic PLC-5 processors in different system configurations. Provides an introduction to Classic PLC-5 processors and their primary features and configurations. Also provides information on using a Classic PLC-5 processor as a remote I/O scanner or a remote I/O adapter.
2	Choosing Hardware	Provides information on your hardware choices when you design a Classic PLC-5 processor system.
3	Placing System Hardware	Describes proper environment, Classic PLC-5 processor protection, and prevention of electrostatic damage for your Classic PLC-5 programmable controller system. Also covers raceway and cable layout, backpanel spacing, and grounding configurations.
4	Assigning Addressing Mode, Rack, and Groups	Describes the I/O addressing modes that you can choose for your chassis. Explains how you assign group and rack numbers to your I/O chassis. Also covers how you configure complementary I/O by assigning rack and group addresses.
5	Choosing Communication	Identifies each Classic -5 processor channel/connector, and explains how to configure your Classic PLC-5 processor. Provides additional information about the Data Highway Plus™ (DH+™) link, programming software, and programming-terminal connections.
6	Planning Your System Programs	Explains the use of sequential function charts (SFCs). Provides guidelines and examples for preparing system programs. Provides a map of data table files and methods to address the data table files. Explains how to use the processor status file.
7	Selecting Interrupt Routines	Summarizes the conditions for which you would choose fault routines for your application. Provides a definition of fault routines.
8	Transferring Discrete and Block-Transfer Data	Explains how your CLassic PLC-5 processor transfers discrete and block-transfer data in both scanner and adapter modes.
9	Calculating Program Timing	Provides an overview of processor scan timing. Lists execution times and memory requirements for bit and word instructions as well as file instructions.
10	Maximizing System Performance	Explains how to calculate throughput, and provides methods for optimizing I/O scan time.
A	Selecting Switch Settings	Describes the switch settings for configuring a Classic PLC-5 programmable controller system.
B	Design Worksheets	Provides worksheets to help the designer plan the system and the installer to install the system.

How to Use this Manual

The following flow chart demonstrates a thought process that you can use when you plan your Classic PLC-5 programmable controller system.



Since your decisions cannot always be made as a part of a strictly linear process, you can choose to complete tasks in parallel. When you select your I/O modules, for example, you can also begin to lay out and address your modules. Consult chapter 3, “Placing System Hardware,” to determine environmental requirements, enclosures needed, cable layout, and grounding requirements for your chassis and I/O links. Also, you can choose to assess block-transfer timing when you determine where you will place your block-transfer modules (in the processor-resident local I/O chassis, extended-local I/O chassis, or remote I/O chassis).

Chapter 1

Understanding Your System

Using this Chapter

Understanding the Terms Used in this Chapter

Become familiar with the following terms and their definitions.

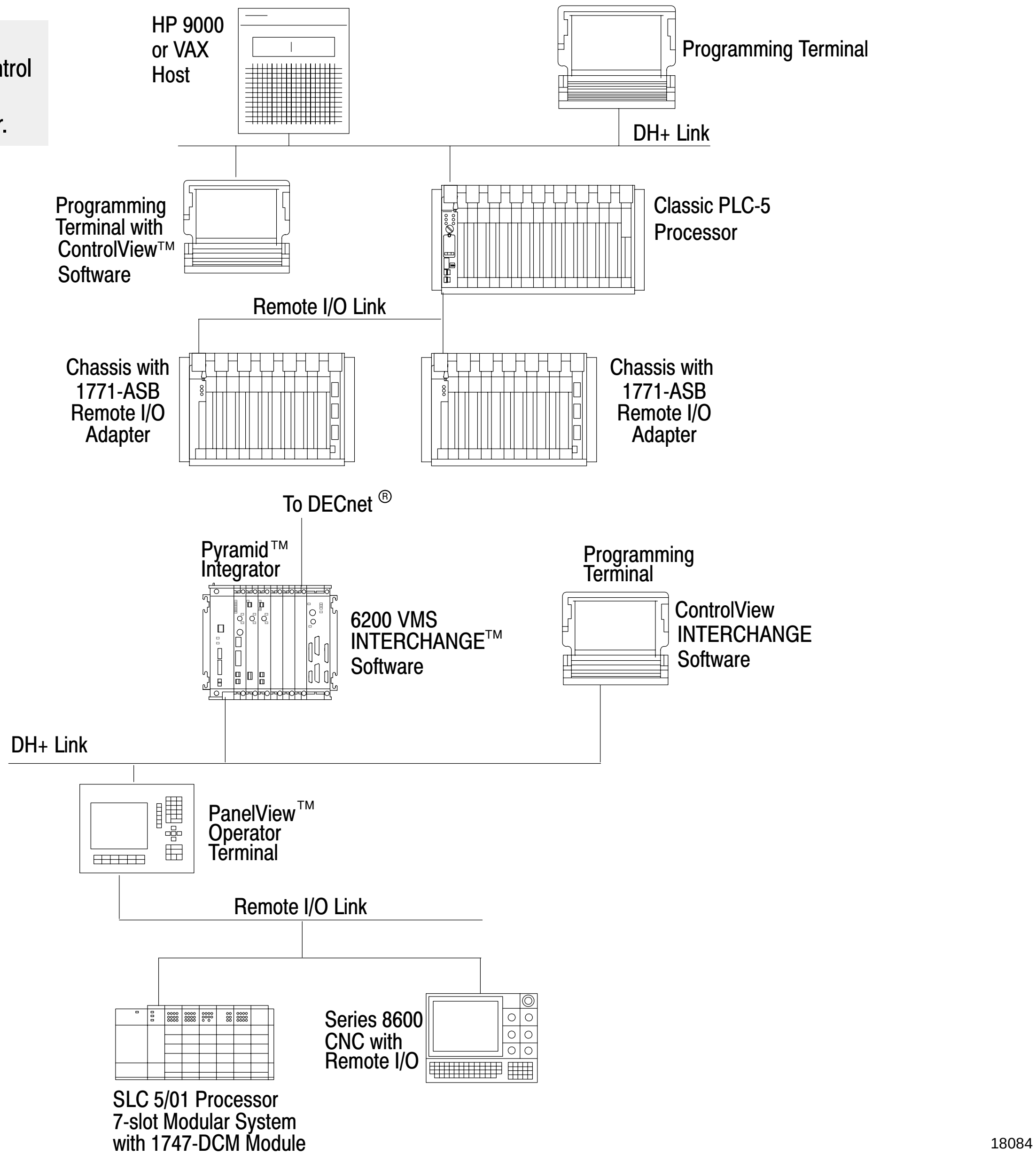
Term	Definition
Processor-resident local I/O chassis	the I/O chassis in which the PLC-5 processor is installed
Processor-resident local I/O	I/O modules located in the same chassis as the PLC-5 processor
Remote I/O link	a serial communication link between a PLC-5 processor port in scanner mode and an adapter as well as I/O modules that are located remotely from the PLC-5 processor
Remote I/O chassis	the hardware enclosure that contains an adapter and I/O modules that are located remotely on a serial communication link to a PLC-5 processor in scanner mode
Discrete-transfer data	data (words) transferred to/from a discrete I/O module
Block-transfer data	data transferred, in blocks of data up to 64 words, to/from a block-transfer I/O module (for example, an analog module)

Designing Systems

You can use Classic PLC-5 processors in a system that is designed for centralized control or in a system that is designed for distributed control.

Centralized control is a hierarchical system where control over an entire process is concentrated in one processor.

Distributed control is a system in which control and management functions are spread throughout a plant. Multiple processors handle the control and management functions and use a Data Highway™ or a bus system for communication.

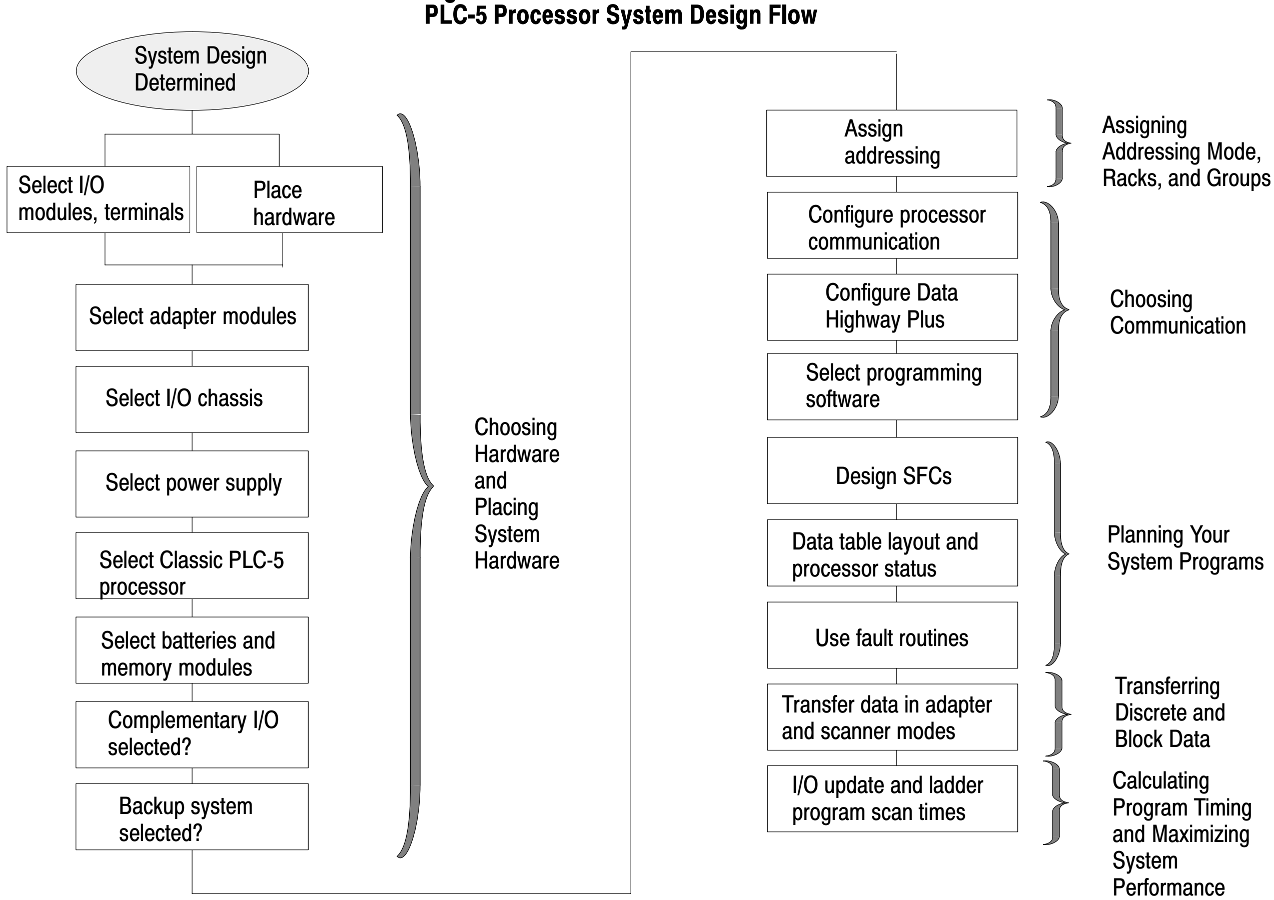


Consider the following items as general guidelines when designing your system.

- Will your processor(s) be used in a centralized or distributed system?
- What type of process(es) will be controlled by the PLC-5 system?
- What processes will be controlled together?
- What are the environmental and safety concerns?
- What is the flow and functionality of your system?

Determine the general criteria for your system. Use the chapters that follow to guide you through the criteria and choices for selecting the major Classic PLC-5 programmable controller system elements, as shown in Figure 1.1.

Figure 1.1
PLC-5 Processor System Design Flow



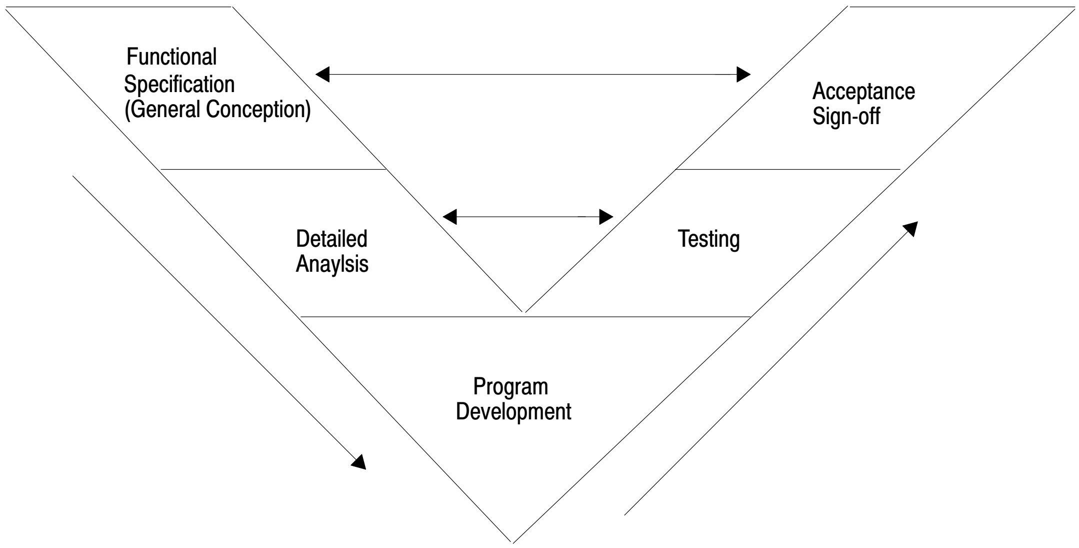
Preparing Your Functional Specification

We recommend that you first develop a specification that defines your hardware selection and your programming application. The specification is a conceptual view of your system. Use it to determine your:

- control strategy
- hardware selection, layout, and addressing
- sequential function chart (SFC)
- special programming features
- ladder-logic requirements

Figure 1.2 illustrates a program-development model that you can use.

Figure 1.2
Program-Development Model



This model allows for the interaction of activities at the different levels. Each section represents an activity that you perform. Prepare a functional specification to start; then, prepare the detailed analysis.

Based on the detailed analysis, you can also develop your programs, enter your programs, and test them. When testing is complete, you are ready to implement the programs in your application. The detailed analysis can be used as the basis for developing your testing procedures and requirements. Because the functional specification is well thought out, it can be used as the program sign-off document.

Functional Specification Content

The functional specification represents a very general view of your process or a description of operation. Identify the events and the overall order in which they must occur. Identify the equipment that you will need for your process/operation. Generally indicate the layout of your system. If your application requires a distributed control system, for example, indicate where you will need remote I/O links. Also, you can have a process that is located close to your processor. The process can require faster update time than that provided by a remote I/O link, so you can select an extended-local I/O link for that process.

Important: Choose a communication rate for your remote I/O link at which every device on the link can communicate.

The program-development portion of your functional specification can be in any form: written statement; flowchart; or rough-draft MCPs, SFCs, and subroutines. Use the form that is most familiar to you. We recommend, however, that you generate rough-draft SFCs and subroutines so that you have a better correspondence between your beginning diagrams and your finished program.

Detailed Analysis

In this phase, you identify the logic needed to plan your programs. This includes inputs, outputs, specific actions, and transitions between actions (i.e., the bit-level details needed to write your program).

Program Development

You enter the programs either offline into your computer or online into a processor. In the next phase, you test the programs that you have entered. Once testing is complete, your resulting programs should match your functional specification.

Checking for Completeness

When you complete the functional specification and the detailed analysis, review them and check for missing or incomplete information such as:

- input conditions
- safety conditions
- startup or emergency shutdown routines
- alarms and alarm handling
- fault detection and fault handling
- message display of fault conditions
- abnormal operating conditions

Introducing Classic PLC-5 Processor Modules

The following is a list of the PLC-5 processors and their catalog numbers.

Processor	Catalog Number
PLC-5/10™	1785-LT4
PLC-5/12™	1785-LT3
PLC-5/15™	1785-LT
PLC-5/25™	1785-LT2

For information on other PLC-5 processors (Enhanced, Ethernet, or ControlNet), see your Allen-Bradley representative.

Classic PLC-5 Family Processor Features

From the family of PLC-5 processors, you can choose the processor(s) that you need for your application. Features common to all Classic PLC-5 processors are:

- same physical dimensions
- use of the left-most slot in the 1771 I/O chassis
- can use any 1771 I/O module in the processor-resident local I/O chassis with up to 32 points per module
- same programming software and programming terminals
- same base set of instructions
- ladder programs and SFCs can be used by any of the PLC-5 processors

Check with your Allen-Bradley sales office or distributor if you have questions regarding any of the features of your PLC-5 processor.

Subprogram Calls
Use a subroutine to store recurring sections of program logic that can be accessed from multiple program files. A subroutine saves memory because you program repetitive logic only once. The JSR instruction directs the processor to go to a separate subroutine file within the logic processor, scan that subroutine file once, and return to the point of departure.

For detailed information about how you generate and use subroutines, see your programming software documentation set.

Sequential Function Charts
Use SFCs as a sequence-control language to control and display the state of a control process. Instead of one long ladder program for your application, divide the logic into steps and transitions. A step corresponds to a control task; a transition corresponds to a condition that must occur before the programmable controller can perform the next control task. The display of these steps and transitions lets you see what state the machine process is in at a given time.

For detailed information about how you generate and use SFCs, see you programming software.

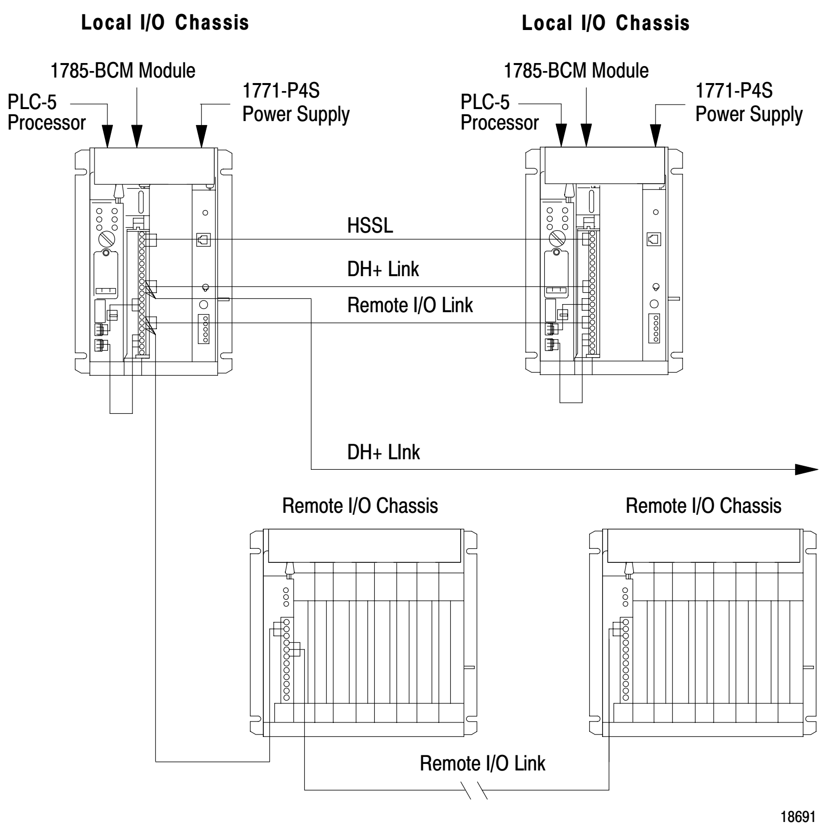
Ladder Logic Programs
A main program file can be an SFC file numbered 1-999; it can also be a ladder-logic file program numbered 2-999 in any program file.

Consider using this technique:	If you are:
SFC	• defining the order of events in a sequential process
Ladder Logic	• more familiar with ladder logic than with programming languages such as BASIC • performing diagnostics • programming discrete control

For detailed information about how you use ladder logic, see your programming software documentation.

Backup System

The following diagram shows a typical PLC-5 backup system:



In a PLC-5 backup system configuration, one system controls the operation of remote I/O and DH+ communications. This system is referred to as the “primary system.” The other system is ready to take control of the remote I/O and DH+ communications in the event of a fault in the primary system. This is referred to as the “secondary system.”

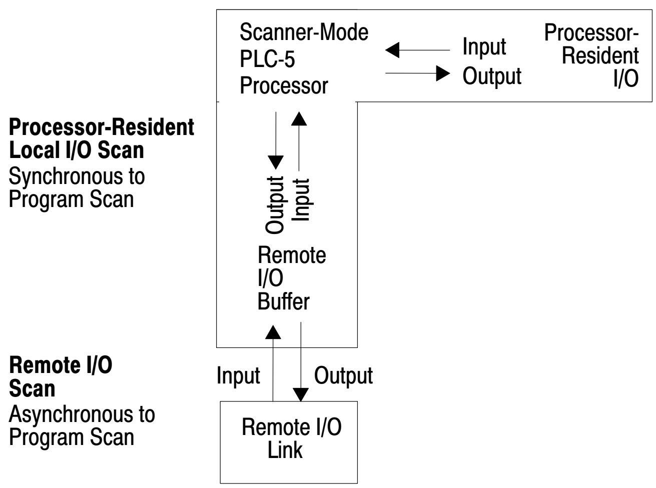
See chapter 2, “Choosing Hardware,” to select backup system hardware. See the PLC-5 Backup Communication Module User Manual, publication 1785-6.5.4, for more information on configuring a PLC-5 backup system.

Using the Classic PLC-5 Processor as a Remote I/O Scanner

Use scanner mode whenever you want a Classic PLC-5 processor to scan and control remote I/O link(s). The scanner-mode processor also acts as a supervisory processor for other processors that are in adapter mode.

The scanner-mode processor scans the processor memory file to read inputs and control outputs. The scanner-mode processor transfers discrete-transfer data and block-transfer data to/from the processor-resident local rack as well as to/from modules in remote I/O racks.

A PLC-5 processor scans processor-resident local I/O synchronously to the program scan. A PLC-5 processor scans remote I/O asynchronously to the program scan, but the processor updates the input/output image data table from the remote I/O buffer(s) synchronously to the program scan. This occurs at the end of each program scan.



The scanner-mode PLC-5 processor can also:

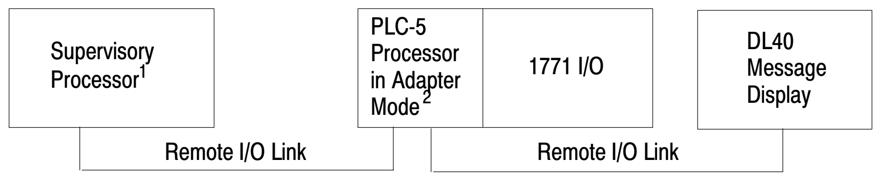
- gather data from node adapter devices in remote I/O racks
- process I/O data from 8-, 16-, or 32-point I/O modules
- address I/O in 2-, 1-, or 1/2-slot I/O groups
- support a complementary I/O configuration
- support block transfer in any I/O chassis

Configure the PLC-5/15 or -5/25 processor for scanner mode by setting switch assembly SW1.

Using the Classic PLC-5 Processor as a Remote I/O Adapter

Use a Classic PLC-5 processor (except the PLC-5/10 processor) in adapter mode when you need predictable, real-time exchange of data between a distributed control PLC-5 processor and a supervisory processor. You connect the processors via the remote I/O link (see Figure 1.3). You can monitor status between the supervisory processor and the adapter-mode PLC-5 processor at a consistent rate (i.e., the transmission rate of the remote I/O link is unaffected by programming terminals and other non-control-related communications).

Figure 1.3
Adapter-Mode Communication



[1] The following programmable controllers can operate as supervisory processors:

- PLC-2/20™ and PLC-2/30™ processors
- PLC-3™ and PLC-3/10™ processors
- PLC-5/11, -5/15, -5/20, -5/25, and -5/30 processors as well as PLC-5/VME™ processors
- PLC-5/40, -5/40L, -5/60, -5/60L, and -5/80 processors as well as PLC-5/40BV™ and PLC-5/40LV™ processors
- PLC-5/20E™, -5/40E™
- PLC-5/250™

[2] All PLC-5 family processors, **except the PLC-5/10**, can operate as remote I/O adapter modules.

The PLC-5 processor in adapter mode acts as a remote station to the supervisory processor. The adapter-mode PLC-5 processor can monitor and control its processor-resident local I/O while communicating with the supervisory processor via a remote I/O link.

The supervisory processor communicates with the PLC-5/12, -5/15, or -5/25 adapter with either eight or four I/O image table words.

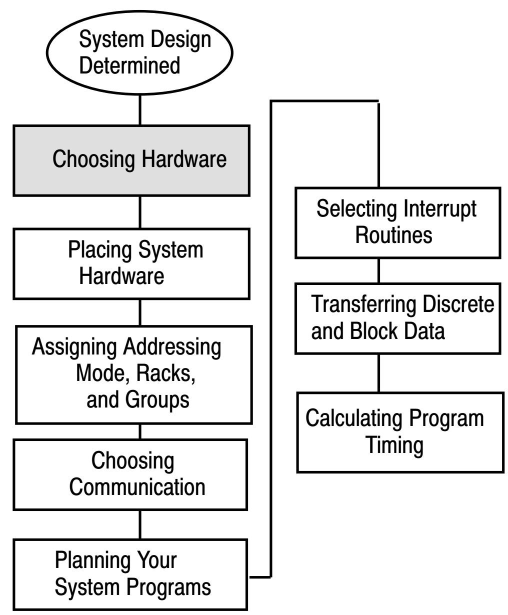
A PLC-5 processor transfers I/O data and status data using discrete transfers and block transfers. You can also use block-transfer instructions to communicate information between a supervisory processor and an adapter-mode processor. The maximum capacity per block transfer is 64 words.

Chapter 2

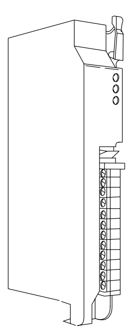
Choosing Hardware

Chapter Objectives

Use this chapter to guide you in the selection of system hardware for your application.



Selecting I/O Modules



You select I/O modules to interface your PLC-5 processor with machines or processes that you have previously determined.

Use the following list and Table 2.A as guidelines for selecting I/O modules and/or operator control interface(s).

- How much I/O is required to control the process(es)?
- Where will you concentrate I/O points for portions of an entire process (when an entire process is distributed over a large physical area)?
- What type of I/O is required to control the process(es)?
- What is the required voltage range for each I/O module?
- What is the backplane current required for each I/O module?
- What are the noise and distance limitations for each I/O module?
- What isolation is required for each I/O module?

Table 2.A
Guidelines for Selecting I/O Modules

Choose this type of I/O module:	For these types of field devices or operations (examples):	Explanation:
Discrete input module and block I/O module[1]	Selector switches, pushbuttons, photoelectric eyes, limit switches, circuit breakers, proximity switches, level switches, motor starter contacts, relay contacts, thumbwheel switches	Input modules sense ON/OFF or OPENED/ CLOSED signals. Discrete signals can be either ac or dc.
Discrete output module and block I/O module[1]	Alarms, control relays, fans, lights, horns, valves, motor starters, solenoids	Output module signals interface with ON/OFF or OPENED/CLOSED devices. Discrete signals can be either ac or dc.
Analog input module	Temperature transducers, pressure transducers, load cell transducers, humidity transducers, flow transducers, potentiometers	Convert continuous analog signals into input values for PLC processor.
Analog output module	Analog valves, actuators, chart recorders, electric motor drives, analog meters	Interpret PLC processor output to analog signals (generally through transducers) for field devices.
Specialty I/O modules	Encoders, flow meters, I/O communication, ASCII, RF type devices, weigh scales, bar-code readers, tag readers, display devices	Are generally used for specific applications such as position control, PID, and external device communication.

1 A 1791 block I/O module is a remote I/O device that has a power supply, remote I/O adapter, signal conditioning circuitry, and I/O connections. A block I/O module does not require a chassis mount. It is used to control concentrated discrete remote I/O such as control panels, pilot lights, and status indications.

Important: Determine addressing in conjunction with I/O module selection. The selection of addressing and the selection of I/O module density are mutually dependent.

Selecting I/O Module Density

The density of an I/O module is the number of processor input or output image table bits to which it corresponds. A bidirectional module with 8 input bits and 8 output bits has a density of 8. Table 2.B provides guidelines for selecting I/O module density.

Table 2.B
Guidelines for Selecting I/O Module Density

Choose this I/O density:	If you:
8-point I/O module	• currently use 8-point modules • need integral, separately-fused outputs • want to minimize cost per module
16-point I/O module	• currently use 16-point modules • need separately fused outputs with a special wiring arm
32-point I/O module	• currently use 32-point modules • want to minimize number of modules • want to minimize the space required for I/O chassis • want to minimize cost per I/O point

Master/Expander I/O Modules

Some I/O modules (called “masters”) communicate with their expanders over the backplane. These master/expander combinations either:

- **can** time-share the backplane, or
- **cannot** time-share the backplane

For masters that **can** time-share the backplane, you can use two masters in the same chassis. For a master/expander combination that **cannot** time-share the backplane, you cannot put another master/expander combination in the same I/O chassis.

Example: The stepper-controller module (cat. no. 1771-M1, part of a 1771-QA assembly) and the servo-controller module (cat. no. 1771-M3, part of a 1771-QC assembly) always act as masters and cannot time-share the backplane. Therefore, you cannot put a second master module in the same chassis with either of these modules.

Table 2.C summarizes the compatibility of master modules within a single I/O chassis.

Table 2.C
Compatibility of Master Modules within a Single I/O Chassis

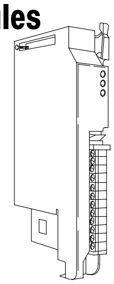
1st Master Module	2nd Master Module 1771-IX[1]	1771-IF[1]	1771-OF[1]	1771-M1	1771-M3
1771-IX[1]		Valid[2]	Valid[2]		
1771-IF[1]	Valid[2]	Valid[2]	Valid[2]		
1771-OF[1]	Valid[2]	Valid[2]	Valid[2]		
1771-M1					
1771-M3					

1 These modules have been superseded by 1771-IXE, -IFE, and OFE master modules that do not exhibit the master/expander conflict in a chassis as 1771-IX, -IF, and OF master modules shown in this table.

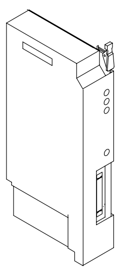
2 These are the only master combinations that you can use in a single I/O chassis. These combinations are valid with or without the module’s associated expanders (1771-M1 and -M3 have expander modules). You can use a maximum of two masters in the same chassis; you can use any other intelligent I/O modules not shown here with these masters.

Important: Density is not relevant to an expander module because it communicates only with its master; an expander module does not communicate directly with an adapter.

Selecting I/O Adapter Modules



ASB



ALX

Select I/O adapter modules to interface your PLC-5 processor with I/O modules. Use Table 2.D as a guide when you select I/O adapter modules.

Table 2.D
Guidelines for Selecting Adapter Modules

Choose:	When your requirements are:
1771-AS or 1771-ASB[1] Remote I/O Adapter Module (or 1771-AM1, -AM2 chassis with integral power supply and adapter module)	a remote I/O link with: • 57.6 kbps with a distance of up to 10,000 cable feet **or** • timing that isn't critical enough to place I/O modules in a processor local I/O chassis or an extended-local I/O chassis
1771-ALX Extended-Local I/O Adapter Module	an extended-local I/O link with timing that is critical and all extended-local I/O chassis are located within 100 ft of the processor.

[1] 1771-ASB series C and later have 230.4 kbps communication rate in addition to 57.6 kbps and 115.2 kbps.

1771-AS/ASB Remote I/O Adapter Modules

Table 2.E shows the I/O density per module and addressing modes you can use with I/O chassis and remote I/O adapter modules.

Table 2.E
I/O Chassis/Adapter Module Combinations

Remote I/O Adapter Module Cat. No.	I/O Density per Module	Addressing		
		2-Slot	1-Slot	1/2-Slot
1771-AS	8 16 32	Yes --[1] No	No No No	No No No
1771-ASB Series A	8 16 32	Yes --[1] No	Yes Yes --[1]	No No No
1771-ASB Series B, C, and D	8 16 32	Yes --[1] No	Yes Yes --[1]	Yes Yes Yes
1771-AM2	8 16 32	-- -- --	Yes Yes --[1]	Yes Yes Yes

[1] Conditional module placement; you must use an input module and an output module in two adjacent slots (even/odd pair) of the I/O chassis beginning with slot 0. If you cannot pair the modules this way, leave the adjacent slot empty.

Using the 1771-ASB Series C or D adapter module, you can choose one of three communication rates: 57.6 kbps, 115.2 kbps, or 230.4 kbps.

1771-ALX Extended-Local I/O Adapter Module

Table 2.F shows the I/O density per module and addressing modes you can use with I/O chassis and extended-local I/O adapter modules.

Table 2.F
I/O Chassis/Extended -Local I/O Adapter Module Combinations

Module Cat. No.	I/O Density per Module	Addressing		
		2-Slot	1-Slot	1/2-Slot
1771-ALX Series A	8 16 32	Yes --[1] No	Yes Yes --[1]	Yes Yes Yes

1 Conditional module placement; you must use an input module and an output module in two adjacent slots (even/odd pair) of the I/O chassis beginning with slot 0. If you cannot pair the modules this way, leave the adjacent slot empty.

Other Devices on an I/O Link

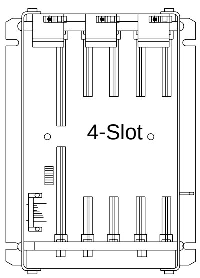
Other devices that you can use on a remote I/O link are:

- PLC-5 processor in adapter mode
- PLC-5/250 remote scanner in adapter mode
- PLC interface module for digital ac and dc drives
- remote I/O adapter for Bulletin 1336 drives
- RediPANEL™ pushbutton and keypad modules
- Dataliner™
- PanelView (see operator interface)
- F30D option module (for T30 plant-floor terminal)
- 8600 or 9/SERIES CNC with remote I/O adapter option
- CVIM™ in adapter mode
- Pro-Spec™ 6000 Fastening System with remote I/O adapter option
- 1747-DCM module (to SLC-500 rack)
- 1771-DCM module
- 1771-GMF robot (remote I/O interface module)

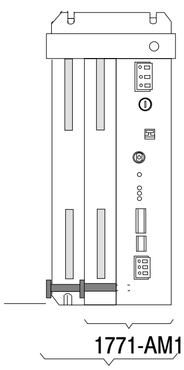
See the appropriate Allen-Bradley product catalog for more information on these devices.

Selecting I/O Chassis

An I/O chassis is a single, compact enclosure for the processor, power-supply modules, remote and extended-local I/O adapter modules, and I/O modules. The left-most slot of the I/O chassis is reserved for the processor or adapter module. Consider the following when selecting a chassis:



1771-A1B



- When you determine the maximum number of I/O in your application, allow space for the I/O slots dedicated to power-supply modules, communication modules, and other intelligent I/O modules.
- You must use series B or later chassis with 16- and 32-point I/O modules.
- Allow space for future addition of I/O modules to chassis.

I/O chassis available are:

- 4-slot (1771-A1B)
- 8-slot (1771-A2B)
- 12-slot—rack mount (1771-A3B), panel mount (1771-A3B1)
- 16-slot (1771-A4B)

You can also choose a chassis with an integral power supply and remote I/O adapter (show at left). The two types are:

- 1-slot (1771-AM1)
- 2-slot (1771-AM2)

Selecting an Operator Interface

PanelView and ControlView are operator interface products or packages that communicate with a PLC-5 processor. Use Table 2.G as a guideline when selecting either PanelView or ControlView for your PLC-5 programmable controller system. Use Table 2.H for a comparison of PanelView and ControlView features.

Table 2.G
Guidelines for Selecting an Operator Interface

Choose this operator interface:	For these types of operations (examples):	Explanation:
PanelView[1]	Starts/stops, auto/manual operations, setpoints, outputs, alarms	Used as an operator window to enter commands that make process adjustments such as starts/stops and loop changes. Can also be used for alarming operations. Can communicate with a single PLC-5 processor on a remote I/O link. Has a fixed number of devices and amount of data that it can handle. Has built-in error checking. Is an industrial-hardened CRT with pushbuttons, solid state memory and processor, and no moving parts (i.e., disk drive). Utilizes pass through, which is the ability to download/upload via DH+/remote I/O links.
ControlView[1]	Store, display, and manipulate data on process performance (i.e., trends, process graphics, formulas, reports, and journals)	Used as an operator window that communicates with a PLC-5 processor on Data Highway Plus (DH+) link. Designed for use as an information link. Can communicate to multiple PLC processors. ControlView is a software package that runs on an IBM® DOS-based personal computer.

[1] Refer to your local Allen-Bradley sales office or Allen-Bradley distributor for more information on PanelView and ControlView.

Table 2.H
Comparison of PanelView and ControlView Features

Category	PanelView	ControlView
Communication with PLC processor	Remote I/O 5 block transfers per terminal maximum (32 words per transfer) 1 discrete transfer per terminal (64 words maximum, one way) This is 8 racks of transfer	DH+ link Data Highway Data Highway II Native Mode
Graphics	Character graphics Create screens with PanelBuilder software Monochrome or color (8 of 16 colors displayed at a time)	Pixel Graphics Create screens with Mouse Grafix editor option or C Toolkit EGA, VGA, or equivalent with 256K RAM Monochrome or color monitor
Number of Screens per Terminal/Workstation	8 to 12 screens of medium complexity typical 200 objects maximum per screen Limited by terminal memory size: 128 Kbytes	Limited only by hard disk capacity 50 data entry locations per screen 50 tags per command list per screen 300 tags/points maximum per screen
Data Capacity	200 objects maximum per screen	10,000 points maximum in database
Communication Rate	Limited by block-transfer and discrete-transfer timing Depends on PLC processor and remote I/O link size	8 scan classes, each with user-configurable foreground and background update times; limited by performance of Data Highway, DH+, or Data Highway II link
Hardware	Keypad or Touchscreen terminals, color or monochrome Allen-Bradley, IBM, or compatible computer required for PanelBuilder software	A-B, IBM, or compatible computer with 286 or 386 processor, math coprocessor, hard disk required at each operator station
Programming	PanelBuilder software Menu-driven with fill-in-the-blank information entry Use PanelBuilder to create application file that defines screens, messages, alarms, then download application file to PanelView terminal	Create data base online via the menu. Menu-driven, fill-in-the-blank information entry, or import data via the ASCII import capability Create screens with the mouse GRAFIX editor option or C toolkit option
Messages	496 maximum per terminal	Not Applicable
Alarms	496 maximum per terminal	2000 points with Alarming option
Security	8 levels	16 levels with individual operator login capability Individual objects with security Screen lockout
Options	Remote serial port EEPROM or EPROM memory	Lots of software options

For more information on selecting and configuring PanelView, see:

- PanelView Operator Terminal and PanelBuilder Development Software User Manual, cat. no. 2711-ND002 version C, PN40061-139-01—request latest revision
- Replacing Node Adapter Firmware for PanelView Terminals Installation Data, PN40062-236-01—request latest revision

For more information on selecting and configuring ControlView, see:

- ControlView Core User Manual, publication 6190-6.5.1
- ControlView Allen-Bradley Drivers User Manual, publication 6190-6.5.5
- ControlView Networking User Manual, publication 6190-6.5.9

Other Operator Interfaces

You can use the following as operator interfaces in your PLC-5 processor system:

- RediPANEL pushbutton and keypad modules
- Dataliner
- 1784-T47 and 1784-T53 programming terminals

See the appropriate Allen-Bradley product catalog for more information on these operator interfaces.

Choosing a Classic PLC-5 Processor for Your Application

Choose from the following PLC-5 processors.

Table 2.I
Classic PLC-5 Processor Selection Chart—Part 1

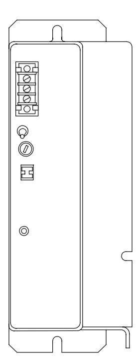
Processor/ Cat. No.	Maximum User Memory Words	EEPROM Module Memory (Words) & Module Number	Total I/O Maximum (any mix)	Analog I/O Max	Program Scan Time / K Word	I/O Scan time/Rack (in a single Chassis, ext-local or remote)	Multiple MCPs / Quantity
PLC-5/10 (1785-LT4)	6 K	8 K (1785-MJ)	• 512 (32-I/O modules) • 256 (16-I/O modules) • 128 (8-I/O modules)	256	2 ms (discrete logic) 8 ms (typical)	N/A	No / 1
PLC-5/12 (1785-LT3)	6 K	8 K (1785-MJ)	• 512 (32-I/O modules) • 256 (16-I/O modules) • 128 (8-I/O modules)	256	2 ms (discrete logic) 8 ms (typical)	• 10 ms @ 57.6 kbps (remote)	No / 1
PLC-5/15 (1785-LT)	6 K expandable to 10 K or 14 K	8 K (1785-MJ)	• 512 (any mix) or • 512 in + 512 out (complementary)	512	2 ms (discrete logic) 8 ms (typical)	• 10 ms @ 57.6 kbps (remote)	No / 1
PLC-5/25 (1785-LT2)	13 K expandable to 17 K or 21 K	8 K (1785-MJ) or 16 K (1785-MK)	• 1024 (any mix) or • 1024 in + 1024 out (complementary)	1024	2 ms (discrete logic) 8 ms (typical)	• 10 ms @ 57.6 kbps (remote)	No / 1

Table 2.J
Classic PLC-5 Processor Selection Chart—Part 2

Processor/ Cat. No.	Number of Remote I/O, Extended-Local I/O, and DH+ Ports	Maximum Number of I/O Racks	Maximum Number of I/O Chassis			Number of RS-232/ 422/ 423 ports	Remote I/O Transmission Rates[1]	Backplane Current Load
			Total	Ext Local	Remote			
PLC-5/10 (1785-LT4)	• 1 DH+	1	1	0	0	0	--	2.5A
PLC-5/12 (1785-LT3)	• 1 DH+ • 1 Remote I/O (Adapter Only)	4	1	0	0	0	57.6 kbps	2.5A
PLC-5/15 (1785-LT)	• 1 DH+ • 1 Remote I/O (Adapter or Scanner)	4	13	0	12	0	57.6 kbps	2.5A
PLC-5/25 (1785-LT2)	• 1 DH+ • 1 Remote I/O (Adapter or Scanner)	8	17	0	16	0	57.6 kbps	2.5A

Selecting Power Supplies

Use the following steps as guidelines for selecting a power supply for a chassis that contains a PLC-5 processor, a 1771-AS or -ASB remote I/O adapter module, or a 1771-ALX extended-local I/O adapter module.



1771-P7

1. Determine the input voltage for the power supply.

2. Calculate the total backplane current draw for I/O modules by adding together the backplane current draw for each I/O module in that chassis.

3. Add to the total of the I/O module backplane current draw either:

  a. 3.3 Amps when the chassis will contain a PLC-5 processor (maximum current draw for any PLC-5 processor) **or**

  b. 1.2 Amps when the chassis will contain either a remote I/O 1771-AS or -ASB module or a 1771-ALX extended-local I/O adapter module

4. If you leave slots available in your chassis for future expansion:

  a. list backplane current draw for future I/O modules

  b. add the total current draw for all expansion I/O modules to the total calculated in step 3.

5. Determine whether the available space for the power supply is in the chassis or mounted external to the chassis.

Choose your power supply from Table 2.K or Table 2.L using the input voltage requirement and the total backplane current draw as determined in the previous steps, 1 through 5.

See the Automation Products Catalog, publication AP100, for more information on power supplies.

Powering a Chassis Containing a PLC-5 Processor

Table 2.K lists the power-supply modules that you can use with a Classic PLC-5 processor.

**Table 2.K
Powering a Chassis Containing a Classic PLC-5 processor**

Power Supply	Input Power	Output Current (in Amps)	Output Current (in Amps) When Parallel with:							Power Supply Location
			P3	P4	P4S	P4S1	P5	P6S	P6S1	
1771-P3	120V ac	3	6	11	11					chassis, 1-slot
1771-P4	120V ac	8	11	16	16					chassis, 2-slot
1771-P4S	120V ac	8	11	16	16					chassis, 1-slot
1771-P4S1	100V ac	8				16				
1771-P4R	120V ac	8/16/24[1]								
1771-P5	24V dc	8					16			chassis, 2-slot
1771-P6S	220V ac	8						16		chassis, 1-slot
1771-P6S1	200V ac	8							16	
1771-P6R	220V ac	8/16/24[1]								
1771-P7	120/220V ac	16								external[2]
1771-PS7	120/220V ac	16								

[1] See publication 1771-2.136 for more information.

[2] You cannot use an external power supply and a slot-based power supply module to power the same chassis; they are not compatible.

Powering a Remote I/O Chassis Containing a 1771-AS or 1771-ASB or an Extended-Local I/O Chassis Containing a 1771-ALX

Table 2.L lists the power supply modules that you can use with a remote I/O chassis or an extended-local I/O chassis.

Table 2.L
Powering a Remote I/O Chassis (Containing a 1771-AS or -ASB) or an Extended-Local I/O Chassis (Containing a 1771-ALX)

Power Supply	Input Power	Output Current (in Amps)	Output Current (in Amps) When Parallel with:							Power Supply Location
			P3	P4	P4S	P4S1	P5	P6S	P6S1	
1771-P3	120V ac	3	6	11	11					chassis, 1-slot
1771-P4	120V ac	8	11	16	16					chassis, 2-slot
1771-P4S	120V ac	8	11	16	16					chassis, 1-slot
1771-P4S1	100V ac	8				16				
1771-P4R	120V ac	8/16/24 [1]								
1771-P5	24V dc	8					16			chassis, 2-slot
1771-P6S	220V ac	8						16		chassis, 1-slot
1771-P6S1	200V ac	8							16	
1771-P6R	220V ac	8/16/24 [1]								
1771-P1	120/220V ac	6.5								external [2]
1771-P2	120/220V ac	6.5								
1771-P7	120/220V ac	16								
1771-PS7	120/220V ac	16								
1777-P2	120/220V ac	9								
1777-P4	24V dc	9								

1 See publication 1771-2.136 for more information.

2 You cannot use an external power supply and a slot-based power supply module to power the same chassis; they are not compatible.

Selecting Memory Modules

Select a memory module from Table 2.M for your PLC-5 processor.

Table 2.M
PLC-5 Processor Memory Modules

Nonvolatile Memory Backup (EEPROM)		RAM Memory (CMOS)	
Words	**Catalog Number (and Processor)**	**Words**	**Catalog Number (and Processor)**
8 K	1785-MJ	4 K	1785-MR (PLC-5/15 and -5/25)
16 K	1785-MK (PLC-5/25)	8 K	1785-MS (PLC-5/15 and -5/25)

Selecting a Replacement Battery

A battery ships with your PLC-5 processor. Select a replacement battery using Table 2.N and Table 2.O. See the Allen-Bradley Guidelines for Handling Lithium Batteries, publication ICCG-5.14, for more information.

Table 2.N
Processor Batteries

Processor	Battery[1]	Function
PLC-5/10, -5/12, -5/15, and -5/25	1770-XY, AA lithium	Retains the processor memory and the memory in an optional CMOS RAM module if the processor is not powered.

[1] The 1770-XY is a 3.6 Volt AA size lithium thionyl chloride battery manufactured by Tadiran as their part number TL 5104 and type AEL/S.

Table 2.O
Average Battery Life

Battery	Temperature	Power Off 100% (Average)	Power Off 50% (Average)
1770-XY	60° C 25° C	329 days 2 years	1.4 years 3.3 years

Selecting Complementary I/O

You configure complementary I/O by assigning an I/O rack number of one I/O chassis (primary) to another I/O chassis (complementary). You complement I/O functions in the primary chassis with opposite functions in the complementary chassis. Use chapter 4, "Assigning Addressing Mode, Racks, and Groups," in conjunction with the following selection of complementary I/O hardware.

Use the following modules in either primary or complementary I/O chassis opposite any type of module:

- Communication Adapter Module (1771-KA2)
- Communication Controller Module (1771-KE)
- PLC-2 Family/RS-232-C Interface Module (1771-KG)
- Fiber Optics Converter Module (1771-AF)
- DH/DH+ Communication Adapter Module (1785-KA)
- DH+/RS-232C Communications Interface Module (1785-KE)

Use the following modules in either primary or complementary I/O chassis opposite any type of module. However, these modules do not work as standalone modules; each one has an associated master module. Use care when placing the master modules in the I/O chassis (refer to the paragraph on Master/Expander I/O modules):

- Analog Input Expander Module (1771-E1, -E2, -E3)
- Analog Output Expander Module (1771-E4)
- Servo (Encoder Feedback) Expander Module (1771-ES)
- Pulse Output Expander Module (1771-OJ)

Selecting a PLC-5 Processor Backup System

A PLC-5 processor backup system contains **two** of each of the following hardware components:

- Classic PLC-5 processor module

Processor	Catalog Number
PLC-5/15	1785-LT Series B
PLC-5/25	1785-LT2

- 1785-BCM Series C Backup Control Module (for 2 channels)
- 1785-BEM Backup Expansion Module (for 2 additional channels)
- Power supply
- Local chassis

Important: The PLC-5 backup system does not back up I/O in the processor-resident local chassis. Do not install I/O in the processor-resident local chassis of a backed up system.

Refer to the PLC-5 Backup Communication Module User Manual, publication 1785-6.5.4, for more information on configuring a PLC-5 processor backup system.

Selecting Link Terminators

Terminate remote I/O links by setting switch assembly SW3. If you cannot use an 82-Ohm terminator because of devices that you place on your I/O link (see the table below for a list of these devices), you must use 150-Ohm terminators. Using the higher resistance reduces the quantity of devices to 16 that you can place per remote I/O link. Also, this limits your communication rates to 57.6 kbps and 115.2 kbps.

DH+ Network Terminator

Terminate your DH+ network with a 150-Ohm, 1/2-watt terminator.

If you have this processor:	Terminate a DH+ link by:
PLC-5/10, -5/12, -5/15, or -5/25	Setting switch assembly SW3 of the PLC-5 processor (refer to your Classic 1785 PLC-5 Family Programmable Controllers Hardware Installation Manual, publication 1785-6.6.1).

Connecting a Programming Terminal to a Processor Module

Connect the programming terminal directly to the processor through the D-shell DH+ COMM INTFC connector on the front panel. You can also connect the programming terminal remotely to a DH+ link through the 3-pin connector or at a remote station.

Choosing Cables

Select cables from the options listed below. See chapter 3, "Placing System Hardware," to determine the lengths that you will need for cables in your system.

Remote I/O Link

Use Belden 9463 twinaxial cable (1770-CD) to connect your PLC-5 processor to remote I/O adapter modules.

Connect your I/O devices using:

- single-conductor wire (analog and some discrete applications)
- multi-conductor cable (analog and some discrete applications)
- multi-conductor shielded cable (some specialty I/O modules and low-voltage dc discrete modules)

See the Classic 1785 PLC-5 Programmable Controllers Hardware Installation Manual, publication 1785-6.6.1, and the installation data for the I/O modules that you have selected for more information on I/O wiring. Also, see Allen-Bradley Programmable Controller Wiring and Grounding Guidelines, publication 1770-4.1, and Control, Communication and Information Reference Guide, publication ICCG-1.2, for more information.

Programming Terminal

The cable that you use to connect a processor to a programming terminal depends on the communication device used. Table 2.P lists the cables that you need for different configurations.

Table 2.P
Cables for Connecting a Classic PLC-5 Processor and Programming Terminal

If you have this device:	With this communication device:	Use this cable:
PLC-5/10, -5/12, -5/15, or -5/25	1784-KT, -KT2 1784-KL, -KL/B	1784-CP
	1784-KTK1	1784-CP5
	1784-PCMK	1784-PCM5
6160-T60, 6160-T70, 6121 IBM PC/AT (or compatible)	1785-KE	1784-CAK
1784-T47, 6123, 6124 IBM PC/XT (or compatible)	1785-KE	1784-CXK
6120, 6122	1785-KE	1784-CYK

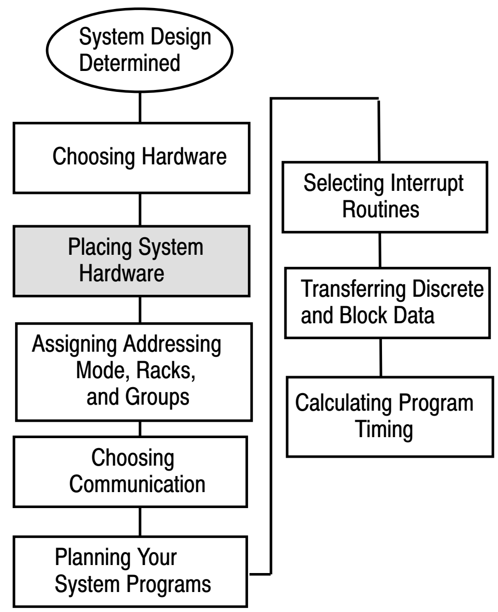
You can also use a 1770-KF2/B communication interface to connect to a PLC-5 processor. You build your own cables to connect your programming terminal via the COM1 or COM2 serial ports to the 1770-KF2/B. For the cable pin assignments, see the Classic 1785 PLC-5 Programmable Controller Hardware Installation Manual, publication 1785-6.6.1.

Chapter 3

Placing System Hardware

Chapter Objectives

A well-planned layout is essential to the proper installation of your Classic PLC-5 programmable controller system. Read this chapter for information on placing hardware.



Determining the Proper Environment

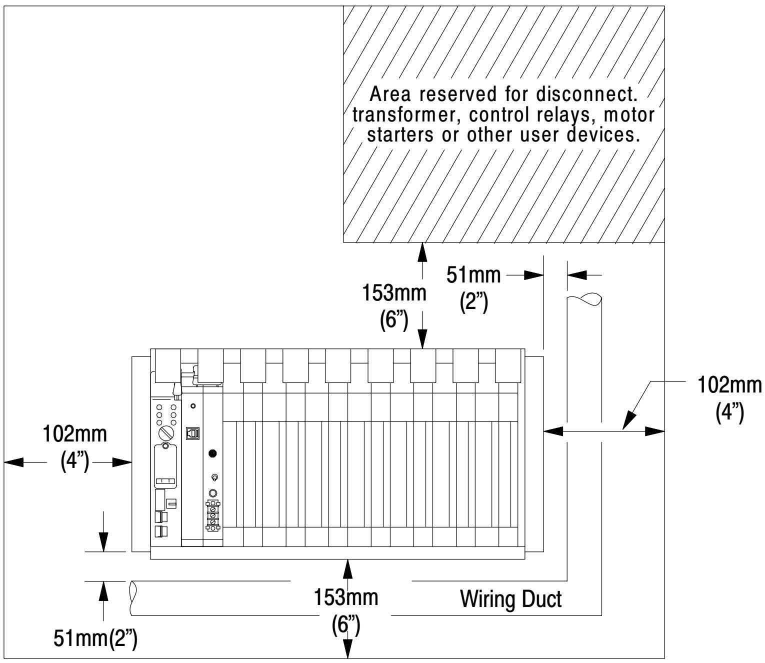
Place the processor in an environment with conditions that fall within the guidelines described in Table 3.A.

Table 3.A
Proper Environmental Conditions For Your Processor

Environmental Condition	Acceptable Range
Operating temperature	0 to 60° C (32 to 140° F)
Storage temperature	-40 to 85° C (-40 to 185° F)
Relative humidity	5 to 95% (without condensation)

Separate your programmable controller system from other equipment and plant walls to allow for convection cooling. Convection cooling draws a vertical column of air upward over the processor. This cooling air must not exceed 60° C (140° F) at any point immediately below the processor. If the air temperature exceeds 60° C, install fans that bring in filtered air or recirculate internal air inside the enclosure, or install air-conditioning/heat-exchanger units.

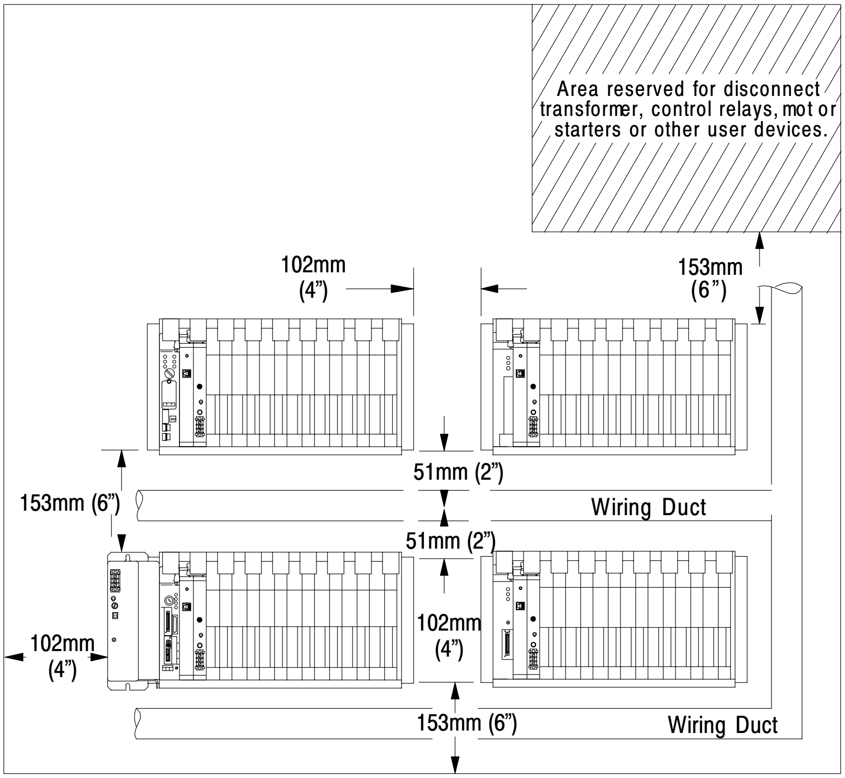
To allow for proper convection cooling in enclosures containing a processor-resident chassis and remote I/O chassis, follow these guidelines.



Minimum spacing requirements for a processor-resident chassis:

- Mount the I/O chassis horizontally.
- Allow 153 mm (6 in) above and below the chassis.
- Allow 102 mm (4 in) on the sides of each chassis.
- Allow 51 mm (2 in) vertically and horizontally between any chassis and the wiring duct or terminal strips.
- Leave any excess space at the top of the enclosure, where the temperature is the highest.

13081



Minimum spacing requirements for a remote I/O chassis:

- Mount the I/O chassis horizontally.
- Allow 153 mm (6 in) above and below all chassis. When you use more than one chassis in the same area, allow 152.4 mm (6 in) between each chassis.
- Allow 102 mm (4 in) on the sides of each chassis. When you use more than one chassis in the same area, allow 101.6 mm (4 in) between each chassis.
- Allow 51 mm (2 in) vertically and horizontally between any chassis and the wiring duct or terminal strips.
- Leave any excess space at the top of the enclosure, where the temperature is the highest.

18749

Protecting Your Processor

You provide the enclosure for your processor system. This enclosure protects your processor system from atmospheric contaminants such as oil, moisture, dust, corrosive vapors, or other harmful airborne substances. To help guard against EMI/RFI, we recommend a steel enclosure.

Mount the enclosure in a position where you can fully open the doors. You need easy access to processor wiring and related components so that troubleshooting is convenient.

When you choose the enclosure size, allow extra space for transformers, fusing, disconnect switch, master control relay, and terminal strips.

Avoiding Electrostatic Damage

ATTENTION: Under some conditions, electrostatic discharge can degrade performance or damage the processor module. Read and observe the following precautions to guard against electrostatic damage.

- Wear an approved wrist strap grounding device when handling the processor module.
- Touch a grounded object to discharge yourself before handling the processor module.
- Do not touch the backplane connector or connector pins.
- When not handling the processor module, keep it in its protective packaging.

Laying Out Your Cable Raceway

The raceway layout of a system reflects where the different types of I/O modules are placed in I/O chassis. Therefore, you should determine I/O-module placement prior to any layout and routing of wires. When planning your I/O-module placement, however, segregate the modules based on the conductor categories published for each I/O module so that you can follow these guidelines. These guidelines coincide with the guidelines for “the installation of electrical equipment to minimize electrical noise inputs to controllers from external sources” in IEEE standard 518-1982.

To plan a raceway layout, do the following:

- categorize conductor cables
- route conductor cables

Categorize Conductors

Segregate all wires and cables into categories as described in the Industrial Automation Wiring and Grounding Guidelines, publication 1770-4.1. See the installation data for each I/O module that you are using for information about its classification.

Route Conductors

To guard against coupling noise from one conductor to another, follow the general guidelines for routing cables described in the Industrial Automation Wiring and Grounding Guidelines, publication 1770-4.1. You should follow the safe grounding and wiring practices called out in the National Electrical Code (NEC, published by the National Fire Protection Association, in Quincy, Massachusetts), and local electrical codes.

Planning Cabling

DH+ Link Cabling

At a DH+ transmission rate of 57.6 kbps, do not exceed 3,048 cable-m (10,000 cable-ft) for a trunkline cable length or 30.5 cable-m (100 cable-ft) for a dropline cable length.

Remote I/O Link Cabling

Refer to Table 3.B for remote I/O link trunkline cable length restrictions.

Table 3.B
Maximum Cable Lengths per Communication Rate

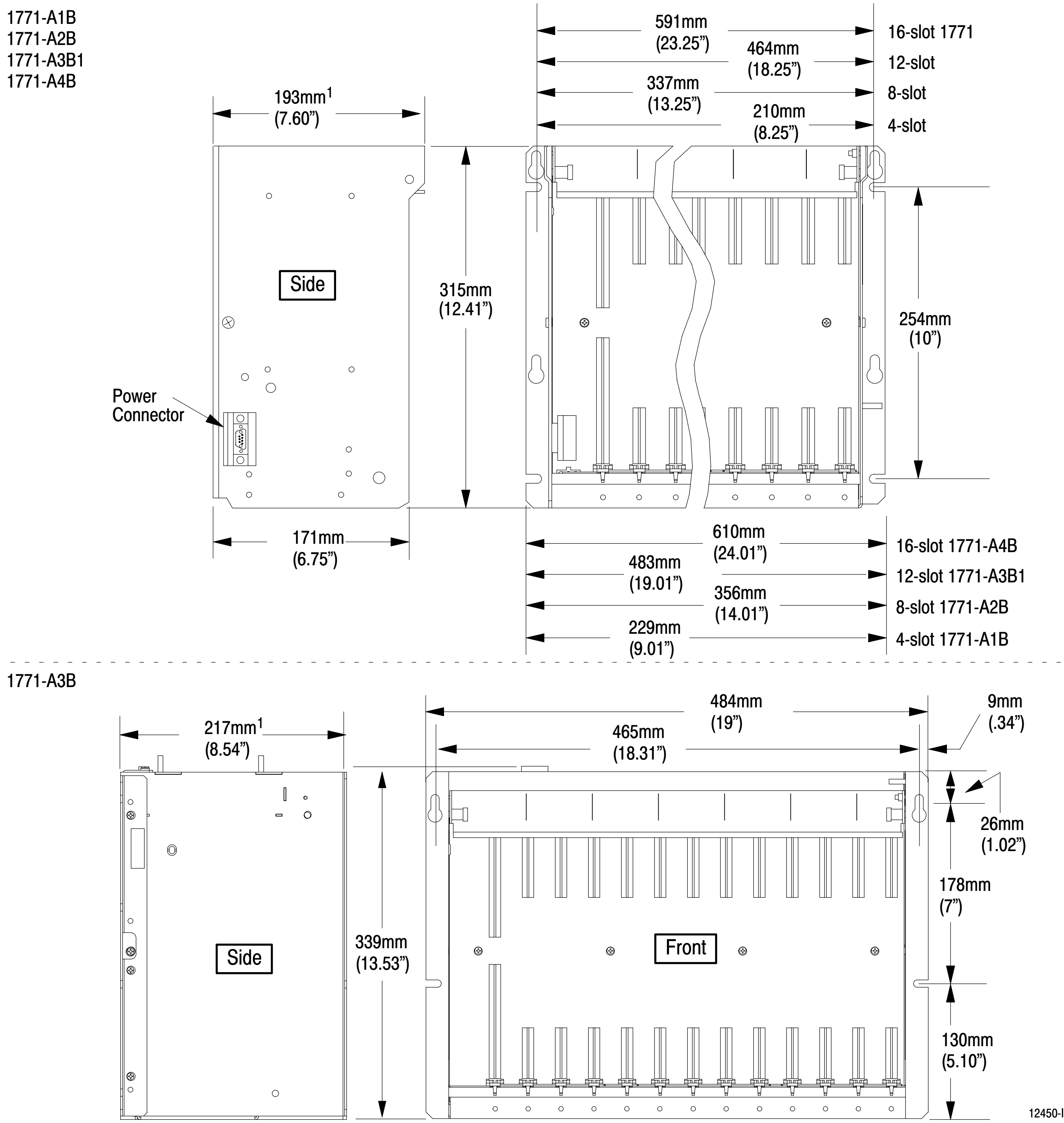
Transmission Rate	Maximum Cable Length
57.6 kbps	3,048 m (10,000 ft)
115.2 kbps	1,524 m (5000 ft)
230.4 kbps	762 m (2500 ft)

Important: All devices on the remote I/O link must be communicating at the same transmission rate.

Laying Out the Backpanel Spacing

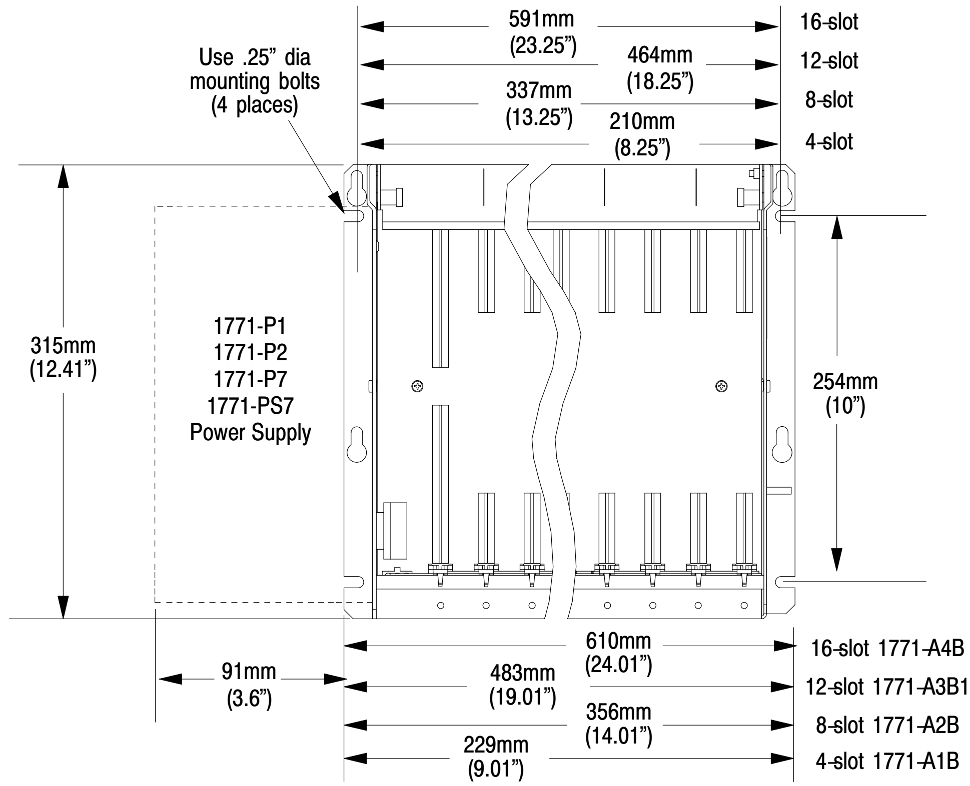
Use 6.35 mm (0.25 inch) mounting bolts to attach the I/O chassis to the enclosure backpanel.

Figure 3.1
Chassis Dimensions (Series B)



[1]Total maximum depth dimension per installation will be dependent upon module wiring and connectors.

Figure 3.2
I/O Chassis and 1771-P2 Power Supply Dimensions

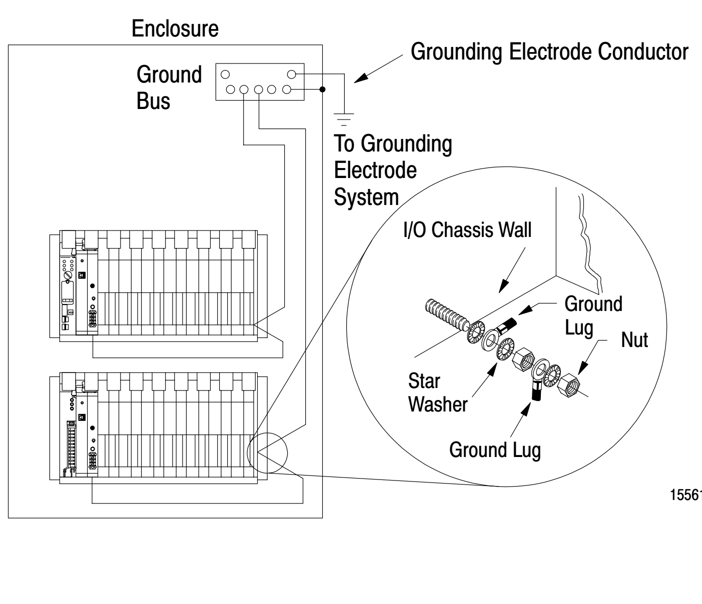


Clearance depth is 204 mm (8 in) for 8 I/O connection points per module. 12451-I

Grounding Configuration

See Figure 3.3 for the recommended grounding configuration for remote I/O systems.

Figure 3.3
Recommended Grounding Configuration for Remote I/O Systems

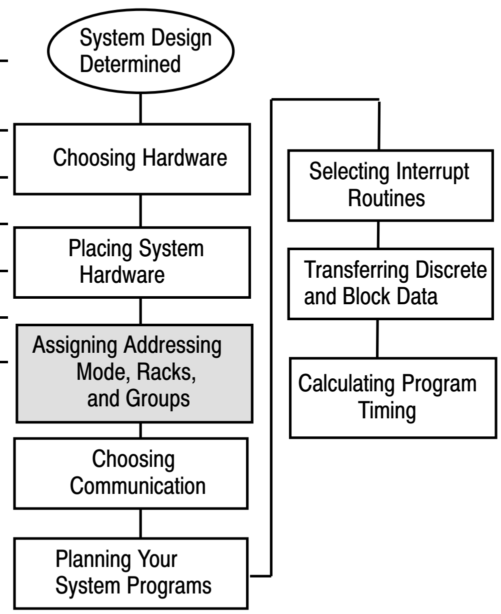


Chapter 4

Assigning Addressing Modes, Racks, and Groups

Chapter Objectives

This chapter conveys basic hardware addressing concepts and gives you guidelines with which to choose the addressing modes (including complementary I/O), racks, and groups to use in your system.



Placing I/O Modules in Chassis

Place I/O modules in a chassis depending on the electrical characteristics of the modules. The placement is made left to right, with the left-most position being closest in the chassis to the PLC-5 processor or the I/O adapter module. The placement order is as follows:

1. block-transfer modules (all types)
2. dc input modules, placed left to right from lowest to highest voltages
3. dc output modules, placed left to right from lowest to highest voltages
4. ac input modules, placed left to right from lowest to highest voltages
5. ac output modules, placed left to right from lowest to highest voltages

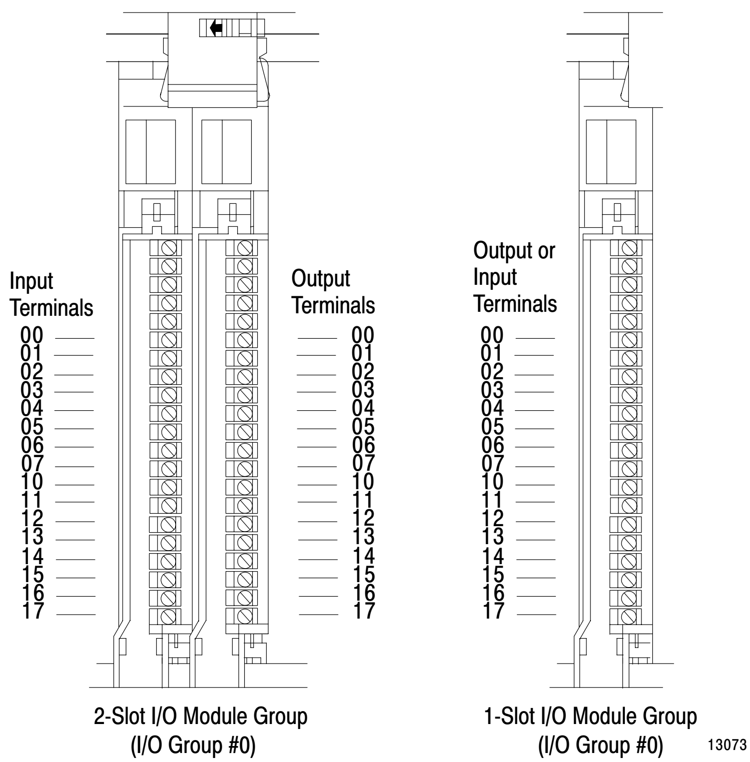
The following guidelines are for placing block-transfer modules.

- Place as many modules as possible for which you need fast block-transfer times in your processor-resident local I/O chassis .
- Place modules that need fast block-transfer times (but space is not available in processor-resident local I/O chassis) in an extended-local I/O chassis.
- Place modules in which timing is not as critical as in other block-transfer modules in remote I/O chassis.
- ac output modules should always be the furthest I/O modules from any block-transfer modules in the same chassis.

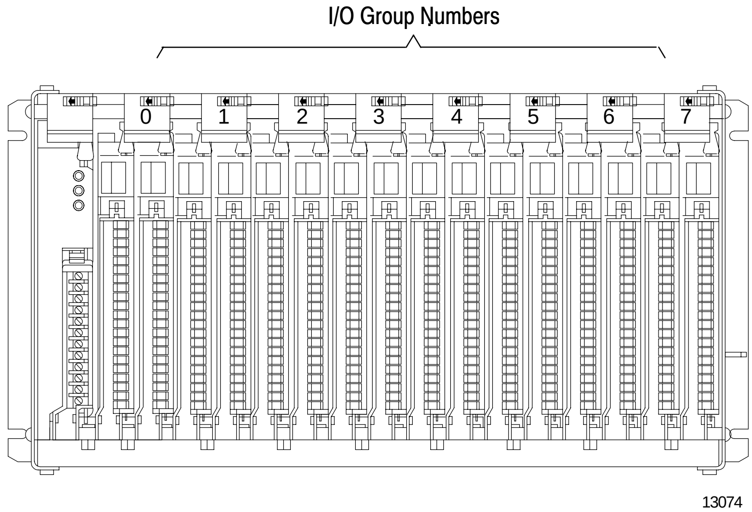
Understanding the Terms Used in this Chapter

Become familiar with the following terms and their definitions:

An **I/O group** is an addressing unit that corresponds to an input image-table word (16 bits) and an output image-table word (16 bits). An I/O group can contain up to 16 inputs and 16 outputs; and it can occupy 2-, 1-, or 1/2-module slots for addressing purposes.



An **I/O rack** is an addressing unit that corresponds to 8 input image-table words and 8 output image-table words. A rack contains 8 I/O groups.



Depending on I/O chassis size and I/O group size, an I/O rack can occupy a fraction of an I/O chassis, a full I/O chassis, or multiple I/O chassis.

Choosing the Addressing Mode

Select an addressing mode for each chassis independently, based on the type and density of the I/O modules contained therein. When you select addressing mode, limit the number of remote I/O adapters and I/O modules to the maximum number that the PLC-5 processor can support.

Using 2-Slot Addressing

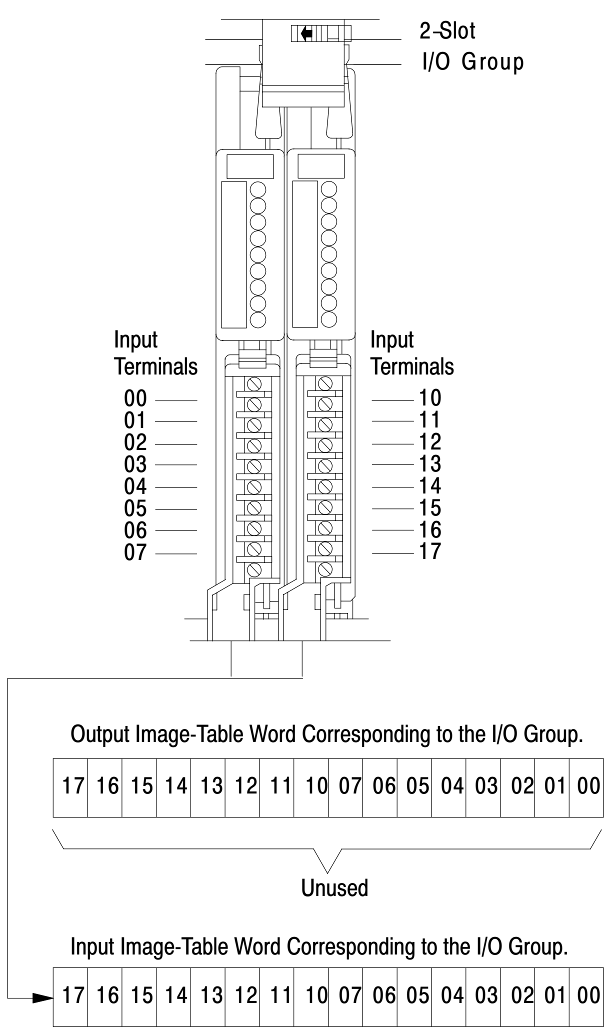
When you select **2-slot addressing**, the processor addresses two I/O module slots as one I/O group. Each physical 2-slot I/O group corresponds to one word (16 bits) in the input image table and one word (16 bits) in the output image table. The type (unidirectional or bidirectional) and density of a module that you install determines the number of bits that are used in each word.

Important: You **cannot** use 32-point I/O modules with 2-slot addressing.

8-Point I/O Modules

Eight-point digital discrete I/O modules have a maximum of eight inputs or up to eight outputs. Because they do not interfere with each other's I/O image, you can place any mix of 8-point I/O modules (including bidirectional modules, such as block-transfer modules) in any order.

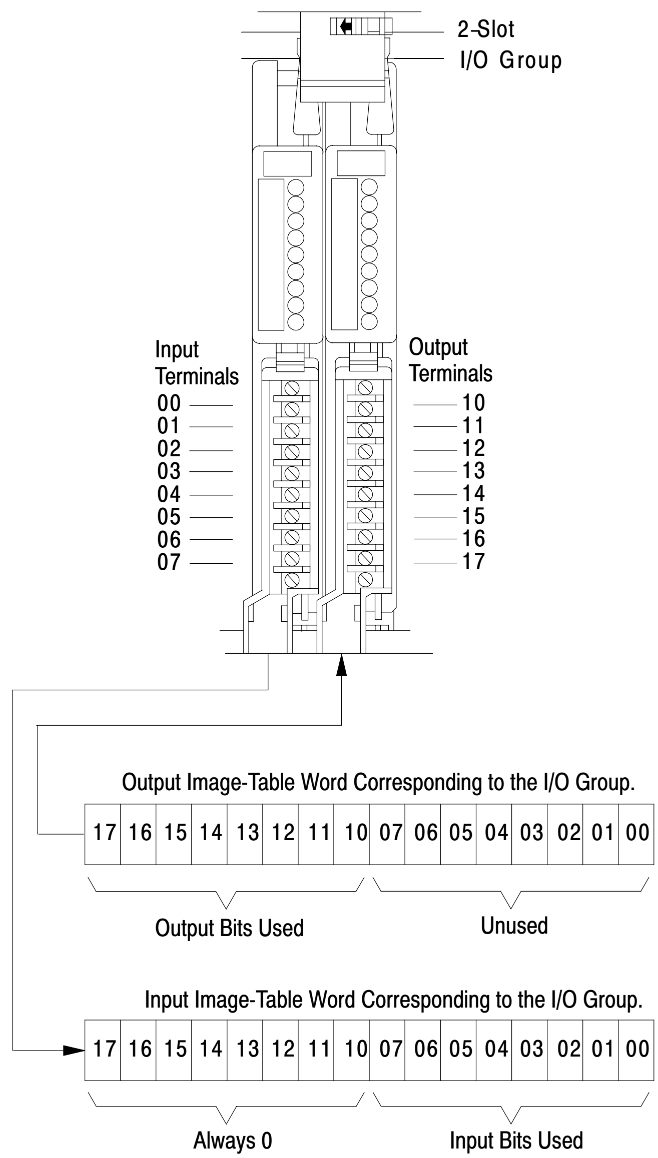
2-Slot I/O Group with Two 8-pt Input Modules



This I/O group uses 16 bits of the input image table.

11867

2-Slot I/O Group with One 8-pt Input Module and One 8-pt Output Module



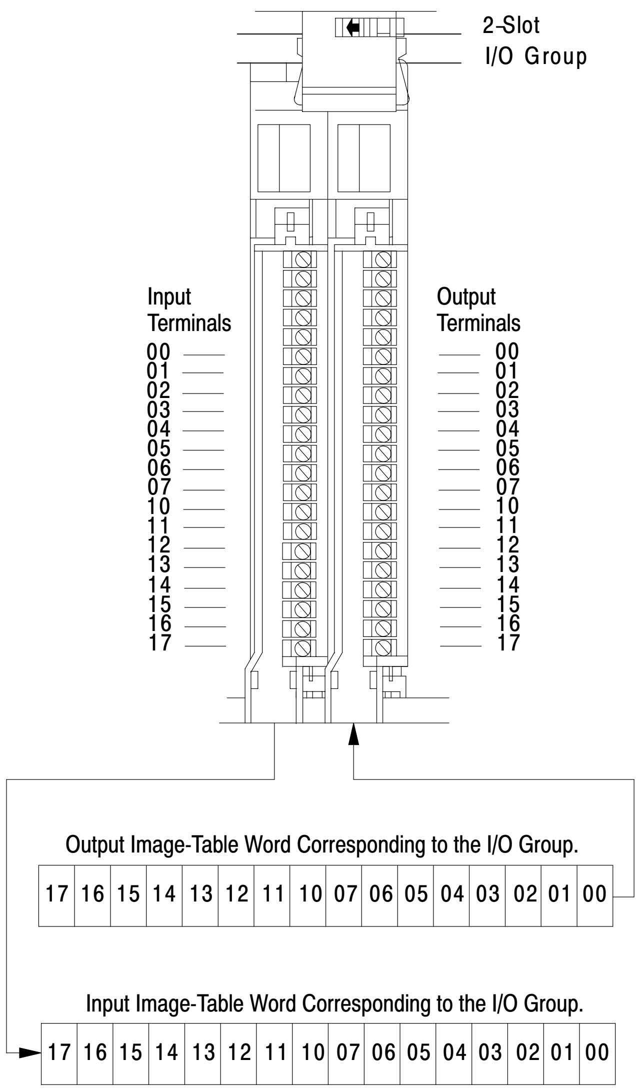
This I/O group uses 8 bits of the input image table and 8 bits of the output image table.

14965

16-Point I/O Modules

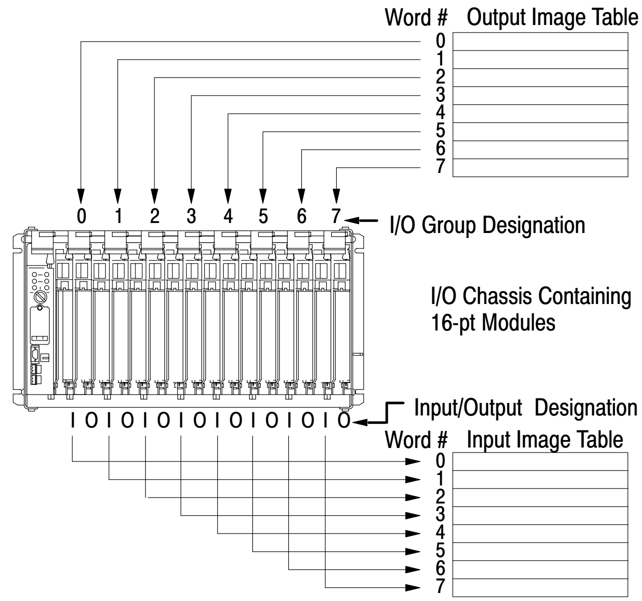
Sixteen-point digital discrete I/O modules have up to 16 inputs or up to 16 outputs. A 16-point I/O module uses a full word in the input or output image table.

2-Slot I/O Group with One 16-pt Input Module and One 16-pt Output Module



This I/O group uses 16 bits of the input image table and 16 bits of the output image table.

15559



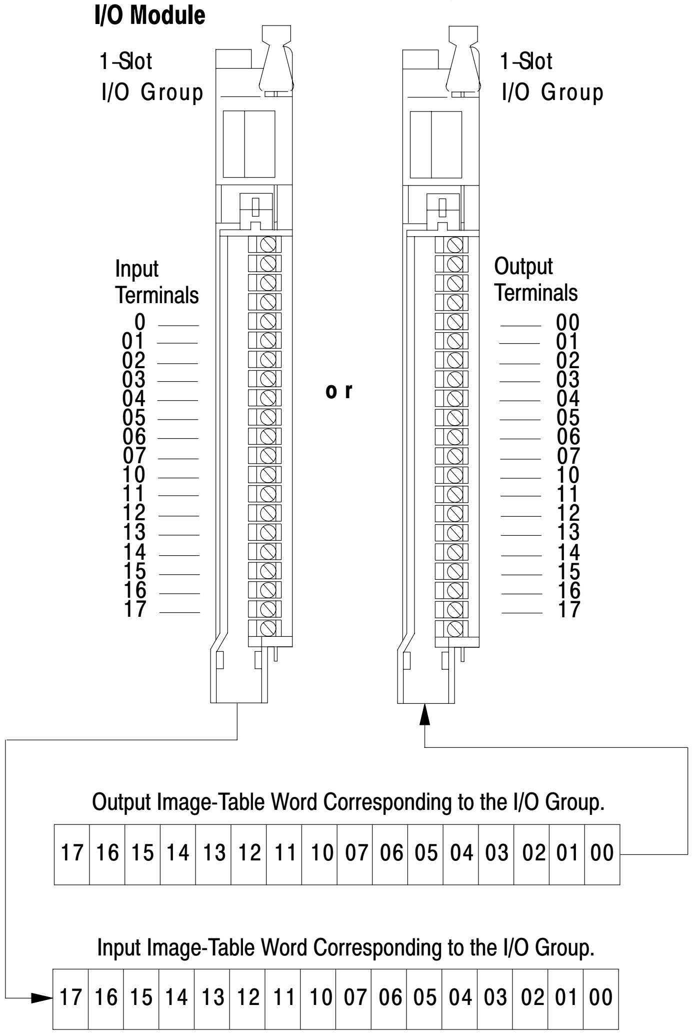
Because each 16-pt module uses a full word in the image table, the only type of module that you can install in a 2-slot I/O group with a 16-pt input module is an 8- or 16-pt output module that performs a complementary function (inputs and outputs complement each other).

Since all block-transfer modules are bidirectional, they cannot be used to complement either input or output modules.

Using 1-Slot Addressing

When you select **1-slot addressing**, the processor addresses one I/O module slot as one I/O group. Each physical slot in the chassis corresponds to an input and output image-table word. The type (unidirectional or bidirectional) and density of module that you install determines the number of bits used in these words.

1-Slot I/O Group with One 16-pt Digital Discrete I/O Module



A single 16-pt module uses an entire word of image table.

11869

8-Point I/O Modules

You can place any mix of 8- or 16-point I/O modules (including bidirectional modules such as block-transfer modules) in any order with 1-slot addressing. The 8- or 16-point modules do not interfere with the I/O image of the other 8- or 16-point modules.

16-Point I/O Modules

A single 16-point module uses an entire word of the processor image table.

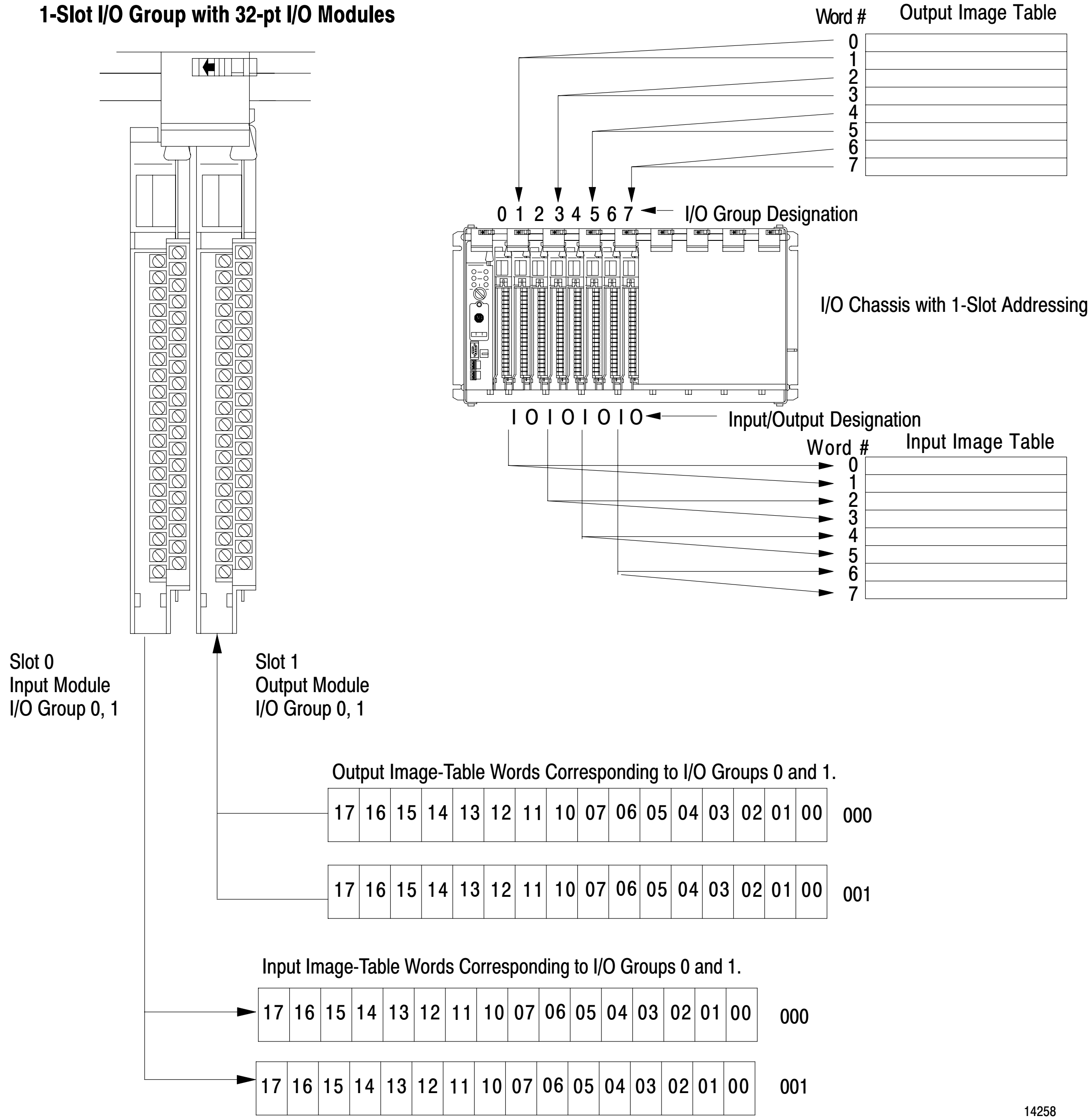
Block-Transfer Module Addressing

To address a single-slot block transfer module in a 1-slot I/O group, use the assigned I/O rack and group numbers of the slot (in which the module resides) and 0 for the module number. To address a double-slot block-transfer module, use the assigned I/O rack number, the lower assigned I/O group number, and 0 for the module number.

32-Point I/O Modules

To use 32-point I/O modules with 1-slot addressing, you must install, as a pair, an input module and an output module in two adjacent slots (even/odd pair) of the I/O chassis, beginning with I/O slot 0. If you cannot pair the modules in this way, one of the two slots of the pair must be empty. For example, if I/O slot 0 holds a 32-point input module, I/O slot 1 must hold an 8-, 16-, or 32-point output module (or a module using the backplane for power only); otherwise the slot must be empty.

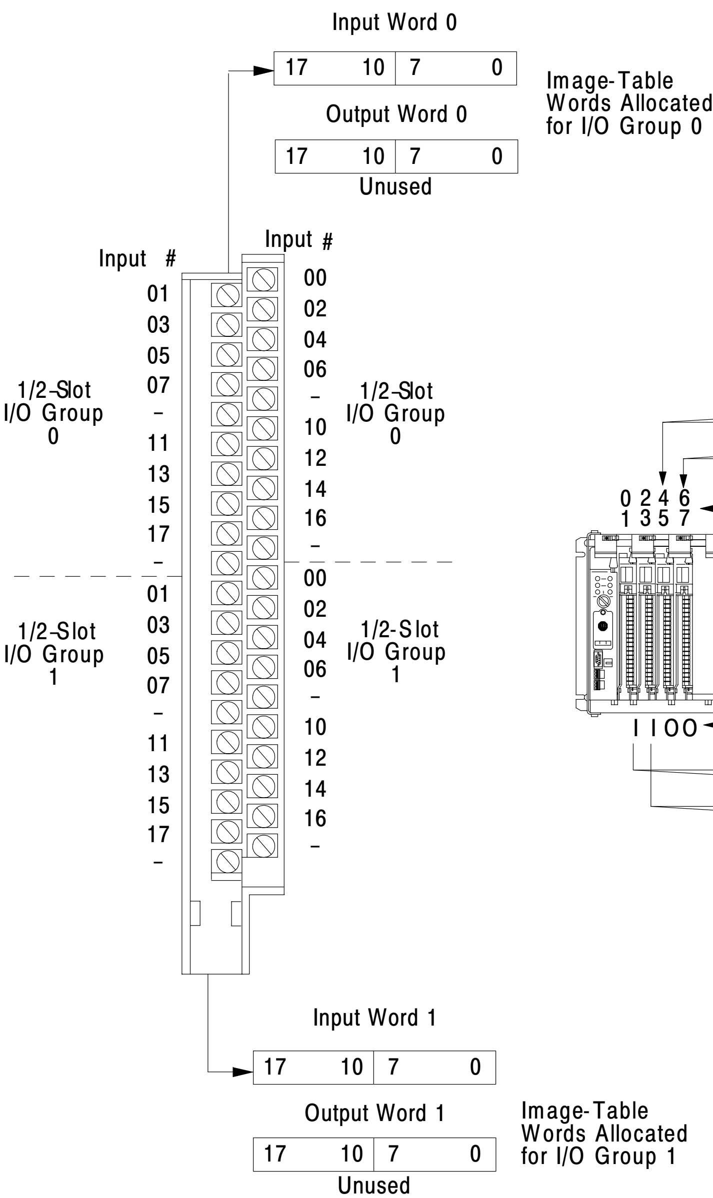
1-Slot I/O Group with 32-pt I/O Modules



Using 1/2-Slot Addressing

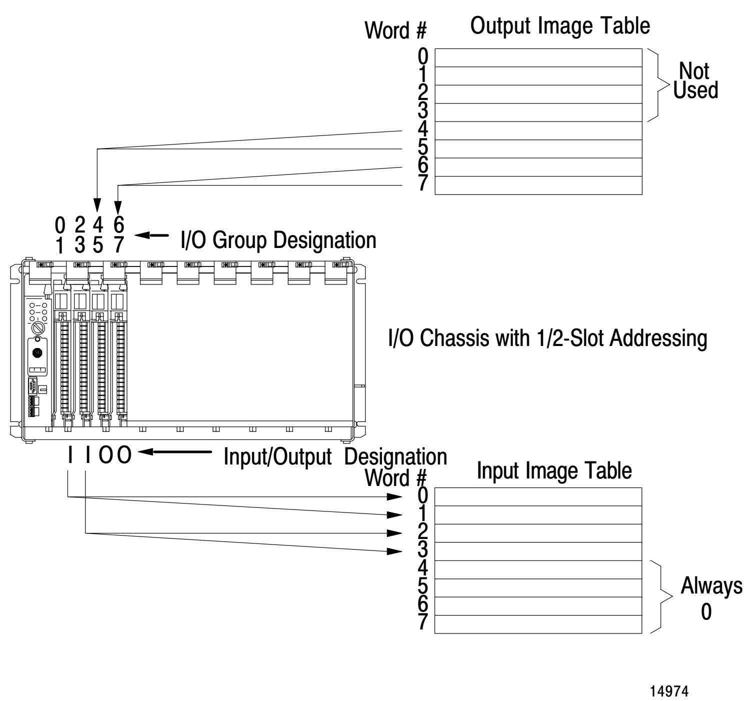
When you select **1/2-slot addressing,** the processor addresses one-half of an I/O module slot as one I/O group. Each physical slot in the chassis corresponds to two input and two output image-table words. The type (unidirectional or bidirectional) and density of the module that you install determines the number of bits that are used in each word.

1/2-Slot I/O Group with One 32-pt Input Module



This I/O group uses two words of the image table.

You can mix 8-, 16- and 32-pt I/O modules in any order in the I/O chassis because 32 input bits and 32 output bits are available in the image table for each I/O slot. When you use 8- and 16-pt I/O modules with 1/2-slot addressing, however, you use fewer total I/O bits in our image table.



Summary

Table 4.A summarizes the guidelines for selecting an addressing mode.

**Table 4.A
Addressing Mode Summary**

Addressing Mode	Guidelines
2-slot	• Two I/O module slots = 1 group • Each physical 2-slot I/O group corresponds to one word (16 bits) in the input image table and one word (16 bits) in the output image table • When you use 16-point I/O modules, you must install as a pair an input module and an output module in an I/O group; if you use an input module in slot 0, you must use an output module in slot 1 (or it must be empty). This configuration gives you the maximum usage of I/O. • You cannot use a block-transfer module and a 16-point module in the same I/O group because block-transfer modules use 8 bits in both the input and output table. Therefore, 8 bits of the 16-point module would conflict with the block-transfer module. • You cannot use 32-point I/O modules.
1-slot	• One I/O module slot = 1 group • Each physical slot in the chassis corresponds to one word (16 bits) in the input image table and one word (16 bits) in the output image table • When you use 32-point I/O modules, you must install as a pair an input module and an output module in an even/odd pair of adjacent I/O group; if you use an input module in slot 0, you must use an output module in slot 1 (or it must be empty). This configuration gives you the maximum usage of I/O. • Use any mix of 8- and 16-point I/O modules, block-transfer or intelligent modules in a single I/O chassis. Using 8-point modules results in fewer total I/O.
1/2-slot	• One half of an I/O module slot = 1 group • Each physical slot in the chassis corresponds to two words (32 bits) in the input image table and two words (32 bits) in the output image table • Use any mix of 8-, 16-, and 32-point I/O or block-transfer and intelligent modules. Using 8-point and 16-point I/O modules results in fewer total I/O. • With the processor-resident local rack set for 1/2-slot addressing, you cannot force the input bits for the upper word of any slot that is empty or that has an 8-point or 16-point I/O module. For example, if you have an 8-point or a 16-point I/O module in the first slot of your local rack (words 0 and 1 of the I/O image table, 1/2-slot addressing), you cannot force the input bits for word 1 (I:001) on or off.

Assigning Racks

The number of racks in a chassis depends on the chassis size and the addressing mode:

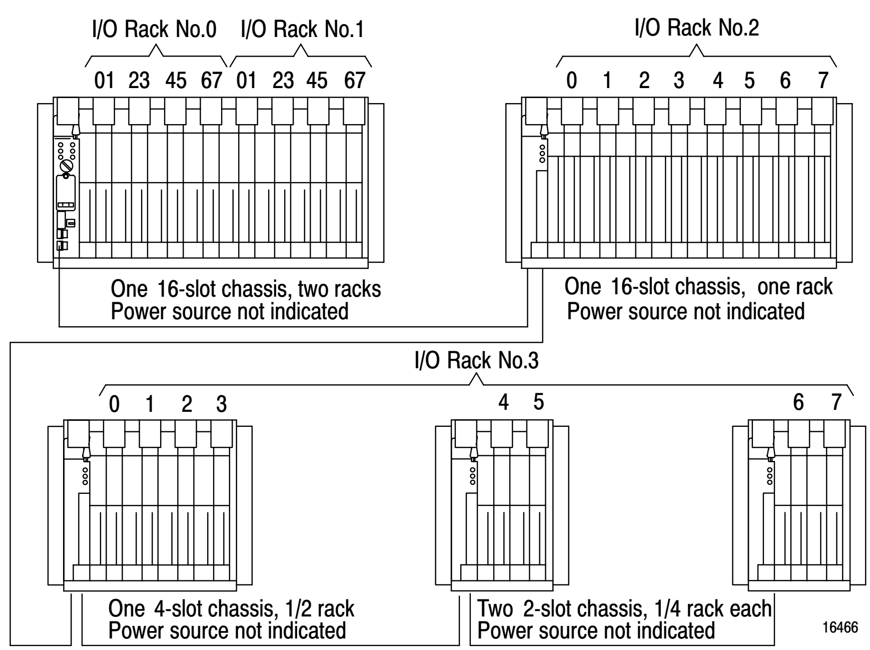
If using this chassis size:	With 2-slot addressing, rack type is:	With 1-slot addressing, rack type is:	With 1/2-slot addressing, rack type is:
4-slot	1/4 rack	1/2 rack	1 rack
8-slot	1/2 rack	1 rack	2 racks
12-slot	3/4 rack	1-1/2 racks	3 racks
16-slot	1 rack	2 racks	4 racks

When assigning rack numbers, use the following guidelines:

- One I/O rack number is eight I/O groups, regardless of the addressing mode that you select.
- You can assign from **one to four racks in your processor-resident local chassis (128 inputs and 128 outputs)** depending on the chassis size and addressing mode. You cannot split a processor-resident local I/O rack over two or more chassis or assign unused processor-resident local I/O groups to remote I/O racks.
- The default address of the processor-resident local rack is 0. You can change the default to 1 by setting bit 2 in the processor control word (S:26) on the processor configuration screen; you must also change the mode of the processor from run to program to run.
- An extended-local I/O and a remote I/O chassis cannot be addressed by the same I/O rack number. For example, if an 8-slot extended-local I/O chassis is configured as I/O groups 0-3 of I/O rack 2, an 8-slot remote I/O chassis cannot be configured as I/O groups 4-7 of I/O rack 2.

Remote I/O Racks

You can assign a remote I/O rack to a fraction of a chassis, a single I/O chassis, or multiple I/O chassis:



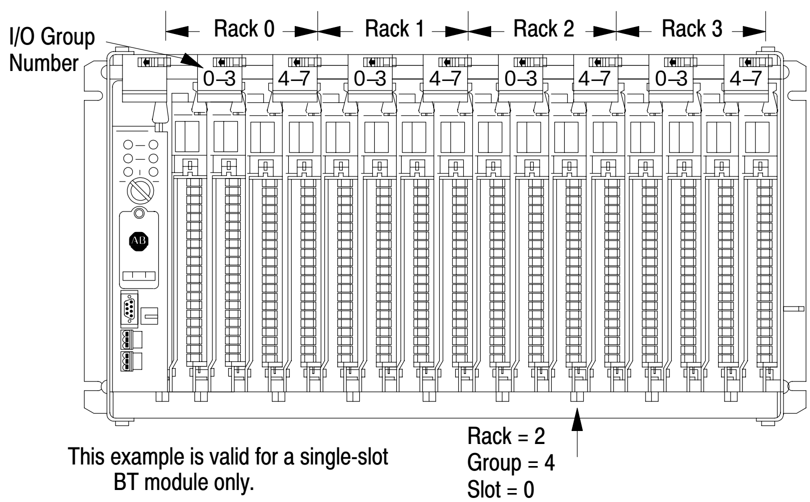
When assigning remote I/O rack numbers, use the following guidelines:

- Limit the number of remote I/O rack numbers to those that your PLC-5 processor can support.
- The PLC-5 processor and the 1771-ASB adapter module automatically allocate the next higher rack number(s) to the remaining I/O groups of the chassis. For example, if you select 1/2-slot addressing for your processor-resident local chassis and you are using a 16-slot (1771-A4B) chassis, the processor will address racks 0, 1, 2, and 3 in this chassis.

Block-Transfer Module Racks Using 1/2-Slot Addressing

To address a block-transfer module in a 1/2-slot I/O group, use the assigned rack number, the lower assigned I/O group number of the slot(s) in which the module resides, and 0 for the module number (Figure 4.4).

Figure 4.4
Example Block-Transfer Module Address Using 1/2-Slot Addressing



Addressing Complementary I/O

You configure complementary I/O by assigning an I/O rack number of one I/O chassis (primary) to another I/O chassis (complementary), complementing modules I/O group for I/O group. The I/O modules in the complementary chassis perform the opposite function of the corresponding modules in the primary chassis.

The PLC-5/15 and -5/25 processors operating as a remote I/O scanner support complementary I/O.

Use these guidelines when you configure your remote system for complementary I/O:

- Assign the complementary I/O rack number to a chassis of any size.
- Do not place an input module opposite an input module; they will use the same bits in the input image table.
- You can place an output module opposite another output module; they use the same bits in the output image table. This allows you to use one output module to control a machine and use the other module with the same address to control an annunciator panel to display the machine condition. We do not, however, recommend this placement of modules for redundant I/O.
- You cannot configure the PLC-5 processor-resident local chassis with complementary I/O. The PLC-5 processor communicates with each processor-resident local I/O chassis as if it were a full I/O rack (eight I/O groups). Thus, if the processor-resident local chassis contains four I/O groups, the remaining four I/O groups of that I/O rack are unused; you cannot assign them to another chassis.
- You cannot use complementary I/O with a chassis that uses a combination of 32-point I/O modules and 1-slot addressing or 16-point I/O modules with 2-slot addressing.

  Important: For the PLC-5/15 and -5/25 processors, an autoconfigure is performed before the scanner begins communicating with the adapter.

Placing the Modules with 2-Slot Addressing

Figure 4.5 shows a possible module placement to configure complementary I/O using 2-slot addressing.

Figure 4.5
Complementary I/O Configurations with 2-Slot Addressing

Primary 16-Slot Chassis	I_8	I_8	O_8	O_8	I_{16}	O_{16}	O_8	O_8	BT	I_8 O_8 BT [2]	BT	O_{16}	Double-slot BT		Double-slot BT	
I/O Group Number	0		1		2		3		4		5		6		7	
Complementary 16-Slot Chassis	O_8	O_8	I_8	I_8	EMPTY	EMPTY	O_8 [1]	O_8 [1]	EMPTY [3]	O_8 [3]	EMPTY [3]	EMPTY [3]	EMPTY [3]	O_8	EMPTY [3]	O_8

Example A

Primary 16-Slot Chassis	I_{16}	O_{16}	I_{16}	O_{16}	I_{16}	O_{16}	I_{16}	O_{16}	I_{16}	O_{16}	I_{16}	O_{16}	I_{16}	O_{16}	I_{16}	O_{16}
I/O Group Number	0		1		2		3		4		5		6		7	
Complementary Chassis Not Allowed Except for Output	Outputs in the complementary chassis use the same bits in the output image table as the outputs in the primary chassis.															

Example B

I = Input Module O = Output Module BT = Block Transfer Module 8 = 8-point I/O Modules 16 = 16- point I/O Modules
1 Output modules use the same output image transfer bits
2 Can be 8-point input or output module or single-slot block transfer module
3 Must be empty if corresponding primary slot is block transfer module

13079

Placing the Modules with 1-Slot Addressing

Figure 4.6 shows a possible module placement to configure complementary I/O using 1-slot addressing.

Figure 4.6
Complementary I/O Configurations with 1-Slot Addressing

Primary 16-Slot Chassis	I	I	O	O	I	O	O [1]	BT	Double-slot BT		O	I	I	I	O	O
I/O Group Number	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
Complementary 16-Slot Chassis	O	O	I	I	O	I	O [1]	EMPTY [3]	EMPTY [3]	I, O, BT [2]	I	O	O	O	I	I

Example A

Primary 16-Slot Chassis	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I
I/O Group Number	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
Complementary 16-Slot Chassis	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O

Example B

I = Input Module (8 or 16-point)
O = Output Module (8 or 16-point)
BT = Block Transfer Module

1 Output modules use the same output image table bits
2 Can be input or output module (8 or 16-point) single-slot block transfer module
3 Must be empty if corresponding primary slot is block transfer module

13080

Placing the Modules with 1/2-Slot Addressing

Figure 4.7 shows a possible module placement to configure complementary I/O using 1/2-slot addressing.

Figure 4.7
Complementary I/O Configurations with 1/2-Slot Addressing

Primary 12-Slot Chassis	I	I	O	O	I	O	O 1	BT	Double-slot BT		O	I
I/O Group Number	01	23	45	67	01	23	45	67	01	23	45	67
Complementary 12-Slot Chassis	O	O	I	I	O	I	O 1	EMPTY 3	EMPTY 3	I, O, BT 2	I	O

Example A

Primary 12-Slot Chassis	I	I	I	I	I	I	I	I	I	I	I	I
I/O Group Number	01	23	45	67	01	23	45	67	01	23	45	67
Complementary 12-Slot Chassis	O	O	O	O	O	O	O	O	O	O	O	O

Example B

I = Input Module (8, 16, 32-point)
O = Output Module (8, 16, 32-point)
BT = Block Transfer Module

1 Output modules use the same output image table bits
2 Can be input or output module (8 or 16-point) single-slot block transfer module
3 Must be empty if corresponding primary slot is block transfer module

14261

Placing Complementary I/O Modules

See Table 4.B for a summary of 8-, 16-, and 32-point I/O module placement guidelines. See Table 4.C for a summary of block-transfer module placement guidelines.

Table 4.B
Placement Summary for 8-, 16-, and 32-point Modules Used in Complementary I/O

Addressing Method	Guidelines	
	Types of Modules Used:	Placement
2-Slot	8- point	Install input modules opposite output modules and output modules opposite input modules.
1-Slot	8- point, 16-point,	
1/2-Slot	8- point, 16-point, 32-point	

Table 4.C
Placement Summary for Block-transfer Modules Used in Complementary I/O

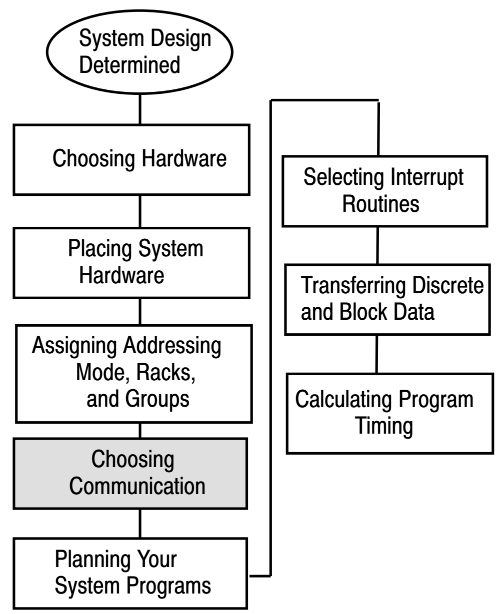
Addressing Method	Block-Transfer Placement Guidelines in Primary Chassis	
	Using single-slot modules:	Using double-slot modules:
2-Slot	• The right slot of the primary I/O group can be another single-slot block transfer module, or an 8-point input or output module. • The left slot of the complementary I/O group must be empty. • In the right slot of the complementary I/O group, you can place an 8-point output module; this slot must be empty if the corresponding slot in the primary I/O group is a single-slot block transfer module.	• The left slot of the complementary I/O group must be empty. • In the right slot of the complementary I/O group, you can only place an 8-point output module (if any).
1-Slot	Leave the corresponding I/O group in the complementary chassis empty.	• The left slot of the two corresponding I/O slots in the complementary chassis must be empty. • In the right slot of the two corresponding I/O slots in the complementary chassis, you can place an input, output, or single-slot block transfer module (if any); the modules can be either 8-point or 16-point I/O modules.
1/2-Slot	Leave the corresponding I/O group in the complementary chassis empty.	• The left slot of the two corresponding I/O slots in the complementary chassis must be empty. • In the right slot of the two corresponding I/O slots in the complementary chassis, you can place an input, output, or single-slot block transfer module (if any); the modules can be 8-point ,16-point and/or 32-point I/O modules.

Chapter 5

Choosing Communication

Chapter Objectives

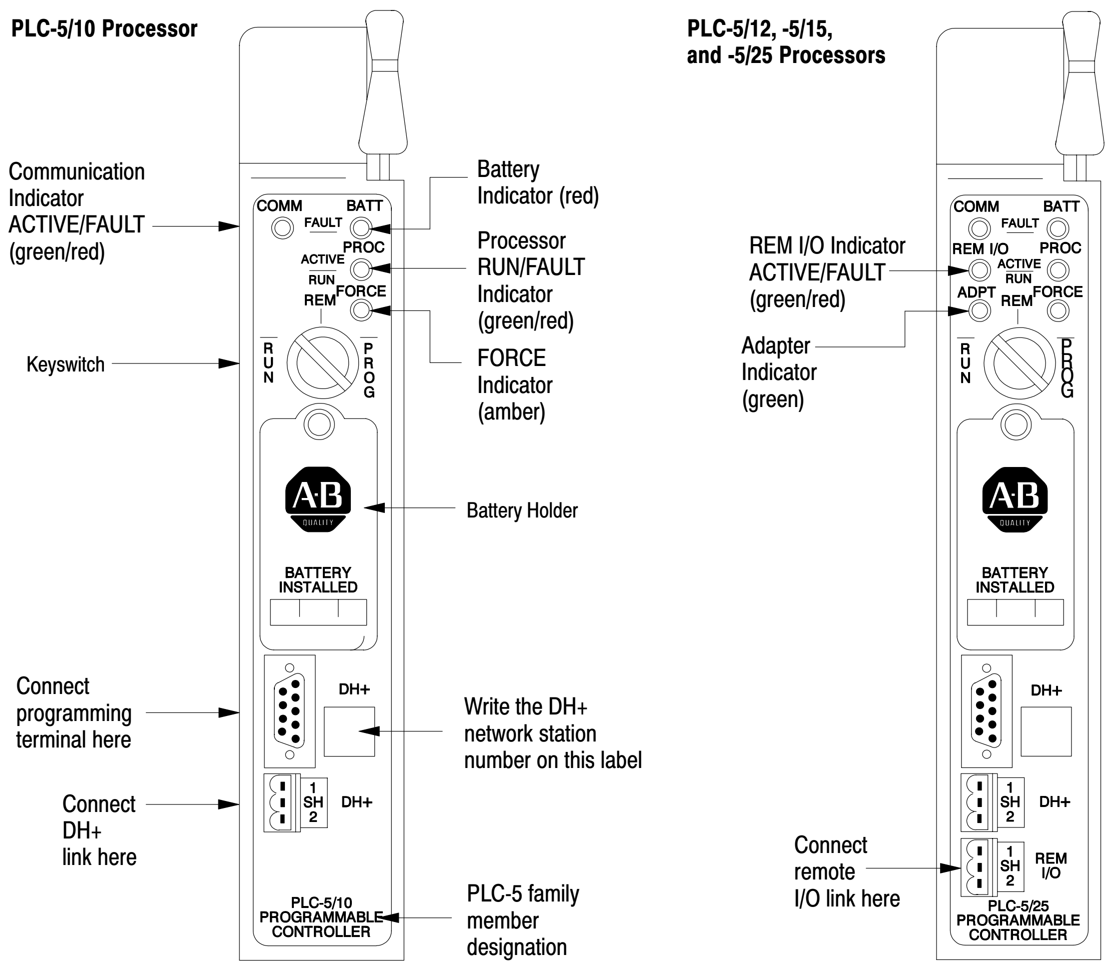
Use this chapter to choose the appropriate communication for your application.



Identifying Classic PLC-5 Processor Channels/Connectors

This section illustrates and describes the processor front-panels. After you are familiar with the processor hardware, see page 5-3 for information on configuring communication.

Figure 5.1
Processor Front Panels



Connector Name	Connector Type	Description
Programming terminal	9-pin, D-shell	Use this connector to directly connect a programming terminal to the processor. This programming terminal connector has a parallel connection with the 3-pin DH+ communications link connector.
DH+ communications link	3-pin	Use this connector to connect to DH+ communications link.
Remote I/O	3-pin	Use this connector for the remote I/O link. (This connector is **not** available for a PLC-5/10 processor.)

Configuring Communication for Your Processor

You select scanner or adapter mode for your PLC-5 processor by setting switches.

Configure Processor Communication

You configure the processor by setting switch assemblies SW1 and SW2 on the processor. See Appendix A for information on switch settings. Follow these steps to plan configuration for your processor.

1. Select scanner or adapter mode on switch assembly SW1 (the PLC-5/10 and -5/12 can not be configured as scanners).

2. If you select adapter mode, assign a rack address (rack number 0-77 octal) on switch assembly SW2. The supervisory processor uses this address to reference the adapter-mode processor.

3. If you select adapter mode, specify the simulated chassis size, either an 8-slot or 16-slot I/O chassis, and the corresponding first I/O group on switch assembly SW2. The simulated chassis size and first I/O group determine the number of discrete-transfer data words (4 words for an 8-slot chassis, 8 words for a 16-slot chassis) that the processor transfers to and from the supervisory processor during the supervisory processor's remote I/O scan.

Note that the actual size of the chassis has no bearing on the simulated size of the chassis.

Configuring a DH+ Link

You can use a DH+ link for data transfer to higher level computers and as a multiple PLC-5 processor programming link. A PLC-5 processor can communicate over a DH+ link with other processors and with a programming terminal. You can connect a maximum of 64 stations to a DH+ link. The network operates under a token-passing protocol with data transfer at 57.6 kbps.

See your programming software documentation set to configure a processor for DH+ communication.

Estimating Data Highway Plus Link Performance

Many factors can affect the performance of your DH+ link, including:

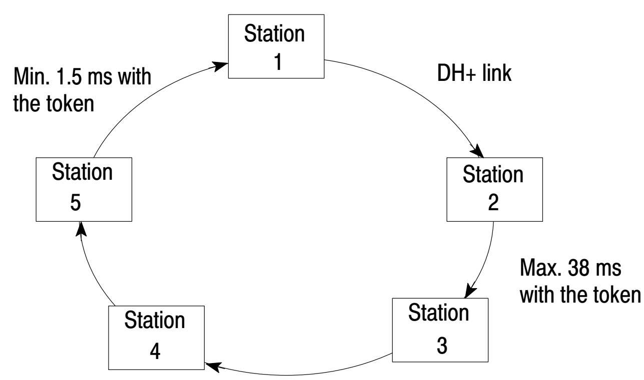
- nodes
- size and number of messages
- message destination
- internal processing time

Nodes

Nodes affect transmission time in the following ways:

- During one complete token rotation, each node on the DH+ link receives the token whether or not it has something to send.
- Each node spends from 1.5 ms (if it has no messages to send) to 38 ms (maximum time allotted) with the token, assuming there are no retries (Figure 5.2).

Figure 5.2
Token Passing



Size and Number of Messages

A PLC-5 processor encodes messages into packets for transmission on the DH+ link. The maximum number of data words in a packet depends on the sending station and command type. This limit comes from the network protocol, which limits a station to transmitting a maximum of 271 bytes per token pass. A station can send more than one message in a token pass, provided that the total number of combined command and data bytes does not exceed 271.

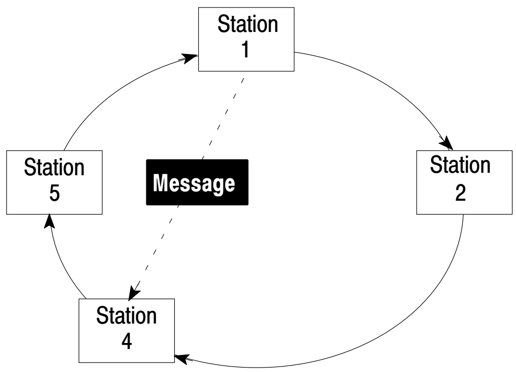
If a message exceeds the maximum packet size allotted, however, the sending station will require more than one token pass to complete the message. For example, if a PLC-5 processor wants to send a 150-word message, it will have to transmit two messages, possibly requiring many token rotations.

The number of messages a station has to send also affects throughput time. For example, if a station has three messages queued and a fourth is enabled, the fourth message may have to wait until the previous three are processed.

Message Destination

Throughput times vary depending on whether a receiving station can process the message and generate a reply before that station receives the token. Figure 5.3 assumes that station 1 wants to send a message to station 4.

Figure 5.3
Message Destination—Example 1

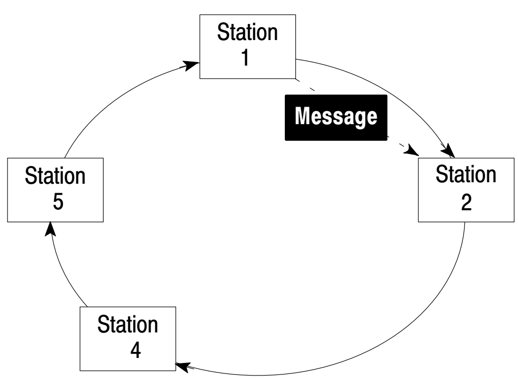


Station 1 has the token. Only the station that has the token can send a message. Station 1 sends the message to station 4. Now station 1 must pass the token on to the next highest station number, which is station 2.

Station 2 has the token. Assume that station 2 has messages to send and holds the token for 30 ms. During this time, station 4 has processed the message from station 1 and has a reply queued up. When finished, station 2 passes the token on to the next highest station number, which is station 4. Station 4 can now reply to the message from station 1. This completes the message transaction.

In Figure 5.3, station 4 has had time to process the message and generate a reply. But, that is not the case with station 2 in Figure 5.4.

Figure 5.4
Message Destination—Example 2



In Figure 5.4, we assume that station 1 wants to send the identical message as shown in Figure 5.3 but to station 2. Station 1 has the token. Station 1 sends the message to station 2 and then passes the token on to station 2. Now station 2 has the token but has not had time to generate a reply to station 1. So station 2 sends any other messages it has queued and then passes the token on to station 4. Stations 4, 5, and 1 all receive the token in order and send any messages they have queued. The token then returns to station 2, which then sends its reply to station 1. In this example, it took an extra token pass around the network to complete the message transaction even though the message was identical to the one shown in Figure 5.3.

Internal Processing Time

Internal processing time depends on how busy a given processor on the network is when sending or receiving a message.

For example, processor A has just received a READ request from processor B on the network. If processor A already has three messages of its own to send, the reply to the READ request from processor B will have to wait until the station completes the processing of the messages queued ahead of it.

Average DH+ Link Response Time Test Results

This section shows graphically the results of testing performed on a DH+ link where the number of stations and words sent in the message varies.

Figure 5.5 shows the average response time of messages of varying sizes on a DH+ link with a varying numbers of stations. It also gives you an idea of the typical response time you can expect on a given DH+ link.

Figure 5.5
Average Response Time for all PLC-5 Processors

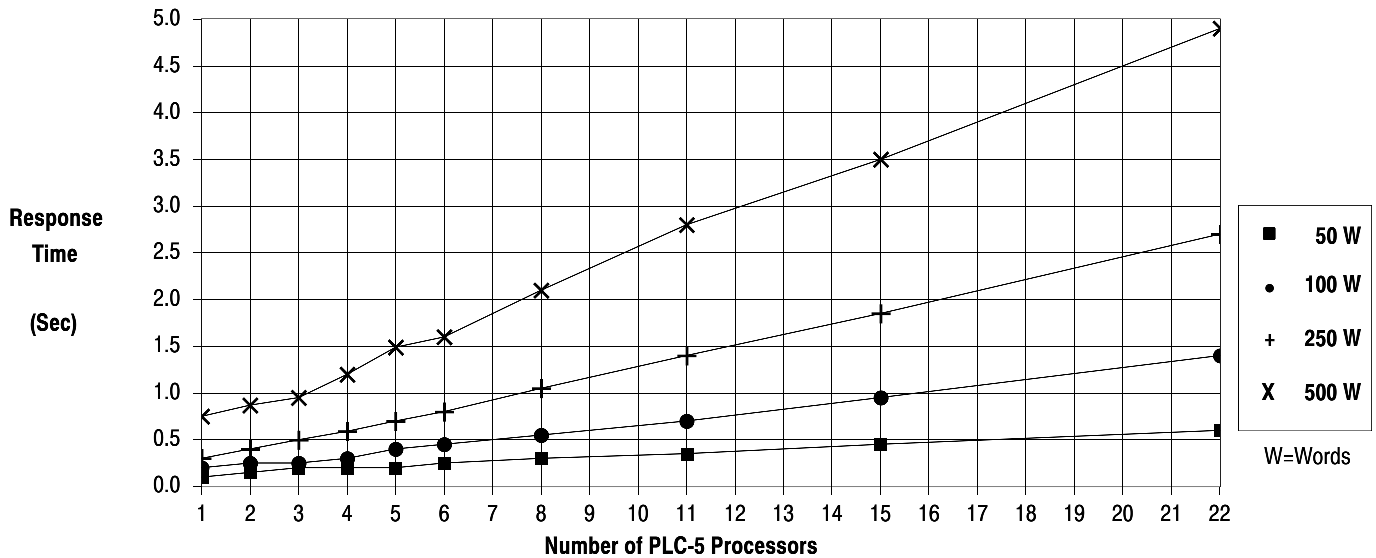
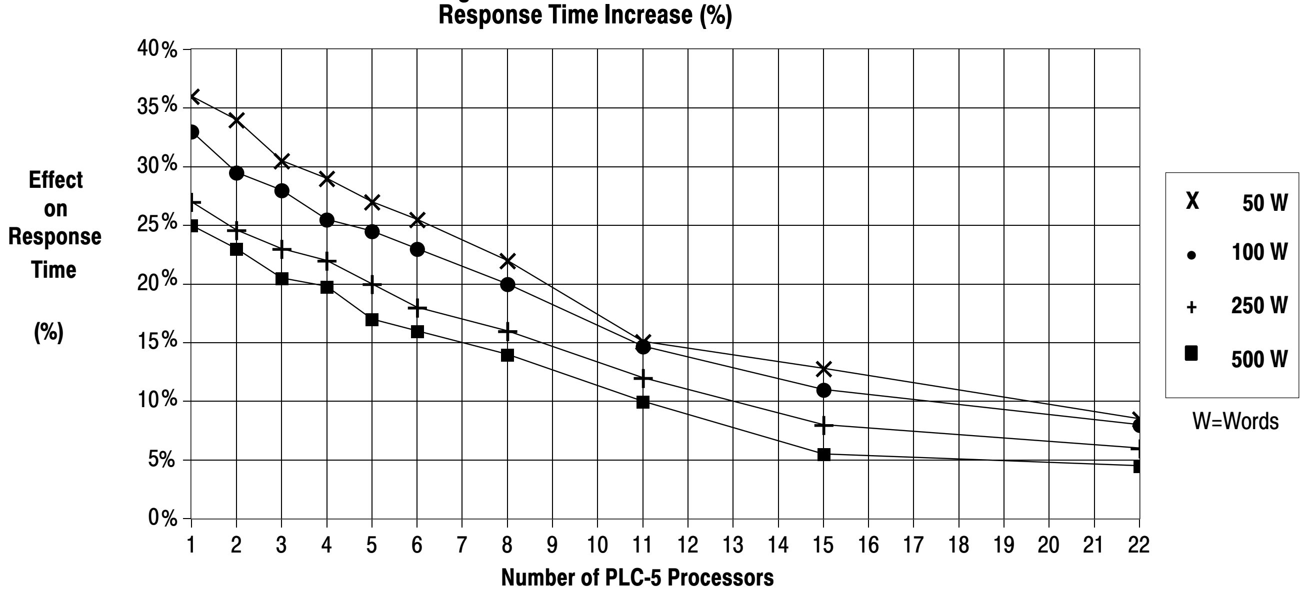


Figure 5.6 shows the effect of a programming terminal on message response time under various configurations.

Figure 5.6
Response Time Increase (%)



Test Setup

One to 22 PLC-5 processors were used with one programming terminal online. Each PLC-5 processor executes 1K of ladder logic.

Initial testing was done with one PLC-5 processor writing data to another PLC-5 processor. The response time was recorded. Additional PLC-5 processors were added to the network, each writing the same amount of data to a PLC-5 processor at the next highest station address. Four separate tests were run using data transmissions of 50, 100, 250, and 500 words.

Application Guidelines

Consider the following application guidelines when configuring a DH+ link for your system.

- Configure the number of nodes on your network dependent on the size and frequency of messages exchanged between devices.
- Limit the number of nodes on your network when you are trying to achieve fastest control response time.
- Do not add or remove nodes from the network during machine or process operation. If the network token resides with a device that is removed, the token may be lost to the rest of the network. The network is automatically re-established, but it could take several seconds. Control would be unreliable or interrupted during this time.
- Include watchdog timers in logic programs for DH+ transfer of data (to provide an orderly shutdown if failure occurs).
- Do not program processors online during machine or process operation. This could result in long bursts of DH+ activity that could increase response time.
- When possible, add a separate DH+ link for programming processors to keep effects of the programming terminal from the process DH+ link.

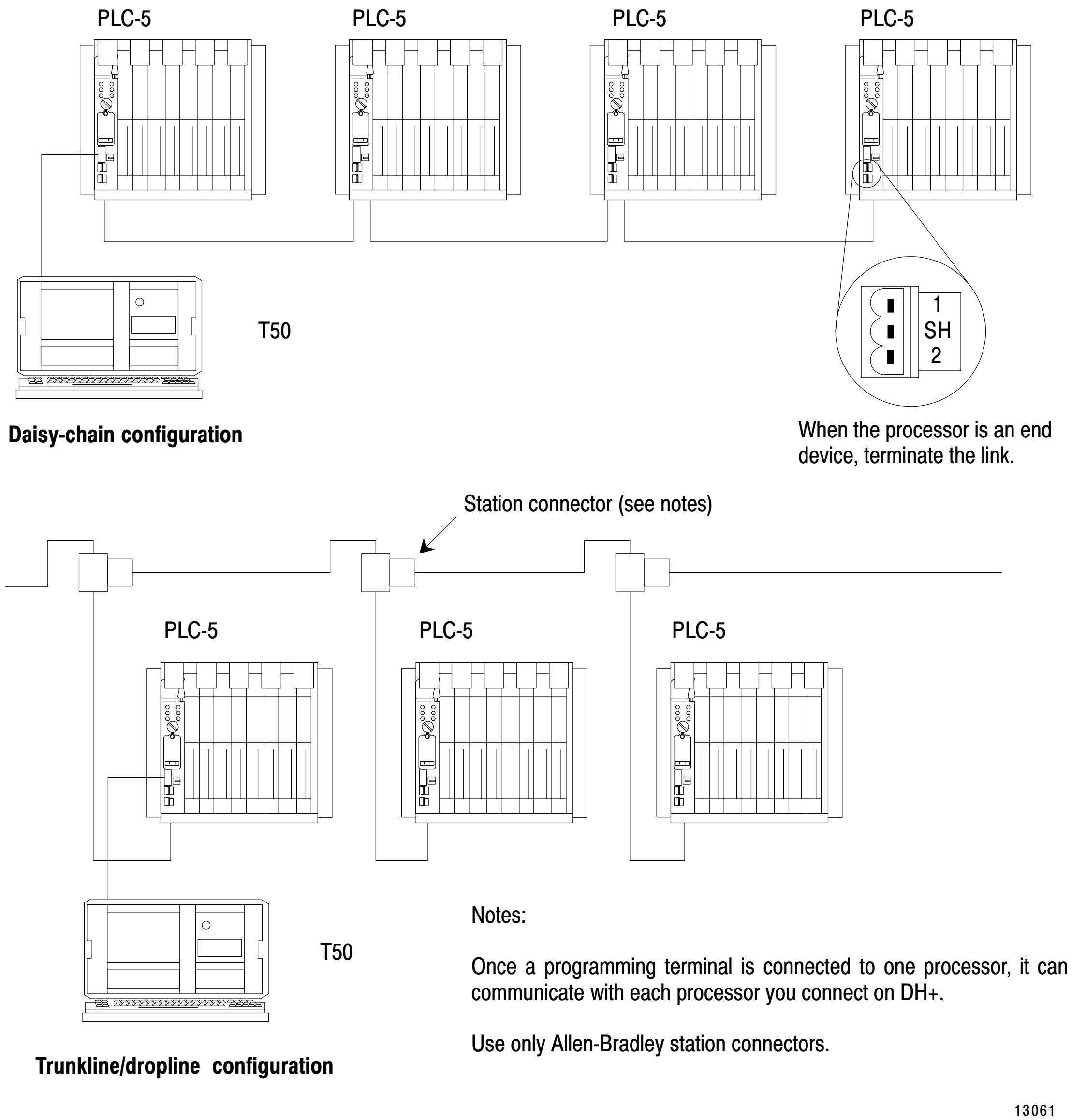
Connecting Devices to DH+ Link

You can connect devices on a DH+ link with:

- daisy-chain connection
- trunkline/dropline connection

See Figure 5.7. Also, see the Data Highway and Data Highway Plus Cable Guide, publication 1770-6.2.1, for complete network wiring instructions.

Figure 5.7
Examples of DH+ Link Connections (Daisy-Chain and Trunkline/Dropline)



The PLC-5 processor has two connectors that are electrically identical. Connection to either one provides the same communication link. These connectors are:

- 9-pin D-shell DH+ COMM INTFC connector
- 3-pin DH+ COMM INTFC connector

Connecting a DH+ Link to Data Highway

You can connect DH+ links to Data Highway via a communication interface such as the 1785-KA module. The 1785-KA module allows nodes on a DH+ link to communicate with nodes on Data Highway or on another DH+ link.

See your local Allen-Bradley sales office or distributor for more information on connecting DH+ to Data Highway. Also, see the Data Highway/Data Highway Plus Protocol and Command Set, publication 1770-6.5.16, for more information.

Choosing Programming Terminal Connection

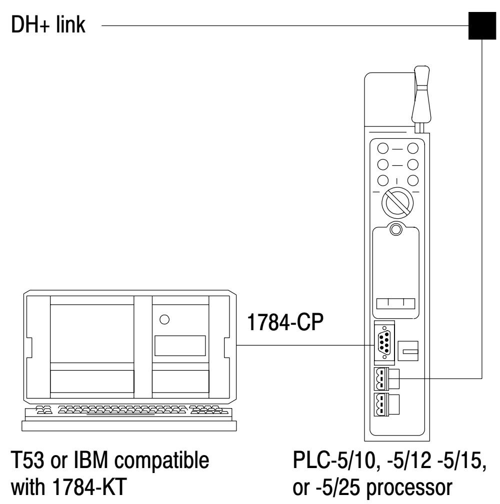
You can connect your programming terminal to a PLC-5 processor in several ways:

- direct connect to the DH+ link
- remote connection (DH+ to Data Highway to DH+)
- serial connections

Direct Connect to DH+ Link

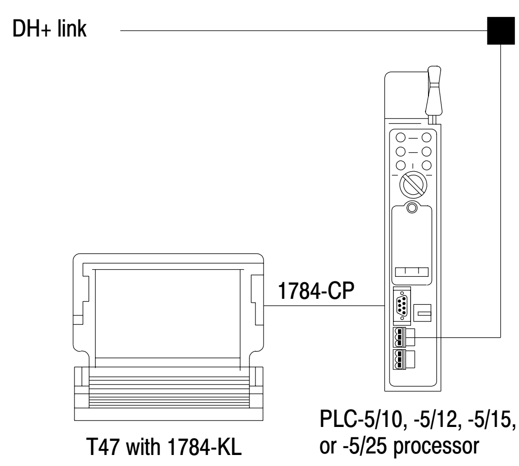
Use a 1784-KT to connect a T53 or IBM-compatible programming terminal directly to a processor or to a DH+ link that connects processors (Figure 5.8).

Figure 5.8
Connection to DH+ Link through 1784-KT Communication Interface Module



Use a 1784-KL/B to connect a T47 programming terminal directly to a processor or to a DH+ link that connects processors (Figure 5.9).

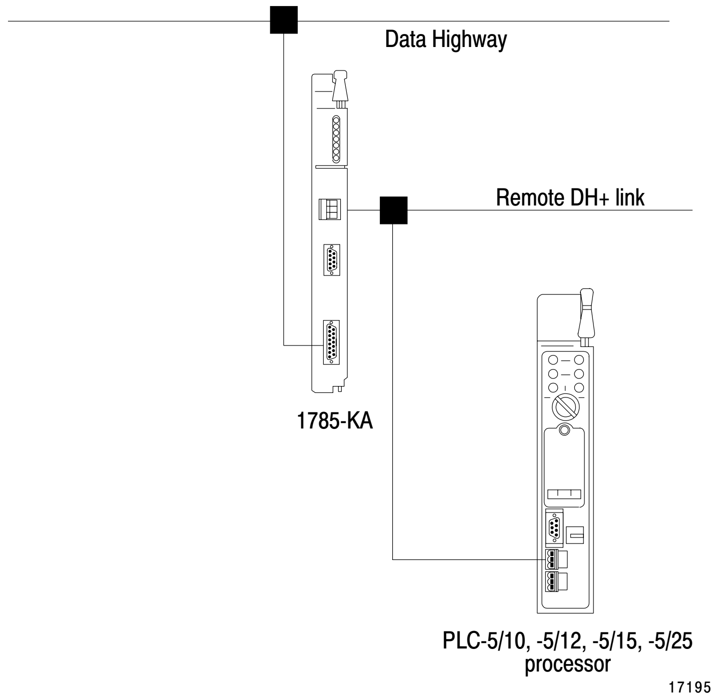
Figure 5.9
Connection to DH+ Link through 1784-KL Communication Interface Module



Remote Connection

The remote programming configurations available with the 1784-KT, 1784-KT2, and 1784-KL boards provide you communication with processors on other DH+ links in the network to expand the range of processors that you can use for program development (Figure 5.10).

Figure 5.10
Example DH+ to Data Highway to DH+ Link Configuration



Serial Connections

You can connect a programming terminal to a PLC-5/10, -5/12, -5/15, or -5/25 processor through a serial port (COM1 or COM2) on the terminal with one of the following communication modules:

- 1785-KE Series A or B Communication Interface Module
  (resides in a 1771 I/O rack)
- 1770-KF2, Series B Communication Interface Module
  (desktop unit as shown in Figure 5.12)

Important: The communication driver is interrupt-driven; the serial port must support hardware interrupts. On most machines, COM1 and COM2 support these interrupts.

Figure 5.11
1785-KE (Series B) Connection through an RS-232-C Serial Port

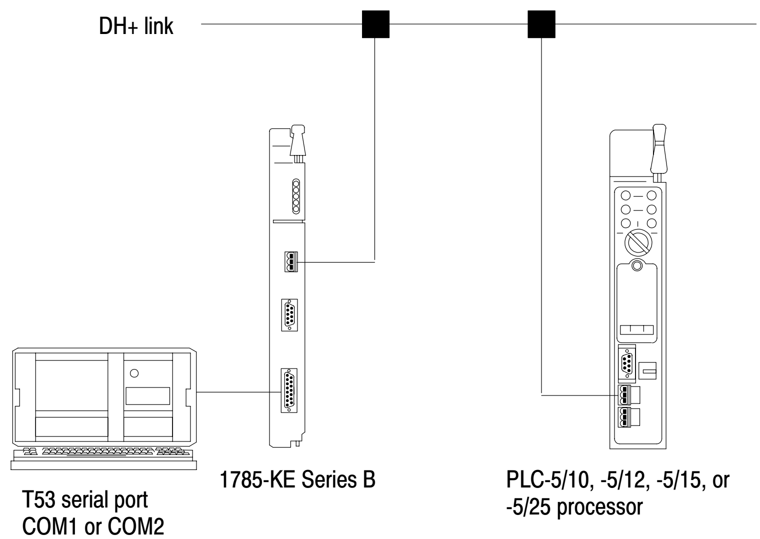
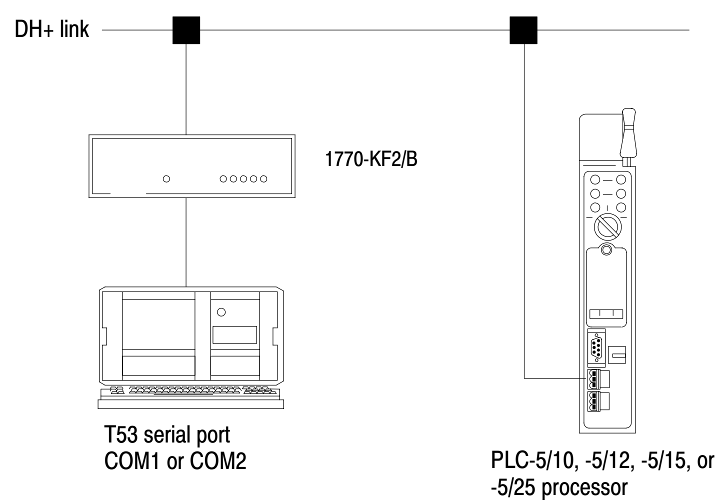


Figure 5.12
1770-KF2/B Connection through an RS-232-C Serial Port

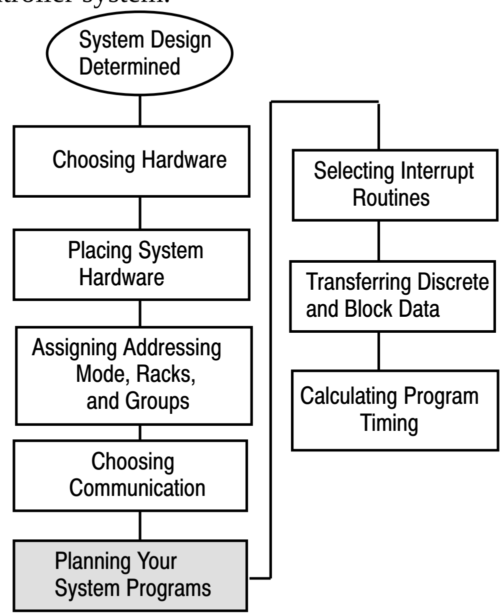


Chapter 6

Planning Your System Programs

Chapter Objectives

This chapter covers basic programming considerations for planning a Classic PLC-5 programmable controller system.



See your programming software documentation for a discussion of the instructions used in ladder-logic programming.

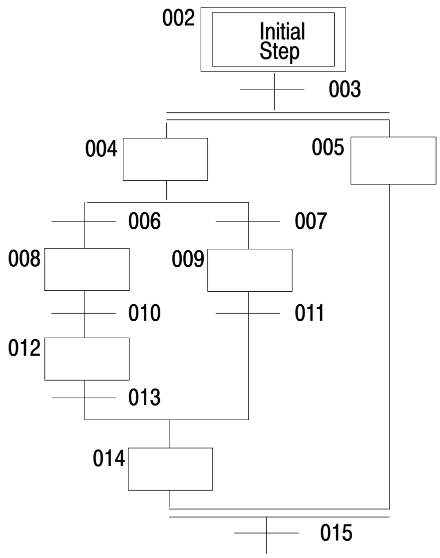
Planning Application Programs

Use the functional specification that you previously developed to define your programming application. The specification is a conceptual view of your application and is used to determine your main program, sequential function chart (SFC), and logic requirements.

In planning and developing the programs for your application, we recommend that you use the program-development model shown in chapter 1, "Understanding Your System."

Using SFCs with PLC-5 Processors

Use SFCs as a sequence-control language by which you can control and display the state of a control process. Instead of one long program for your application, divide the logic into steps and transitions. The display of these steps and transitions lets the user see what state the machine process is in at a given time.



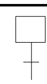
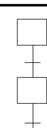
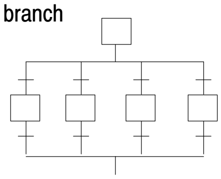
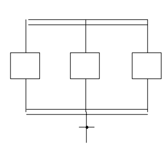
Each step corresponds to a control task (displayed as a box); each step is related to a program file that contains the logic for the associated control task. Each transition (displayed as a horizontal line) examines conditions, specified in an associated program file, that determines when the processor can continue to the next task.

Deciding How to Use an SFC

After you identify the major areas of machine operation, convert the logical paths and steps that you labeled in your design specification to SFC building blocks. Table 6.A helps explain when to use which SFC building blocks.

Important: At this point, do not worry about the actual logic for each step and transition. After you complete the SFC, you can develop the logic.

Table 6.A
Deciding When to Use the SFC Structures

If you have:	Then draw:	Using these rules:
An independent machine state	A step with its transition 	A step must always be followed by a transition.
A clearly defined chain of events that occur sequentially For example, in one heat-treating area, the temperature must ramp up at a particular rate; maintain the temperature for a certain duration, then cool at a particular rate.	A simple path of steps and transitions 	For design purposes, number steps and transitions consecutively from 2. Start the path with a step; end the path with a transition.
Two or more alternative paths where only one is selected For example, depending on a build code, one station must either drill or polish.	A selection branch 	The transitions beginning each path are scanned from left to right. The first true transition determines the path taken.
Two or more parallel paths that must be scanned simultaneously at least once For example, communications and block transfers must occur while control logic is executing.	A simultaneous branch 	All paths are active in the structure. You can define up to 7 parallel paths.

Application Example for SFCs

For typical SFC applications, an SFC program controls the order of events in your process by issuing commands. A command, such as `fwdcyr_cmd` to move a conveyor forward, is simply a data table storage bit (for example B3:0/7) that you set up in the SFC. You then program the logic for `fwdcyr_cmd` in a separate ladder program to control the actual outputs to move the conveyor.

You can have only one main program file, which is either an SFC or a ladder-logic program. You enter the programs into your computer using the SFC or ladder editor. For more information on entering SFCs or ladder logic, see your programming software documentation set.

Programming Considerations for SFCs

Use the information in Table 6.B for SFC rules for special programming.

Table 6.B
SFC Rules for Special Programming Considerations

If you have:	Use these rules:
To jump within the SFC	Use a GOTO statement and label.
A step that needs to be run in multiple places within the SFC	Repeat the step where needed or use a global subroutine that gets called from multiple steps.
A step that can be ignored based on logic conditions	Create two selection branches, one with and one without the step; or place the step in a subroutine; or combine the step with another step that is segregated by an MCR zone.
An SFC branch structure within another branch structure (nesting)	Nest the branch structures. The software supports as many levels of nested branches as you can store based on processor memory.
A mini-SFC (compressed steps) within the main SFC	Create an SFC macro. A macro begins a with a step; the transition for the ending step follows the macro.
To reset the logic in an SFC program	Set the SFR instruction to reset the chart.
To disable an MCP	Set the disable bit on the Processor Configuration screen.
See your programming software documentation for further information on any of the techniques listed in this table.	

Preparing the Programs for Your Application

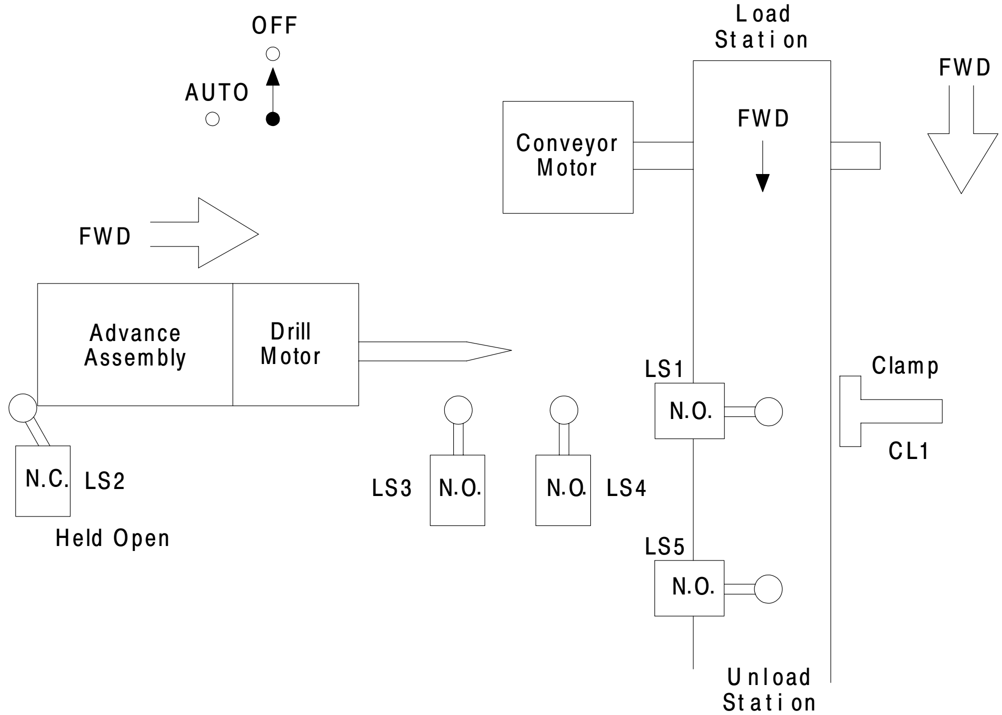
This section uses a drill-machine application example. Information on the program entry phase is in the programming software documentation set.

You can use only one main program; but you can still apply some of the steps by incorporating them into your main SFC and supporting ladder programs.

Organizing a Machine Example

This section uses an example of a specific machine operation to show how to identify conditions and actions and how to group the actions into steps of machine operation.

Figure 6.1
Hardware Block Diagram and Description of Machine Process

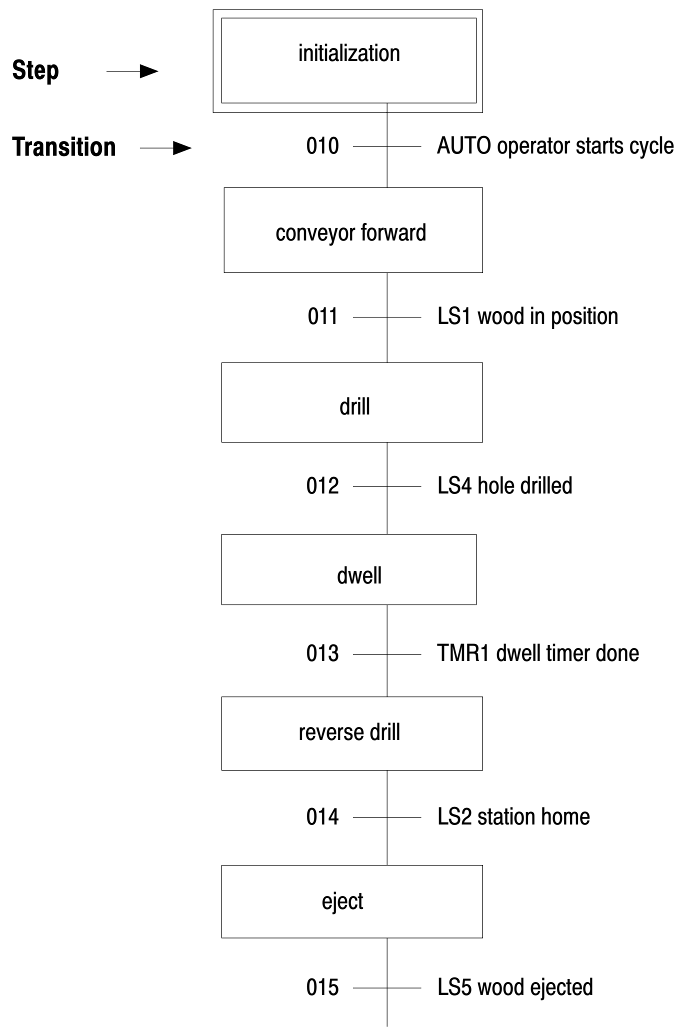


A description of this operation might be as follows:

1. The operator starts the conveyor by selecting AUTO.
2. The operator puts a block of wood onto the conveyor.
3. The wood moves into position and actuates LS1.
4. When the wood is in position:
   a. the conveyor stops
   b. CL1 clamps the wood
   c. the drill station moves forward
5. The drill station moves forward and closes LS3. This action turns on the drill motor.
6. The drill station moves to full depth and closes LS4. This action:
   a. stops forward motion of the drill station
   b. initiates a 2-second dwell
7. The drill station backs up after the 2-second dwell.
8. The drill motor stops when LS3 is released.
9. The drill station reaches home position and opens LS2. This action:
   a. stops the reverse motion
   b. opens the clamp
   c. starts the conveyor forward
10. The wood is ejected when LS5 toggles to indicate that the cycle is complete.

We recommend that you then create a rough-draft SFC to represent the operation (see Figure 6.2).

Figure 6.2
Drill Machine Example Functional Specification



Creating the Detailed Analysis for Your Functional Specification

Begin determining the details of your process as discussed in chapter 1, “Understanding Your System.” Identify the hardware requirements. Table 6.C identifies hardware requirements for the inputs and outputs of the drill machine.

Table 6.C
Hardware Requirements for the Inputs and Output of the Drill Example

Input		Part	Description
AUTO		selector switch	select automatic mode
LS1	N.O.	limit switch	part in place
LS2	N.C.	limit switch	drill station home
LS3	N.O.	limit switch	drill motor on
LS4	N.O.	limit switch	drill station at full depth
LS5	N.O.	limit switch	cycle complete
DSF		drive motor	move drill station forward
DSB		drive motor	move drill station back
DM		drill motor	drill motor on
CL1		electric clamp	clamp 1 on
CMF		drive motor	move conveyor forward
TMR1		timer	dwell timer

Use the hardware requirements (with the functional specification) to match the inputs and outputs with the actions of the process. Table 6.D shows the hardware requirements with the general description of the drill machine example.

Table 6.D
List of Conditions and Actions for the Drill Example

When this happens:	This happens:	
AUTO switch closes	Conveyor moves forward	(CMF = on)
LS1 closes	Conveyor stops Clamp holds wood Drill station advances	(CMF = off) (CL1 = on) (DSF = on)
LS3 closes	Drill motor starts	(DM = on)
LS4 closes	Drill station stops Dwell timer starts	(DSF = off) (TMR1 = on)
Timer done	Drill station backs up	(DSB = on)
LS3 opens	Drill motor stop	(DM = off)
LS2 opens	Drill station stops Clamp releases wood Conveyor starts	(DSB = off) (CL1 = off) (CMF = on)
LS5 closes	Wood is ejected	

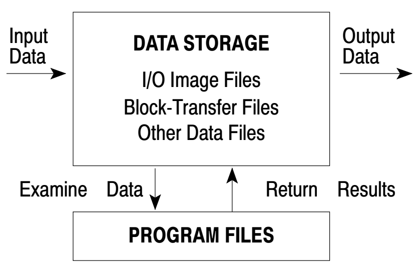
Once you identify the individual actions, you can add these actions to your plan to complete your program. Once you have an SFC program that defines the individual machine actions for your process, you can create a ladder-logic program that controls the outputs of those machine actions. It does not matter in what order you program these rungs. This program merely contains the ladder logic that defines a command for each machine action in your process.

Program Entry

When you finish your detailed analysis, you have your main program planned. Now, enter your program into your terminal.

Addressing Data Table Files

PLC-5 memory is divided into two areas: data and program-file storage.



Areas of Storage	Description
Data	All of the data the processor examines or changes is stored in files in data storage areas of memory. These storage areas store: • Data received from input modules • Data to be sent to output modules; this data represents decisions made by the logic • Intermediate results made by the logic • Preloaded data such as presets and recipes • Control instructions • System status
Program Files	You create files for program logic, depending on the method you are using: ladder logic, sequential function charts, and/or structured text. These files contain the instructions to examine inputs and outputs and return results.

Data Table Memory

You can address data files in different formats when you write your programs. Refer to Table 6.E for valid data table file-type specifications.

Table 6.E
Data Table Memory Usage

File Type	File Type Identifier	File Number	Memory Used in Over-head for Each File (16-bit words)	Memory Used (16-bit words) per Word, Float Word, Character, or Structure
Output Image	O	0	2	1/word
Input Image	I	1	2	1/word
Status	S	2	2	1/word
Bit (binary)	B	3	2	1/word
Timer	T	4[1]	2	3/structure
Counter	C	5[1]	2	3/structure
Control	R	6[1]	2	3/structure
Integer	N	7[1]	2	1/word
Floating-Point	F	8[1]	2	2/float word
ASCII	A	3-999	2	$^1/_2$ per character
BCD	D	3-999	2	1/word
Undefined	--	9-999	2	0

[1] This is the default file number. For this file type, you can assign any file number from 3 thru 999.

Data table files are contiguous in memory. Size in words for I/O files 0 and 1 are:

For this processor:	Files O0 and I1 memory size:
PLC-5/10, -5/12, -5/15	Is fixed at 32 words
PLC-5/25	Varies from 32-64 words (32 is the default)

Status file 2 is fixed at 32 words for each processor. Files 3-999 vary in size. These files contain only the number of words corresponding to the highest address that you assign. Each B, N, A, and D file can be 1,000 words maximum. Each F file can be 1,000 float words (32-bit words) maximum. Each T, C, R, and SC file can be 1,000 structures maximum.

Data Table Addressing Formats

Address Type	Description	Example
Logical address	Alpha-numeric coded format to specify the data location	N23:0 addresses an integer file 23, word 0
I/O image address	Logical address format, but relates physical locations in the I/O chassis to memory locations in the I/O image file	I:017/17 addresses input file word 017 (octal), bit 17 (octal), which corresponds to rack 01, module group 7, and terminal 17
Indirect address	Logical address format, but allows you to change address values in the base address with your ladder logic program	N[N7:6]:0 has the file number as the variable The file number is stored in integer file 7, word 6
Indexed address	Index prefix (#) is followed by a logical address format, but it adds an index value (offset) from processor status file to the base address	When #N23:0 is the indexed address and the offset value stored in the processor status file is 10, then • the base address is integer file 23, word 0 • and the offset address is integer file 23, word 10
Symbolic address	ASCII character string that relates the address (file, structure, word, or bit) to a descriptive, meaningful name that you assign	For example, a floating point address F10:0 could be given a symbolic address of Calc_1. These symbols are a feature of the programming software and not of the processor. Guidelines for setting up an address are as follows: • Start the name with an alphabetic character. • The symbol must begin with a letter and can have up to 10 of the following characters: A-Z (upper and lower case), 0-9, underscore (_) and @. • You can substitute a symbolic address for structure, word, or bit addresses. • Record the symbols you define and their corresponding logical addresses.

Using the Processor Status File

Use the Processor Status screen to monitor:

- processor status information
- major and minor faults
- STIs
- program scan times
- I/O status

Processor status data is stored in status file S2. See Table 6.F.

Table 6.F
Processor Status File Addresses

This word of the status file:		Stores:
S:0		Arithmetic flags • bit 0= carry • bit 1 = overflow • bit 2= zero • bit 3 = sign
S:1		Processor status and flags
S:2		Switch settings: • bits 0 - 5 = DH+ station # • bit 7 = set is scanner; reset is adapter (PLC-5/15, -5/25 only) • bit 11, 12 = HW addressing bit 12 bit 11 0 0 illegal 1 0 1/2-slot 0 1 1-slot 1 1 2-slot • bit 13, 14 = EEPROM bit 13 bit 14 0 0 EEPROM transfer if processor memory bad 0 1 EEPROM transfer disabled 1 1 EEPROM transfer at power-up • bit 15 = set is memory unprotected
S:3 to S:6		Active Node table: Word Bits DH+ Station # 3 0-15 00-17 4 0-15 20-37 5 0-15 40-57 6 0-15 60-77
S:8		Last program scan duration (in ms)
S:9		Maximum program scan duration (in ms)
S:10		Minor fault bits
S:11		Major fault bits
S:12		Fault code storage location
S:13		Program file where fault occurred
S:14		Rung number where fault occurred
S:16		I/O status file number storage location
S:18		Processor clock year
S:19		Processor clock month
S:20		Processor clock day
S:21		Processor clock hour
S:22		Processor clock minute
S:23		Processor clock second
S:24		Indexed addressing offset
S:25	(PLC-5/12, -5/15, -5/25 only)	I/O adapter image file
S:26		User control bits for processor start-up routine

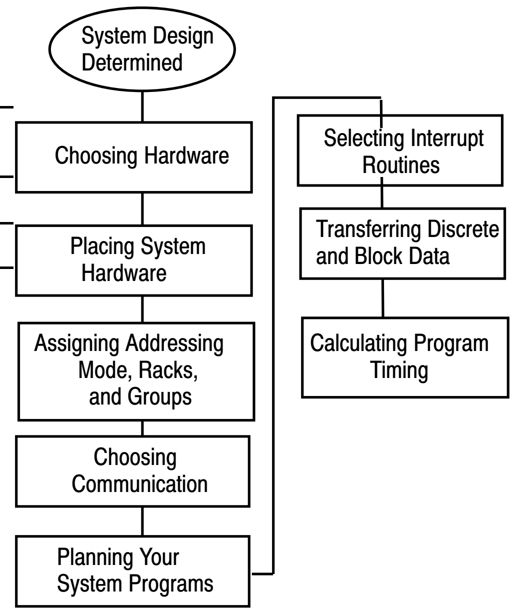
This word of the status file:		Stores:
S:28		Program watchdog setpoint (in ms)
S:29		Fault routine file
S:30		STI setpoint (in ms)
S:31		STI file number

Chapter 7

Selecting Interrupt Routines

Chapter Objectives

This chapter covers interrupt routines that you can choose to include when you program your system.



Using Programming Features

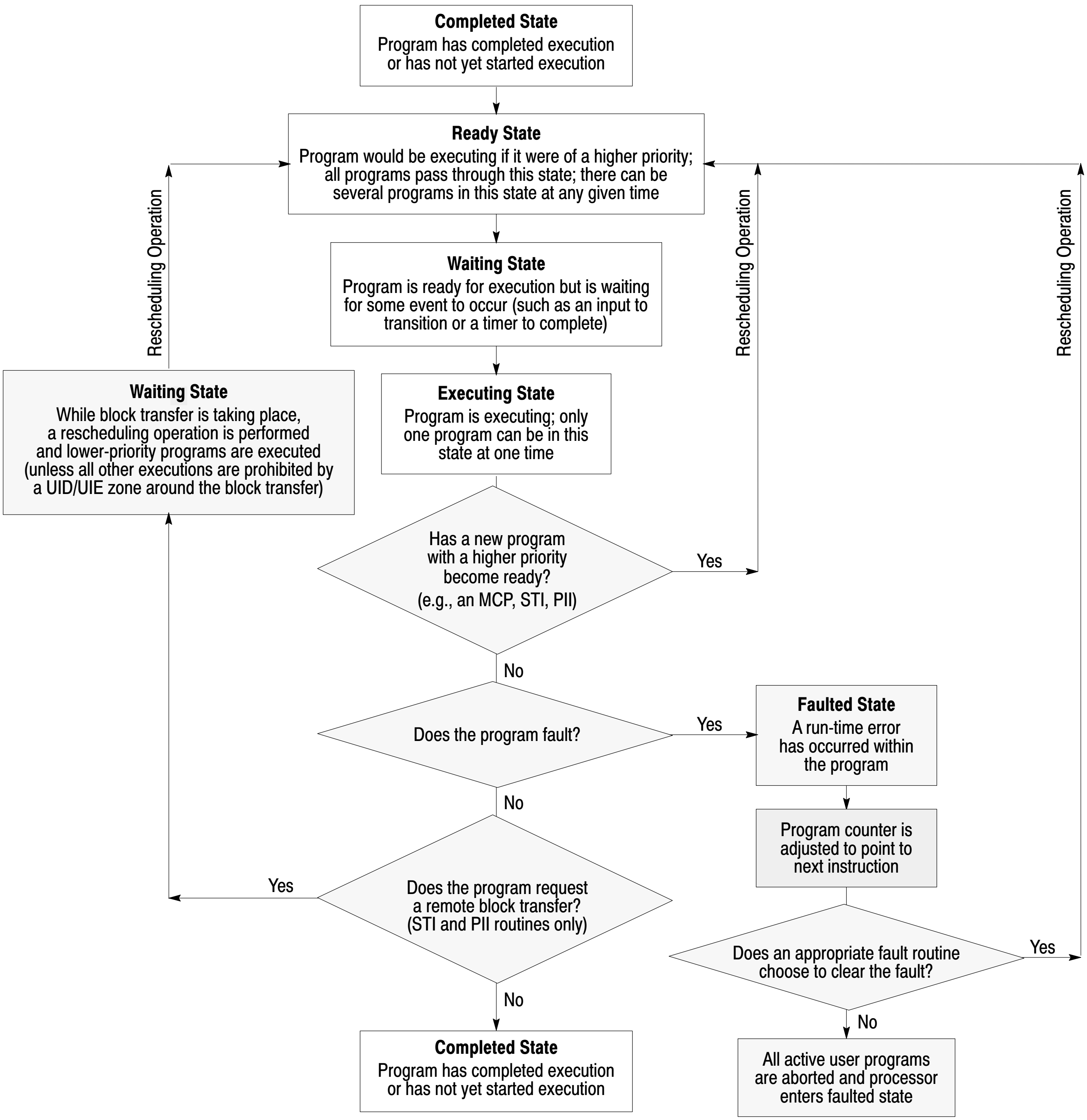
Use your design specification to determine if you need one or more of the following programming features:

- program execution control
- power-up routines

If a portion of logic should execute:	Example:	Use:	By doing the following:
Immediately on detecting conditions that require a startup	Restart the system after the system has been shut down	Powerup/Fault Routine	Create a separate file for a controlled start-up procedure for the first time that you start a program or when you start a program after system down time. The processor executes the power-up/fault routine to completion.
Immediately on detecting a major fault	Send critical status to a supervisory processor via DH+ after detecting a major fault	Fault Routine	Create a separate file for a controlled response to a major fault. The first fault detected determines which fault routine is executed. The processor executes the fault routine to completion. If the routine clears the fault, the processor resumes the main logic program where it was interrupted. If not, the processor faults and switches to program mode.

Program Execution States

User programs in the Classic PLC-5 processor are always in one of the following five states: completed, ready, executing, waiting, or faulted.



Writing a Fault Routine

You can write a fault routine that the processor runs when it detects a major fault. For example, if your program file becomes corrupted, you can tell the processor to interrupt the current program, run your fault routine and then continue processing the original program.

This section shows you how to set and write a fault routine and how to protect your processor from powering up in run mode after a power loss.

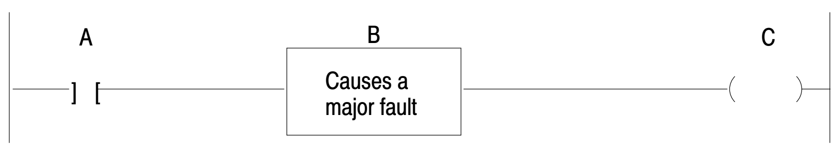
Responses to a Major Fault

When the processor detects a major fault, the processor immediately interrupts the current program. If a fault routine exists (i.e., specified in S:29 as a fault routine), the processor runs that fault routine program for recoverable faults. Then, depending on the type of fault, the processor:

- returns to the current ladder program file if the processor can recover from the fault
- enters fault mode if the processor cannot recover from the fault

For example, the rung in Figure 7.3 includes an instruction that causes a major fault.

Figure 7.3
Sample Ladder Logic for a Fault



In the example in Figure 7.3, the processor runs the fault routine after detecting the fault. If the fault routine resets the faulted bits, the processor returns to the next instruction in the program file that follows the one that faulted and outputs on the remainder of the rung.

If you do not program a fault routine for fault B, the processor immediately faults.

The bits in word 11 of the processor status file indicate the type of major fault. See Table 7.G to determine whether a fault is recoverable.

Table 7.G
Response to Major Faults (Word 11 of the Status File)

This bit:	Indicates this fault:	And the fault is:
00	Corrupted program file	Recoverable—the fault routine can instruct the processor to clear the fault and then resume scanning the program.
01	Corrupted address in ladder program (see fault codes 10-19)	
02	Programming error (see fault codes 20-29)	
05	Start-up protection fault (see word 26, bit 1) Processor sets bit 5; if your fault routine does not reset this bit, the processor inhibits startup	
07	User-generated fault; processor jumped to fault routine (see fault codes 0-9)	
08	Watchdog faulted	
13	STI file does not contain ladder logic or does not exist	
03	Processor detected an SFC fault (see fault codes 74-79)	Not recoverable—the processor enters fault mode without scanning the fault routine.
04	Processor detected an error when assembling a ladder program file (see fault code 70)	
09	System is configured wrong; you installed a RAM cartridge but configured the system for an EEPROM or you violated 32-point I/O module placement rules for 1-slot addressing	
10	Non-recoverable hardware error	
14	Fault routine does not contain ladder logic or does not exist	
15	Fault routine program file does not contain ladder logic	

A remote block transfer from a fault routine causes the processor to stop scanning all programs until the block transfer completes.

Major Fault Codes

Table 7.H lists major fault codes. The processor stores the fault code in word 12 of the processor status file.

Table 7.H
Major Fault Codes

Code	Fault
00-09	Reserved for user-defined fault codes
12	Bad integer operand type, restore new processor memory file
13	Bad mixed mode operation type, restore new processor memory file
14	Not enough operands for instruction, restore new processor memory file
15	Too many operands for instructions, restore new processor memory file
16	Corrupted instruction, probably due to restoring an incompatible processor memory file
17	Can't find expression end; restore new processor memory file
18	Missing end of edit zone; restore new processor memory file
20	You entered too large an element number in an indirect address
21	You entered a negative element number in an indirect address
22	You tried to access an undefined program file
23	You used a negative file number, you used a file number greater than the number of existing files, or you tried to indirectly address files 0, 1, or 2
24	You tried to indirectly address a file of the wrong type
30	You tried to jump to one too many nested subroutine files
31	You did not enter enough subroutine parameters
32	You jumped to an invalid (non-ladder) file
33 [1]	You entered a CAR routine file that is not 68000 code
34	You entered a negative preset or accumulated value in a timer instruction
35	You entered a negative time variable in a PID instruction
36	You entered an out-of-range setpoint in a PID instruction
37	You addressed an invalid module in a block transfer, immediate input, or immediate output instruction
38	You entered a return instruction from a non-subroutine file
39	FOR instruction with missing NXT
40	The control file is too small for the PID, BTR, BTW, or MSG instruction
41	NXT instruction with missing FOR
42	You tried to jump to a deleted label
44-69	Reserved
70	The processor detected duplicate labels
74	SFC file error detected
75	The SFC has too many active functions
77	SFC missing file or of wrong type for step, action, transition; or Subchart is created but empty; or SC or timer file specified in SFC empty or too small
78	The processor cannot continue to run the SFC after power loss
79	You tried to download an SFC to a processor that cannot run SFCs; or This specific PLC does not support this enhanced SFC
80	You incorrectly installed a 32-point I/O module in a 1-slot configuration (PLC-5/15, -5/25)
81	You illegally set an I/O chassis backplane switch; either switch 4 or 5 must be off

Important: If the PLC-5 processor detects a fault in the fault routine (double fault condition), the PLC-5 processor goes directly to fault mode without completing the fault routine.

Programming a Fault Routine

If you choose to program a fault routine, first have the fault routine examine the major fault information recorded by the PLC-5 processor and decide whether to do the following before the PLC-5 processor automatically goes to fault mode:

- set an alarm
- clear the fault
- shutdown in an orderly manner

On detecting a major fault, the PLC-5 processor immediately suspends the program file it was running and, if programmed, runs the fault routine file once to completion. If the PLC-5 processor does not run a fault routine, or the fault routine does not clear the fault, the PLC-5 processor automatically switches to fault mode.

Set an Alarm
You may need an alarm to signal when a major fault occurs. Put this rung first in your fault routine program

and combine it with a counter. You can also set an alarm in your fault routine to signal when the fault routine clears a major fault.

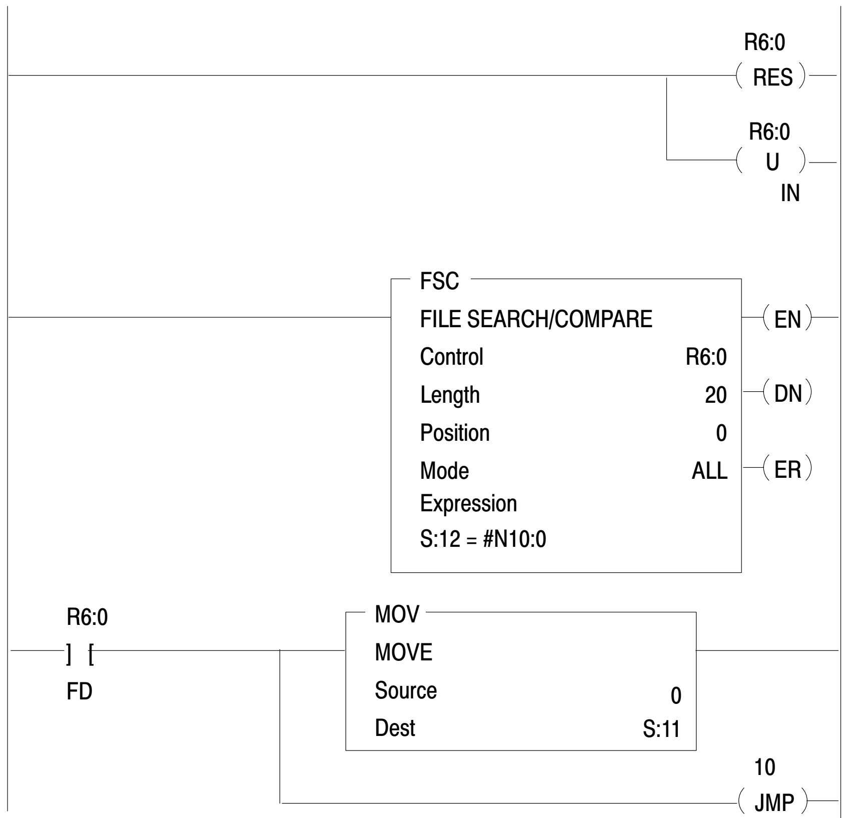
Clearing the Fault
If you decide to clear the fault in the fault routine, place the ladder logic for clearing the fault at the beginning of the fault routine. You can compare the fault code with a reference.

Compare fault code with a reference—Identify the possible major faults and then select only those your application will let you safely clear. These are your reference fault codes.

From the fault routine, examine the major fault code that the processor stores in S:12. Use an FSC instruction to compare the fault code to the reference file that contains “acceptable” fault codes (word-to-file comparison). If the processor finds a match, the FSC instruction sets the found (.FD) bit in the specified control structure. Use a MOV instruction to clear the fault in S:11. Then jump to the end of the fault routine to quickly complete running the fault routine.

In Figure 7.4, #N10:0 is the reference file.

Figure 7.4
Example of Comparing a Major Fault Code with a Reference



Last rung in fault routine

The processor completes the scan of the fault routine. If the routine clears S:11, the processor returns to the program file and resumes program execution. If the fault routine does not clear S:11, the processor executes the rest of the fault routine and goes into FAULTED mode.

Important: If the fault routine clears the major fault, the processor completes the fault routine and returns to the next instruction in the program file that follows the one that contained the faulted instruction. The remainder of the rung is executed. It appears that the fault never occurred. The fault routine execution continues until you correct the cause of the fault.

Using Shutdown Logic
Shutdown programming should include the following considerations.

- Store initial conditions and reset other data to achieve an orderly start-up later.
- Monitor the shutdown of critical outputs. Use looping if needed to extend the single fault routine scan time up to the limit of the processor watchdog timer so that your program can confirm that critical events took place.

Testing a Fault Routine

To test a fault routine, use a JSR instruction to jump to the fault routine. Send a fault code as the first input parameter of the JSR instruction. The processor stores the fault code in status word 12 and sets the corresponding bit in word 11.

You may detect and set your own faults using fault codes 0-9 or by using the processor-defined fault codes 10-87.

Setting Up a Fault Routine

You can write multiple fault routine programs and store them in multiple fault routine files, but the logic processor runs only one fault routine program when the PLC-5 processor detects a major fault. The number of the fault routine the PLC-5 processor runs is stored in word 29 of the processor status file. Typically, you enter a fault routine file number with the programming software and change the specified fault routine file from the ladder program.

To set up a fault routine, you need to:

- enable the fault routine by entering a fault routine file number in the status file
- create the program file and enter fault routine logic
- clear a major fault (other than by the fault routine)

Enabling a Fault Routine

To enable a fault routine, store the program file number (3-999) of the file that contains the fault routine logic in word 29 of the processor status file. When the processor encounters a major fault, the processor runs the fault routine logic to handle the fault.

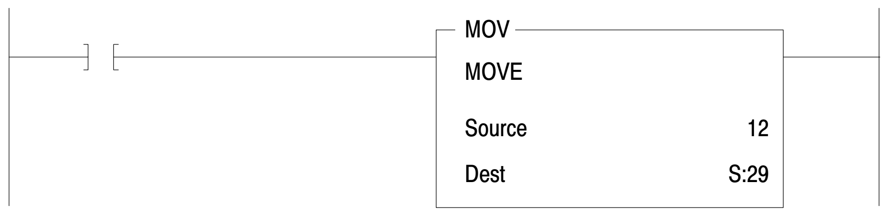
If you do not specify a program file number, the processor immediately enters fault mode after detecting a fault.

Changing the Fault Routine File Number from Ladder Logic

You can change the specified fault routine from ladder logic by copying a new fault routine file number into word 29 of the processor status file.

Figure 7.5 shows an example program for changing the fault routine file number.

Figure 7.5
Example of Changing the Fault Routine File Number



ATTENTION: Do not corrupt the program-file number of the fault routine or use the same file for any other purpose. If the file number that you specify results in a non-existent fault routine, the processor immediately enters fault mode after detecting a fault. Unexpected machine operation may result with damage to equipment and/or injury to personnel.

Clearing a Major Fault

You can clear a major fault with one of the following methods.

- Use the programming software to clear the major fault.

  For more information about using the programming software to clear major faults, see the chapter on clearing faults in the programming software documentation set.

- Turn the keyswitch on the PLC-5 processor from REM to PROG to RUN.

Important: Clearing a major fault does not correct the cause of the fault. The PLC-5 processor might continue to repeat the fault cycle until you correct the cause(s) for the major fault.

Setting Power-Up Protection

You can set your processor so that after a power loss the processor does not come up in run mode. Bit 1 in word 26 of the processor status file sets power-up protection. Table 7.I shows the states for this bit.

Table 7.I
Setting and Resetting the Power-Up Protection Bit

If word 26, bit 1 Is:	After power loss, the processor:
Set (1)	Scans the fault routine before returning to normal program scan
Reset (0)	Powers up directly at the first rung on the first program file

Set word 26, bit 1 manually from the processor status screen (see the chapter on using status data in programming software documentation). Or you can latch this bit through ladder logic. When set, the processor scans the fault routine once to completion after the processor recovers from a power loss. You can write the fault routine to determine whether or not the processor's current status permits the processor to respond correctly to ladder logic—i.e., whether to allow or inhibit the startup of the processor.

Allowing or Inhibiting Powerup

Bit 5 of status word 11 indicates whether or not you want to power up the processor after a loss of power. After a power loss, the processor automatically sets this bit; Table 7.J shows how you can change it from your fault routine.

Table 7.J
Setting and Resetting the Startup Bit

If the fault routine makes word 11, bit 5:	Then the processor:
Set (1)	Faults at the end of scanning the fault routine. Leave this bit set to inhibit startup.
Reset (0)	Resumes scanning the processor memory file. Reset this bit to allow startup

Important: You can use JMP and LBL instructions to scan only the portion of the fault routine associated with a particular fault or power-up condition.

For information about startup protection on SFCs, see the programming software documentation set.

Understanding Processor-Detected Major Faults

In general, if the processor detects a hardware fault, it sets a major fault and resets I/O. If the processor detects a run-time error, it sets a major fault bit and the remote I/O racks are set according to their last state switch. Module outputs in remote racks remain in their last state or they are de-energized, based on how you set the last state switch in the 1771 I/O chassis.

To decide how to set this switch, evaluate how the machines in your process will be affected by a fault. For example, how will the machine react to outputs remaining in their last state or to outputs being automatically de-energized? What is each output connected to? Will machine motion continue? Could this cause the control of your process to become unstable?

To set this switch, see the Classic 1785 PLC-5 Family Programmable Controllers Hardware Installation Manual, publication 1785-6.6.1.

Important: In the PLC-5 processor local chassis, outputs are reset—regardless of the last state switch setting—when one of the following occurs:

- processor detects a run-time error
- you set a status file bit to reset a local rack
- you select program or test mode

Fault in a Processor-Resident Local I/O Rack

The chassis that contains the Classic PLC-5 processor is the processor-resident local I/O chassis. If a problem occurs with the chassis backplane, the input and output data table bits for the resident local I/O rack are left in their last state. The processor sets a minor fault and continues scanning the program and controlling extended-local and remote I/O.

Your ladder program should monitor the I/O rack fault bits and take the appropriate recovery action (covered later in this section).

ATTENTION: If a resident local I/O rack fault occurs and you have no recovery methods, the input image table and outputs for the faulted rack remain in their last state. Potential personnel and machine damage may result.

Fault in a Remote I/O Chassis

In general, when a remote I/O chassis faults, the processor sets an I/O rack fault bit and then continues scanning the program and controlling the remaining I/O. The outputs in the faulted rack remain in their last state or they are de-energized, based on how you set the last state switch in the 1771 I/O chassis.

ATTENTION: If outputs are controlled by inputs in a different rack and a remote I/O rack fault occurs (in the inputs rack), the inputs are left in their last non-faulted state. The outputs may not be properly controlled and potential personnel and machine damage may result. Make sure that you have recovery methods.

Recovering from a Processor-Resident Local I/O or Remote I/O Rack Fault

In the PLC-5 processor, you can monitor I/O rack faults using processor status bits and then recover from the fault using a fault routine or ladder logic.

Using Status Bits to Monitor Rack Faults

There are two types of status bits used to display information about your I/O system: global status bits and I/O rack status bits.

The **global status bits** are set if a fault occurs in any one of the logical racks.

Processor	Possible Logical Rack Bits
PLC-5/10-5/12, or -5/15	4
PLC-5/25	8

Each bit represents an entire rack, no matter how many chassis make up a rack. (Remember that you can have up to four chassis configured as quarter racks to make up one logical rack.) These bits are stored in the lower eight bits of words 7, 32, and 34 of the status file.

For more information on these global status bits, see your programming software documentation set.

The **I/O rack status bits**, also known as the “partial rack status bits,” are used to monitor the racks in your I/O system. The software automatically creates an integer data file to store this information when an I/O status file is defined. This file contains 2 words of status bits for every rack configured in your system. The number of the data file that contains this I/O information is stored in word 16 (low byte) of the status file. You must enter this information on the processor status screen. For more information on monitoring I/O status with I/O rack status bits, see your programming software documentation set.

Using Fault Routine and Ladder Logic to Recover

You may want to configure a I/O rack fault as a minor fault if you have the appropriate fault routine and ladder logic to perform an orderly shutdown of the system. You can program ladder logic in several ways to recover from a I/O rack fault. These methods are:

- user-generated major fault
- reset input image table
- fault zone programming

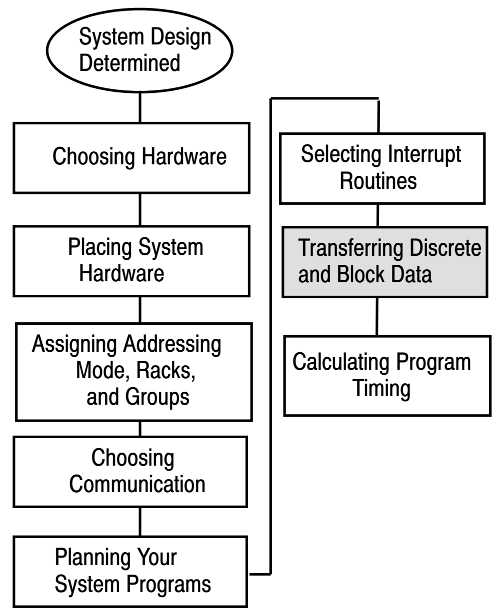
Methods:	Description:
User-generated major fault	You jump to a fault routine when a remote I/O rack fault occurs. In other words, if the status bits indicate a fault, you program the processor to act as if a major fault occurred (i.e., jump to the fault routine). You then program your fault routine to stop the process or perform an orderly shutdown of your system. When the processor executes the end-of-file instruction for the fault routine, a user-generated major fault is declared.
Reset input image table	You monitor the status bits and, if a fault is detected, you program the processor to act as if a minor fault occurred. After the status bits indicate a fault, use the I/O Status screen to inhibit the remote rack that faulted. You then use ladder logic to set or reset critical input image table bits according to the output requirements in the non-faulted rack. If you reset input image table bits, during the next I/O update, the input bits are set again to their last valid state. To prevent this from occurring, your program should set the inhibit bits for the faulted rack. The global inhibit bits control the input images on a rack-by-rack basis; the partial rack inhibit bits control the input images on a 1/4 rack basis. For more information on these bits, see the programming software documentation set. This method requires an extensive and careful review of your system for recovery operations. For more information on inhibiting I/O racks, see your programming software documentation set
Fault zone programming method	Using fault zone programming method, you disable sections of your program with MCR zones. Using the status bits, you monitor your racks; when a fault is detected, you control the program through the rungs in your MCR zone. With this method, outputs within the MCR zone must be non-retentive to be de-energized when a rack fault is detected. For more information, see your programming software documentation.

Chapter 8

Transferring Discrete and Block-Transfer Data

Chapter Objectives

This chapter covers discrete and block transfer of I/O data when a processor is configured for either adapter or scanner mode. Discrete-transfer data are words transferred to/from a digital discrete I/O module. Block-transfer data is transferred, in blocks of data of up to 64 words, to/from a block-transfer I/O module (such as an analog module).



Transferring Data Using Adapter Mode

You can transfer data in adapter mode in two ways.

If you want to transfer:	Use this method:
Words to/from a digital I/O module	Discrete-data transfer
Blocks of data (up to 64 words) to/from a block-transfer module (such as an analog module)	Block transfer

The processor transfers discrete and block I/O data in a similar way.

The adapter-mode processor and the supervisory processor automatically discrete transfer I/O data between themselves via the supervisory processor's remote I/O scan.

During each remote I/O scan:

- the supervisory processor transfers 2, 4, 6 or 8 words—depending on whether the adapter-mode processor is configured as a 1/4, 1/2, 3/4 or full rack
- the adapter-mode processor transfers 2, 4, 6 or 8 words—depending on whether the adapter-mode processor is configured as a 1/4, 1/2, 3/4 or full rack

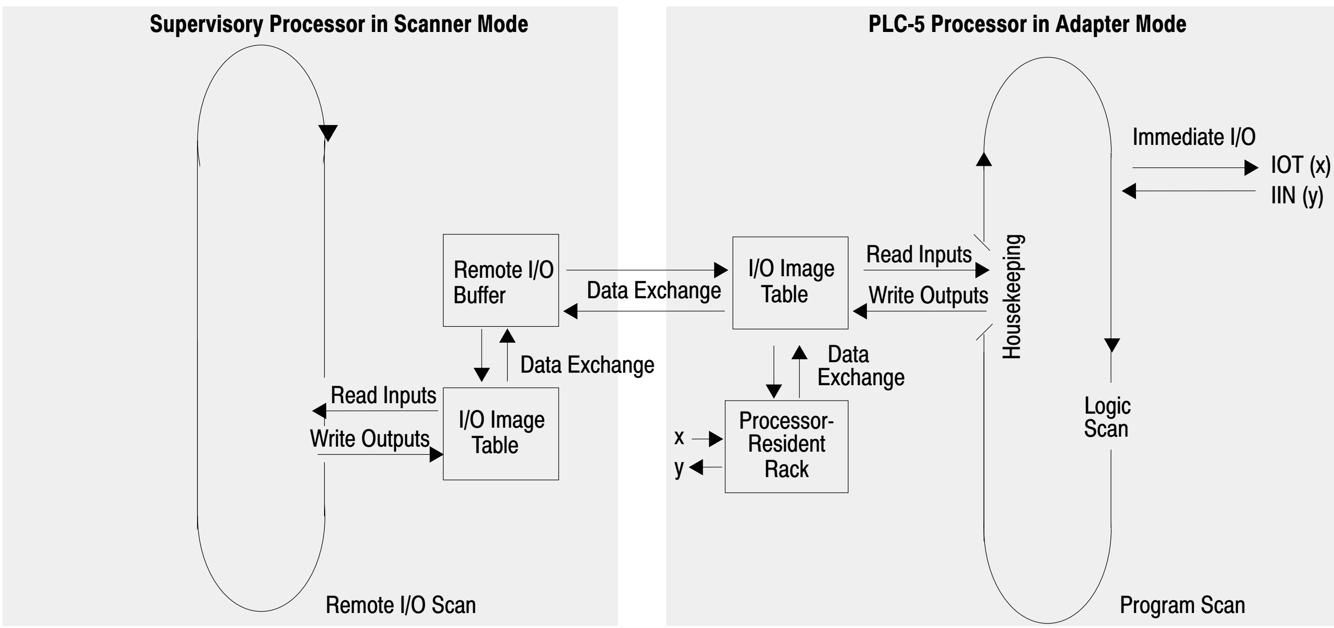
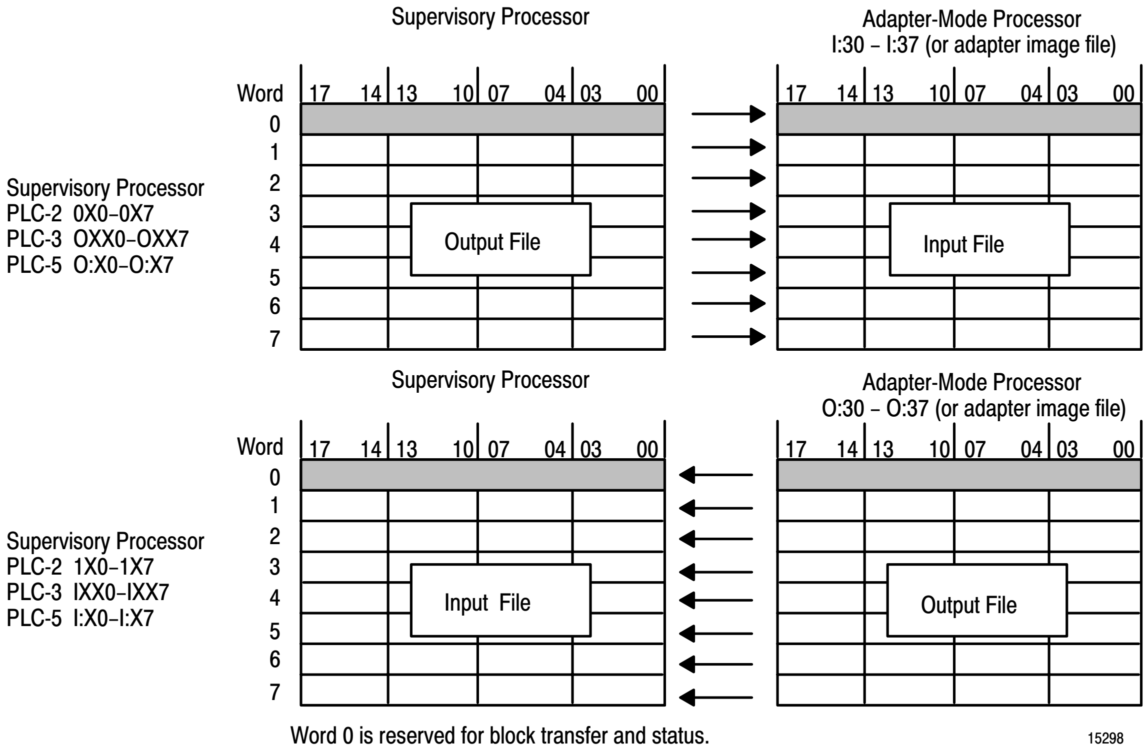


Figure 8.1 shows the transfers between supervisory processor output file and adapter-mode processor input file as well as between adapter-mode processor output file and supervisory processor input file.

Figure 8.1
Automatic I/O Transfer between Supervisory and Adapter-Mode Processors



If data from the supervisory processor is intended to control outputs of the adapter-mode processor, the ladder logic in the adapter-mode processor must move the data from its input file (I/O rack 3 or the adapter image file) to its output file (local I/O). Use XIC and OTE instructions for bit data; use move and copy instructions for word data.

If you want the supervisory processor to read data from a data file in the adapter-mode processor, ladder logic in the adapter-mode processor must move that data to its output file (I/O rack 3 or the adapter image file) for transfer to the supervisory processor.

Programming Discrete Transfer in Adapter Mode

For the supervisory processor, use the adapter's configured I/O rack number to receive data or store data for transfer.

Using Rack 3 (Addresses 0:30-0:37 and I:30-I:37)

Rack 3 is the default discrete-transfer file for PLC-5/12, -5/15, and -5/25 processors. Typically, each output instruction in one processor should have a corresponding input instruction in the other processor. The rack number determines the addresses you use.

- The ladder logic in the supervisory processor uses the rack number (0-76 octal) of the adapter-mode processor.
- Condition the ladder logic in the adapter processor with I30/10. When set, this bit indicates a communication failure between the adapter and supervisory processors.

Creating an Adapter Image File—PLC-5/12, -5/15, and -5/25 Processors

If you use 1/2-slot addressing in a 16-slot chassis, you need rack 3 addresses for scanning processor-resident local I/O on the adapter-mode processor. In this case, you can create an adapter image file for transferring data. Before you create an adapter image file, make sure that these conditions are true:

- the PLC-5 processor is in adapter mode
- the adapter-mode processor is in a 1771-A4B I/O chassis
- you are using 1/2-slot addressing
- you have not inhibited rack 3 by setting the rack inhibit bit 3 in processor status word 27

To create the adapter image file, create a 16-word integer file. This file must be 16 words regardless of whether you use 4-word or 8-word transfers. This file must be a unique integer file, for use only as an adapter image file. Words 0-7 are used for output; words 8-15 are used for input. Bits are numbered in decimal 0-15 for each word.

To tell the processor which file is the adapter image file, enter the file number in word 25 of the processor status file. You enter this file number on the processor status screen. For more information about the processor status screen, see the chapter on using status data in the programming software documentation.

Important: If you are using an adapter image file (instead of the rack 3 image), then you cannot use block transfers between the supervisor and the adapter-mode processor.

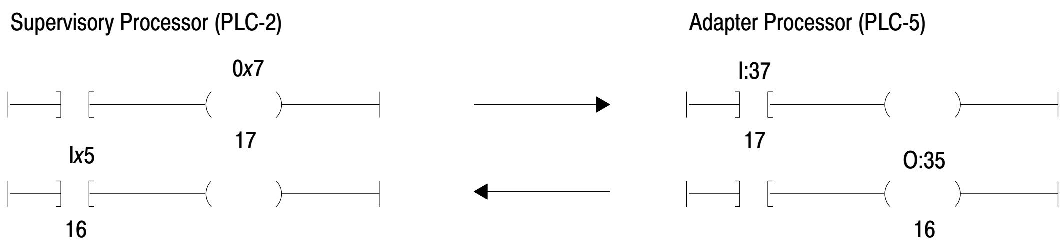
Condition the ladder logic in the adapter-mode processor with word 8, bit 8 decimal of the adapter image file. When set, this bit indicates a communication failure between the adapter and supervisory processors.

ATTENTION: Do not program block transfers to a supervisory processor if you create an adapter image file.

Transferring Bits between Supervisory and Adapter-Mode Processors

Figure 8.2 shows ladder logic for transferring bit 17 of the supervisory processor's output image word 7 and bit 16 of the adapter-mode processor's output image word 5. The *x* represents the adapter-mode processor's rack number; rack 3 is the simulated rack for the adapter-mode processor. This example assumes 1-slot or 2-slot hardware addressing.

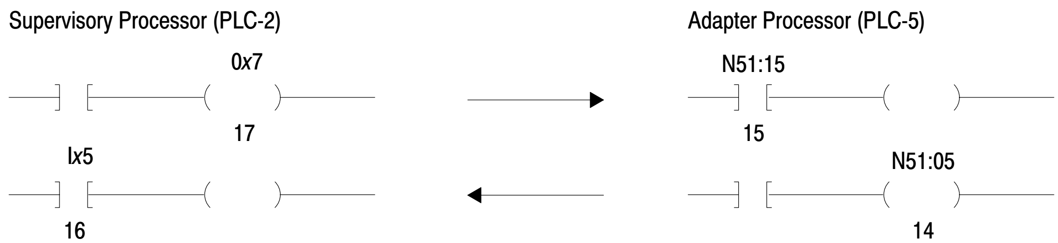
Figure 8.2
Transferring Bits Using Rack 3 in the Adapter-Mode Processor



When the supervisory processor sets its output file bit 0*x*:7/17, input file bit I:37/17 in the adapter-mode processor is automatically set. In the same way, when the adapter-mode processor sets output file bit O:35/16, input file bit I*x*:5/16 in the supervisory processor is automatically set.

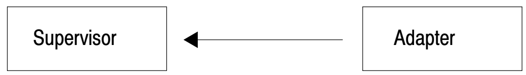
Figure 8.3 shows the ladder logic if you created an adapter image file because you need rack 3 addresses for local I/O. This example uses N51 as the adapter image file.

Figure 8.3
Transferring Bits Using Your Own Adapter Image File



For PLC-5/12, -5/15, and -5/25 processors, words 0-7 in the integer file represent output, words 8-15 represent input.

Determining the Status of the Adapter-Mode Processor



Adapter-mode processor sends to Supervisory processor

The supervisory processor receives these status bits (Table 8.K) from the adapter-mode processor in word 0 of the input file for the rack that the adapter-mode processor is emulating.

Table 8.K
Status Bits of the Adapter-Mode Processor

When this bit is set:	It indicates this condition:
10 octal	data not valid
15 octal	adapter-mode processor is in program or test mode

If you use an adapter image file in a PLC-5/12, -5/15, or -5/25 processor, then these status bits are not sent.

The supervisory processor should monitor the rack fault bits for the rack the adapter-mode processor is emulating to determine the status of the remote I/O link.

Determining the Status of the Supervisory Processor

Supervisor processor sends to adapter-mode processor

The adapter-mode processor receives these status bits (Table 8.L) from the supervisory processor in I:30 (or word 8 of the adapter image file) of the adapter-mode processor's data table. These bits tell the adapter-mode processor the status of the supervisory processor and the integrity of the remote I/O communication link.

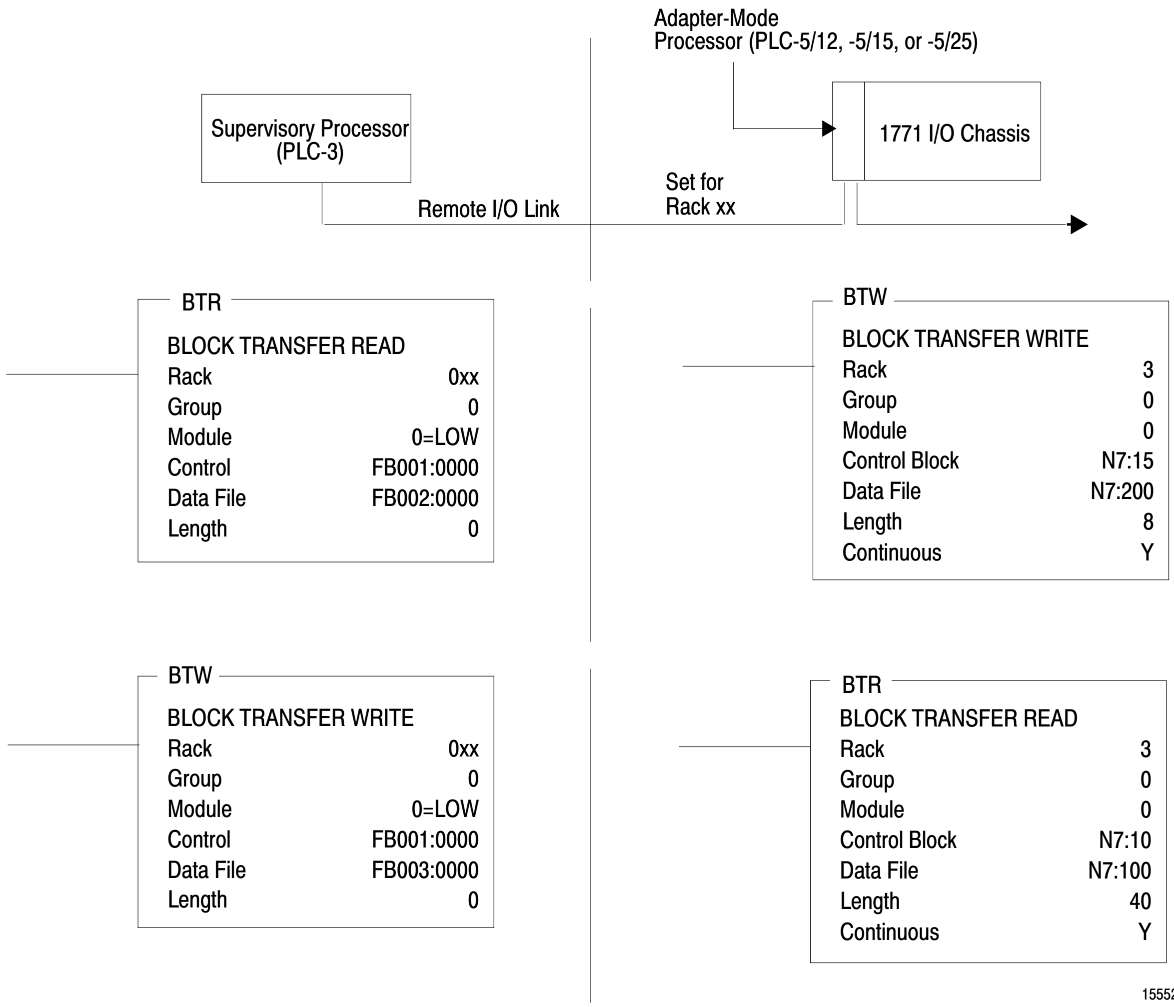
**Table 8.L
Status Bits of the Supervisory Processor Set in the Adapter-Mode Processor's Data Table**

When this bit is set:		It indicates that the adapter-mode processor:
Rack 3 Input Image Table (octal)	Adapter Input Image File (decimal)	
10	8	detected a communication failure or received a reset command from the supervisory processor
11	9	received a reset command from the supervisory processor (processor in program or test mode)
13	11	detected that the supervisory processor is powering up; this bit is reset with the first communication from the supervisory processor
15	13	detected a communication failure (e.g., no communication activity on the remote I/O communication link within the last 100ms)

Programming Block Transfer in Adapter Mode

To transfer blocks of data between a PLC-5/12, -5/15, or -5/25 adapter-mode processor and a supervisory processor, the adapter-mode processor must have a BTW to respond to the BTR from the supervisory processor (and a BTR to respond to the supervisory processor's BTW). For example, when the supervisory processor enables a BTR instruction, the adapter-mode processor responds by enabling a BTW instruction. The supervisory processor controls the transfer; the adapter-mode processor responds to the request. Figure 8.4 shows an example of block-transfer programming between an adapter-mode processor and a supervisory processor.

Figure 8.4
Example Adapter/Supervisor Block-Transfer Programming for a PLC-5/12, -5/15, or -5/25 Adapter-Mode Processor in Rack *xx*



Addressing Tips

Table 8.M lists some addressing tips for programming block transfers between a PLC-5/12, -5/15, or -5/25 adapter-mode processor and a supervisory processor.

Table 8.M
Addressing Tips for Adapter/Supervisor Block Transfers with a PLC-5/12, -5/15, or -5/25 Adapter-Mode Processor

BTR/BTW Parameter	BTR/BTW in Supervisor	BTR/BTW in Adapter
Rack	PLC-2/30: 1-7 octal PLC-3: 0-77 octal PLC-5/25: 1-7 octal	Must be 3[1]
Group	0	Must be 0
Module	0	
Length	Must be 0	Number of words transferred
Continuous	Yes (PLC-5 and PLC-5/250 only)	Yes

1 If you need rack 3 I/O for the adapter-mode processor's local I/O, then you must specify an adapter image file and block transfers can not be used between the supervisory and adapter-mode processor.

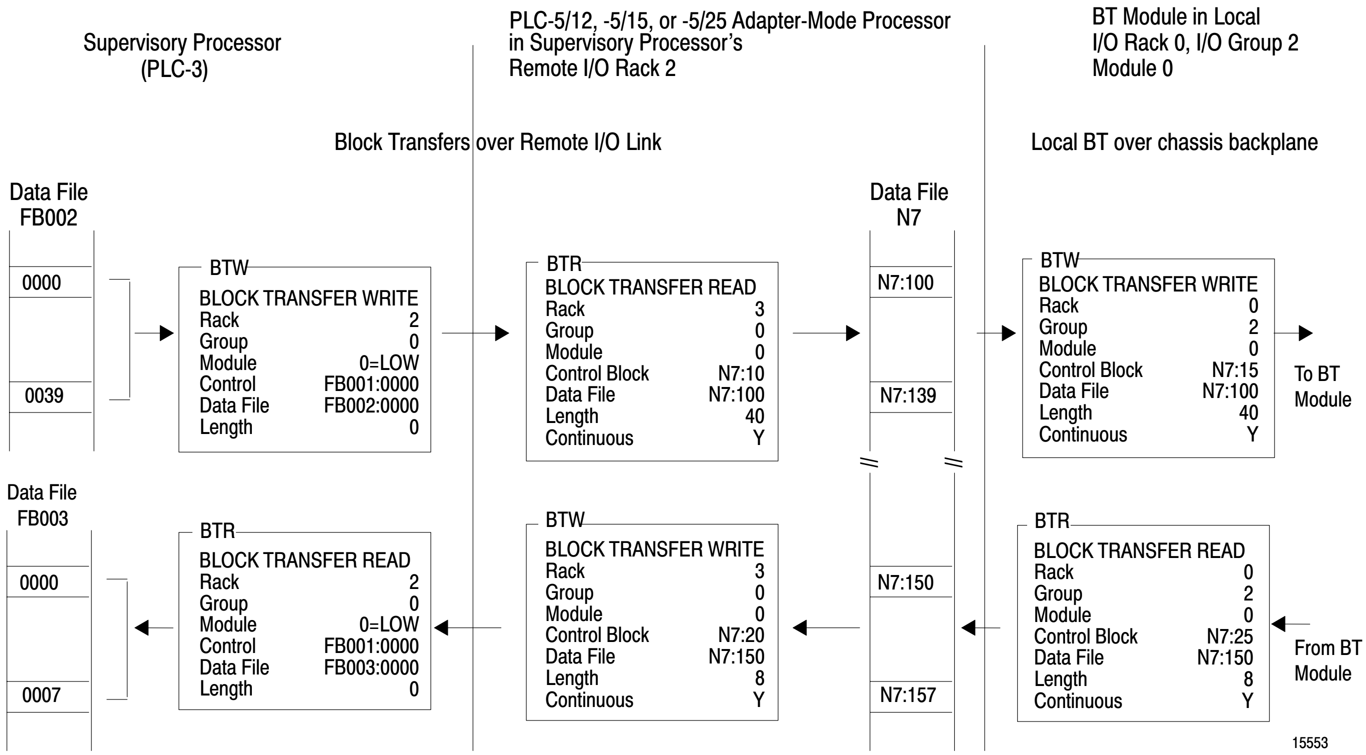
ATTENTION: To guarantee the correct destination of block-transfer data, program only one set of bidirectional block transfers between supervisory and adapter-mode processors with PLC-5/12, -5/15, and -5/25 processors in adapter mode.

Important: If you are using a PLC-5/12, -5/15, or -5/25 processor, set the supervisory processor's communication rate for remote I/O to 57.6 kbps.

Block transfers between adapter and supervisory processors transfer data between data table addresses. If you want to transfer processor-resident local I/O data of the adapter mode processor to a supervisory processor or if you want to transfer data from the supervisory processor to processor-resident local I/O of the adapter mode processor, you must use MOV or COP instructions within the adapter-mode processor to move the data in or out of the data file used in the adapter block-transfer instruction.

Figure 8.5 shows data transfers from a supervisory processor to a PLC-5/12, -5/15, or -5/25 adapter-mode processor to a local block-transfer module and vice versa.

Figure 8.5
Example Block Transfer from Supervisory Processor to a PLC-5/12, -5/15, or -5/25 Adapter-Mode Processor to Local Block-Transfer Module and Vice Versa



If you block transfer data with a supervisory processor, you cannot use 1/2-slot addressing with a 1771-A4B chassis because the adapter-mode processor needs the rack 3 I/O image table for block-transfer communication. (This applies only to PLC-5/12, -5/15, and -5/25 processors in adapter mode.)

ATTENTION: Do not try block transfers to a supervisory processor when the adapter-mode processor uses rack 3 for scanning processor-resident local I/O (when you create your own adapter image file using a PLC-5/12, -5/15, and -5/25 processor in adapter mode). Using rack 3 addresses under this condition will result in unpredictable machine operation with possible damage to equipment or personnel.

Example of Block-Transfer Ladder Logic

The following figures show example ladder logic for block transfers between an adapter-mode processor and a supervisory processor.

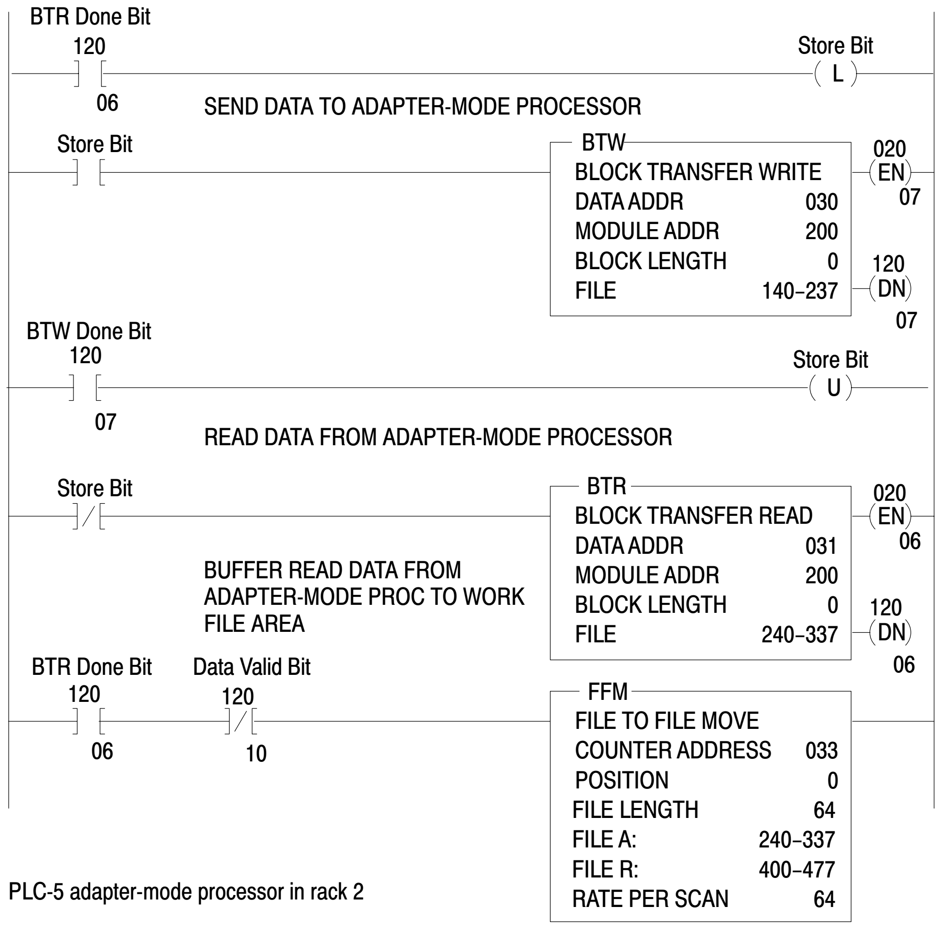
Supervisory Processor (PLC-2/30, PLC-3, PLC-5, or PLC-5/250)

Follow these guidelines when programming block-transfer instructions in the supervisory processor.

- Set the length to 0.
- Set the continuous bit for continuous operation (PLC-5 and -5/250 processors only).
- Use the remote I/O rack number for which you configure the adapter-mode processor.
- Use 0 for the group and module numbers.
- Condition the use of BTR data with a "data valid" bit.

All address comments for contacts shown in the following examples represent the set (1) state of the bit in the PLC-5 processor.

Figure 8.6
Example Block Transfer in PLC-2/30 Supervisory Processor



PLC-5 adapter-mode processor in rack 2

Figure 8.7
Example Block Transfer in PLC-3 Supervisory Processor

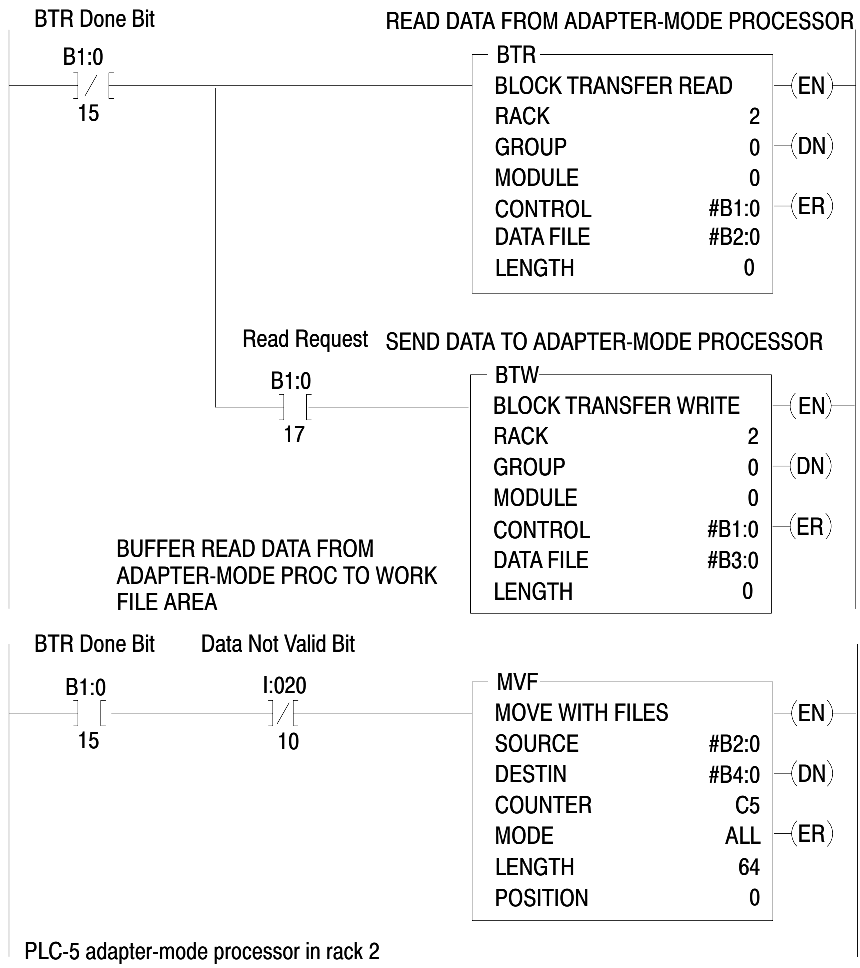
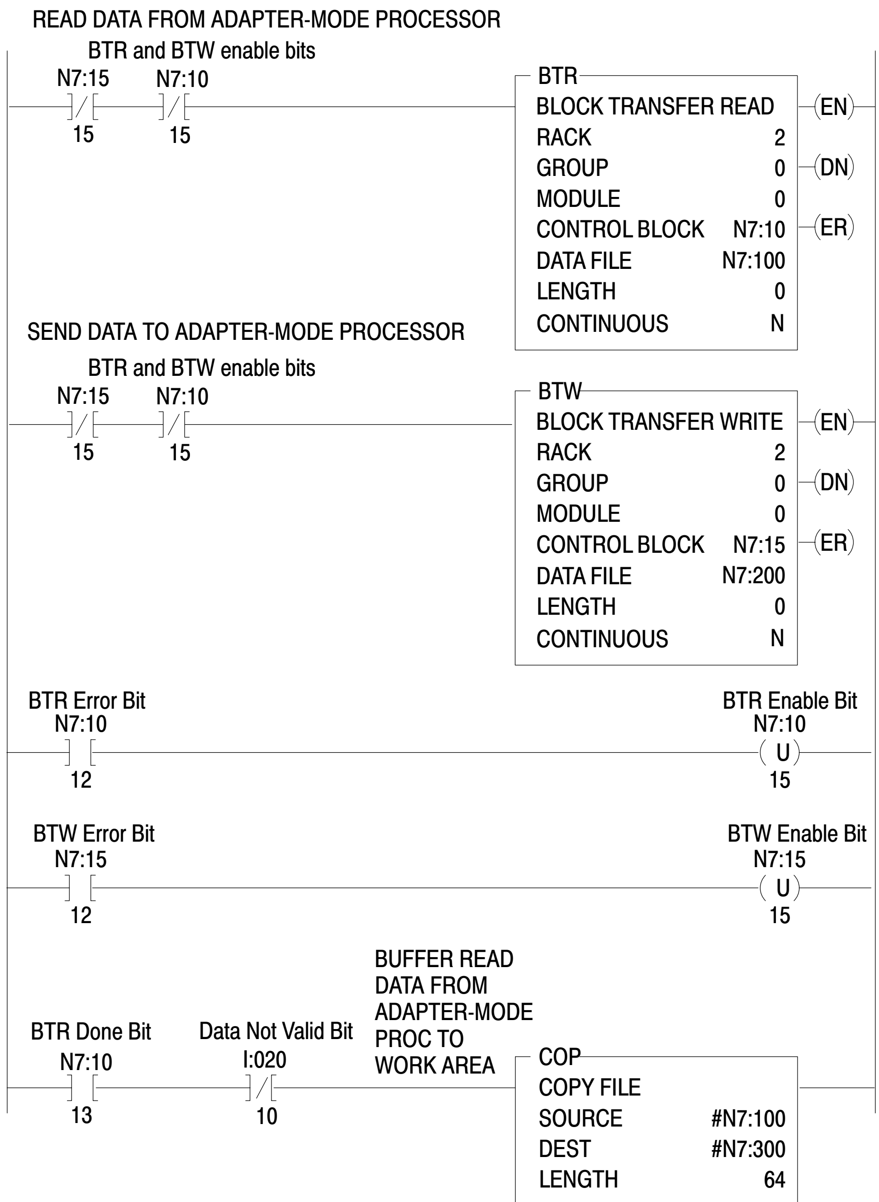
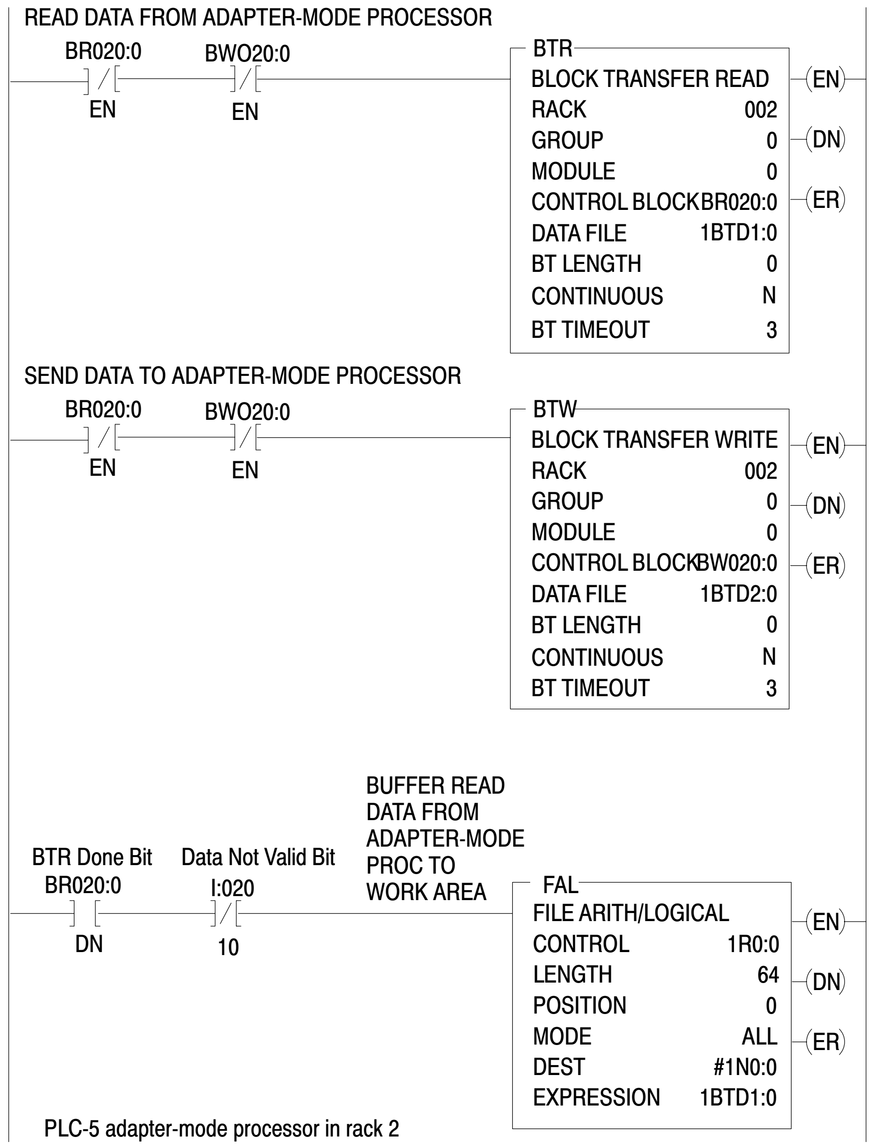


Figure 8.8
Example Block Transfer in PLC-5 Supervisory Processor



PLC-5 adapter-mode processor in rack 2

Figure 8.9
Example Block Transfer in PLC-5/250 Supervisory Processor

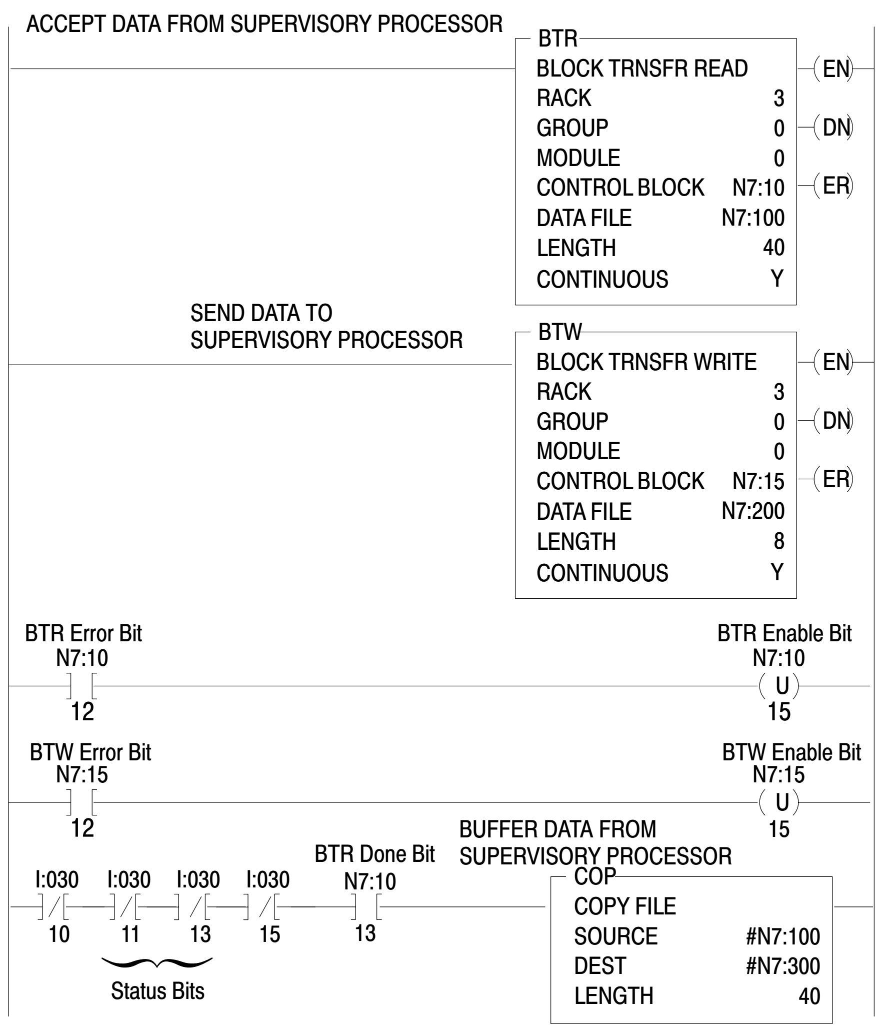


Adapter-Mode Processor (PLC-5/12, -5/15, and -5/25)

Follow these guidelines when programming block-transfer instructions in the adapter-mode processor.

- Use 3 for the rack, 0 for the group, and 0 for the module.
- Set the continuous bit for continuous operation.
- Condition the use of BTR data with status bits from the supervisory processor.

Figure 8.10
Example Block Transfer for a PLC-5/12, -5/15, or -5/25 Processor in Adapter Mode



Transferring Data Using Scanner Mode

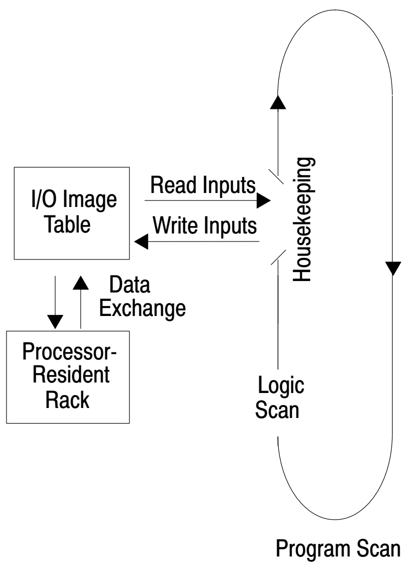
A PLC-5 processor, in scanner mode, transfers discrete-transfer and block-transfer data with processor-resident local and remote I/O chassis. If you have your processor configured for scanner mode, refer to the following sections for more information on how a PLC-5 processor transfers data in scanner mode. Also, the following sections provide information on how to handle I/O rack faults for processor-resident local and remote I/O (in scanner mode).

Programming Discrete Transfer in Scanner Mode

The processor scans processor-resident local I/O synchronously and sequentially to the program scan.

The processor:

- scans discrete-transfer data in the processor-resident local I/O chassis synchronously and sequentially to the program scan.
- scans discrete-transfer data in remote I/O chassis asynchronously to the program scan. The remote I/O scan discrete-transfers I/O data between remote I/O adapters in I/O chassis and the remote I/O buffer in the processor.
- performs housekeeping once per program scan: 3 ms maximum; typically 1.5 ms



Programming Block Transfer in Scanner Mode

The processor block transfers data to and from its processor-resident local and remote I/O chassis when operating in scanner mode. The processor performs block transfers asynchronously to the program scan. The processor also interrupts the program scan asynchronously to momentarily access BTW and BTR data files. The processor performs one remote block transfer per addressed rack per remote I/O scan in classic PLC-5 systems.

Queued Block-Transfer Requests

If your ladder program requests more than one block transfer to or from the same I/O chassis in the same program scan, the processor queues the requests. The PLC-5/12, -5/15, and -5/25 processors can handle up to 17 requests per rack address.

After the processor queues the requests, a PLC-5/12, -5/15, or -5/25 processor runs the block transfers in the order they are requested. The only exception is a block-transfer request in a fault routine.

The processor has an active buffer. The processor places a block transfer in the active buffer when the processor takes the request off the queue. The processor places a block-transfer request directly in the active buffer only if the queue is empty.

When the processor is changed to program mode, the block-transfer instructions still in the buffer are cancelled.

Block Transfers to Processor-Resident Local I/O

Block transfers to processor-resident local I/O follow these procedures.

- Block-transfer requests are queued for the addressed processor-resident local I/O rack.
- The active buffer handles all block-transfer modules whose block-transfer instructions were enabled in the program scan continuously via the queue scan in the order the requests were queued.
- The processor momentarily interrupts program scan when the active buffer performs a block-transfer request to access the block-transfer data file.
- Block transfers of I/O data can finish and the done bit can be set anytime during the program scan.

The processor runs all enabled block transfers of I/O data to processor-resident local I/O continuously as each block-transfer request enters the active buffer. The processor does not wait for the I/O scan to queue the requests.

Block Transfers of Remote I/O Data

Block transfers of I/O data to remote I/O follow these procedures.

- block-transfer requests are queued for each addressed remote I/O rack
- Each active buffer transfers one data block per remote I/O scan.
- The processor momentarily interrupts program scan when the active buffer performs a block-transfer request to access the block-transfer data file.

If program scans are two or three times longer than remote I/O scans, the processor can run two or three remote block transfers per program scan and interrupt the program scan two or three times.

Important: If you split remote rack numbers between scanner channels, block transfers to lower priority scanner channels do not work. Discrete transfers function properly. Scanner channels have priority according to the following order: 1A, 1B, 2A, then 2B. If you configure channels 1B and 2A as remote scanners and split rack #2 between them, for example, block transfers will work to 1B (the higher priority channel) but will not work to the second half of rack #2 (2A, the lower priority channel).

Block Transfers in Fault Routines (STIs)

If the processor runs a fault routine that contains block-transfer instructions, the processor performs these block transfers immediately upon completing any block transfers currently in the active buffer, ahead of block-transfer requests waiting in the queue.

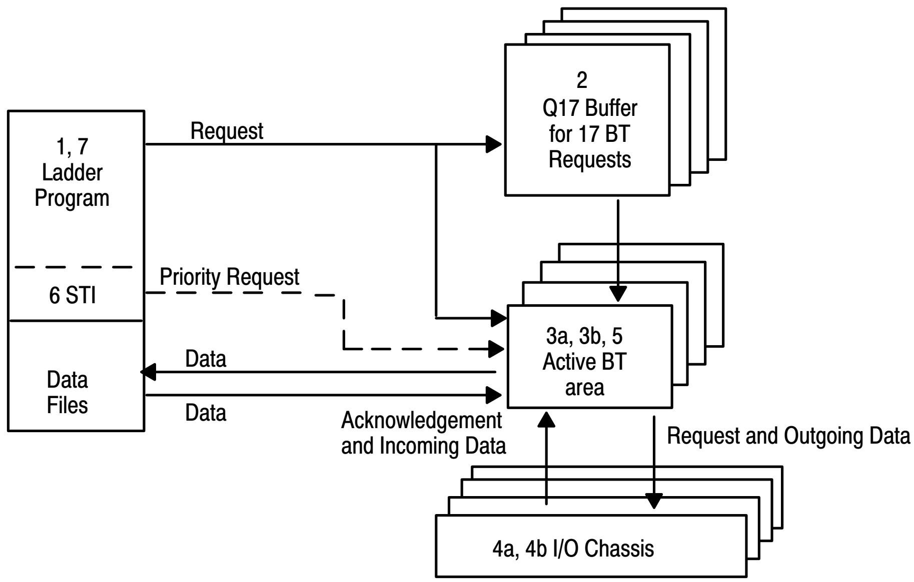
The block transfers in a fault routine should only be between the processor and processor-resident local I/O.

ATTENTION: The program scan stops when the PLC-5/15 or -5/25 processor runs a fault routine with a block-transfer instruction to a remote chassis. The delay for a block transfer could be unacceptable for your application.

Block-Transfer Sequence

Figure 8.11 shows the sequence the processor follows to run a block transfer.

Figure 8.11
Block-Transfer Sequence



1. Ladder logic enables the block transfer.
2. The processor places the block-transfer request in the queue, or in the active buffer if the queue is empty. If the queue is full, the request is ignored until the next scan.
3. If the block transfer is a:
   - **BTW**: The processor interrupts the program scan momentarily to transfer data from the BTW file to the active buffer. The active buffer transfers the request and outgoing data to the processor-resident local I/O module or to the remote I/O adapter.
   - **BTR**: The active buffer sends the block-transfer request to the processor-resident local I/O module or remote I/O adapter. In the same local block-transfer update or in the next remote I/O scan, the active buffer receives the block-transfer acknowledgement and incoming data.

     Important: The processor interrupts the program scan momentarily to transfer incoming data to the BTR file one word at a time; therefore, some ladder logic could execute in between word transfers to the BTR file. We recommend that you buffer your BTR data with a file-to-file move or a copy instruction using a BTR done bit to condition the rung if you need file integrity of the data.
4. If the block transfer is to:
   - Processor-Resident Local I/O: The processor continuously runs block-transfer requests for all processor-resident local I/O modules in the order the processor queues the requests.
   - Remote I/O: The processor runs one block-transfer request for one block-transfer module per rack address per remote I/O scan.
5. The processor clears the active buffer and the active buffer accepts the next request after the buffer receives a confirmation of a valid read or write.
6. When the processor enables a fault routine or STI, the processor runs any block-transfer program in the fault routine or STI ahead of any block-transfer requests in the queue, as soon as the active buffer completes any block transfer currently in the active buffer. The program scan is stopped until the STI or fault routine block transfer is complete.
7. The block-transfer process runs asynchronously to the program scan, so data can change during program scan.

Block-Transfer Sequence with Status Bits

The following explanations describe how the ladder logic and the I/O scanner handle block transfers with status bits:

Ladder logic:

- detects that the rung containing a block transfer is enabled
- sets the enable .EN bit (15)
- detects the status of the read/write .RW bit (07)
- places the block transfer in the active buffer if the queue is empty; the processor sets the start .ST bit (14) and begins the transfer
- places the block transfer in the queue if the active buffer is not empty; the processor sets the enabled waiting .EW bit (10)

If the queue is full, block-transfer requests may not occur in the order the ladder logic requests the transfers. The processor sets the enabled waiting .EW bit (10) when the request enters the queue.

The I/O scanner:

- transfers the request to or from the I/O chassis after the request reaches the active buffer
- detects whether the module responds; if the module does not respond, the processor sets the no response .NR bit (09)

  If there is no response and the timeout .TO bit (08) is reset, the processor re-queues the request until the watchdog timer times out (4 seconds). If there is no response and the .TO bit is set, the scanner retries the request one more time before setting the .ER bit.
- If the request is a:
  - **BTW,** the processor transfers the data to the module
  - **BTR,** the processor moves data from the module to the BTR data file one word at a time
- sets the done .DN bit (13) on completion of a valid transfer; sets the error .ER (12) bit if there were errors
- checks the status of the continuous .CO bit (11); if set and no error occurred, the scanner re-queues the block transfer
- notifies the active buffer to accept the next request

For a list of block-transfer error codes, see the block-transfer instructions chapter in the programming software documentation.

Programming Considerations

In a distributed control system where your process is controlled by several independent programmable controllers, make sure that your program considers the status of the PLC processors and the integrity of the communication link by using the status bits that the supervisory and adapter mode processor provide for each other.

For example, consider how your process should respond if:

- there is an incremental degradation of the systems control due to the loss of one of the programmable controllers
- the supervisory processor is in program mode and someone manually activates a valve normally controlled by the supervisory processor
- the adapter-mode processor faults

The adapter-mode processor can monitor the status of the supervisory processor by examining the status bits in the first word of the the data being transferred from the supervisory processor.

The supervisory processor can monitor the status of the adapter-mode processor by examining the status bits in the first word of the data being transferred from the adapter-mode processor. The supervisory processor can also monitor the rack fault bits for the rack the adapter is emulating to determine the integrity of the remote I/O communications between the supervisor and the adapter-mode processors. For more information on rack fault bits, see the faults chapter in the programming software documentation.

General Considerations for Block-Transferring I/O Data

The following are general programming considerations when you are block-transferring I/O data.

- When performing block transfers (processor-resident local, extended-local, or remote I/O) in any PLC-5 processor clear the output image table corresponding to the block-transfer module rack location before changing to RUN mode. If you do not clear the output image table, then you encounter block-transfer errors because unsolicited block transfers are being sent to the block-transfer module (i.e. if a block-transfer module is installed in rack 2, group 4, clear output word O:024 to 0. Do not use the word for storing data).
- If you use remote block-transfer instructions and have the timeout bit (.TO) set to 1, then the processor disables the 4 sec timer and requests additional block transfers anywhere from 0-1 sec before setting the error (.ER) bit.

Considerations for Processor-Resident Local Racks

The following are programming considerations when you are block-transferring data in a processor-resident local rack.

- Within the processor-resident local rack, limit the number of continuous-read block transfers to 16 transfers of 4 words each or 8 transfers of 64 words each. If you attempt to exceed this block-transfer limit, a checksum error (error code -5) occurs.
- Block-transfer instructions to any of the following modules residing in the processor-resident local rack result in frequent checksum errors.
  - 1771-OFE1, -OFE2, and -OFE3 modules, all versions prior to series B, revision B.
  - 2803-VIM module, all versions prior to series B, revision A
  - IMC-120, all versions

  To eliminate the checksum errors, replace your modules with the current series and revision. If replacement is not possible then:

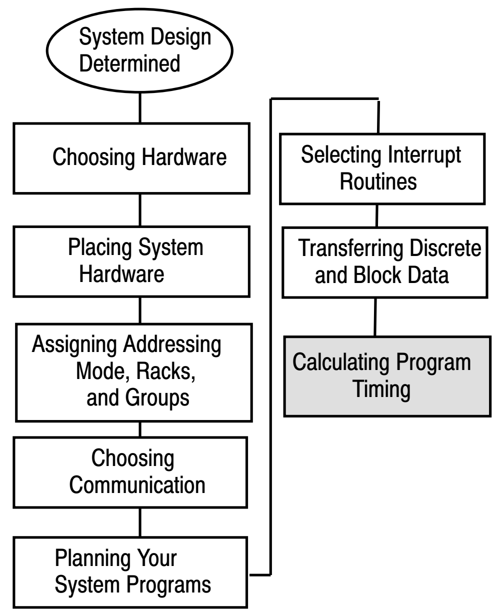
  1. Using 6200 Series PLC-5 Programming Software release 4.11/4.12 or later, go to the Processor Status screen.
  2. With the processor in PROG mode, set user control bit 4 to 1. The User Control Bit word is S:26.
  3. Change the processor mode from PROG to RUN.
- Do not program IIN or IOT instructions to a module in the same physical module group as a BT module unless you know a block transfer is not in progress. If you must do this, then use an XIO instruction to examine the EN bit of the block-transfer instruction to condition the IIN and IOT.

Chapter 9

Calculating Program Timing

Chapter Objectives

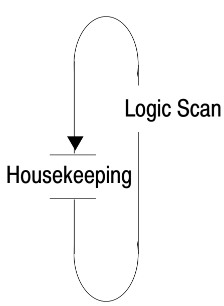
This chapter provides information to help you determine the program timing for your PLC-5 programmable controller system.



Introduction to Classic PLC-5 Processor Scanning

The basic function of a programmable controller system is to read the status of various input devices (such as pushbuttons and limit switches), make decisions based on the status of those devices, and set the status of output devices (such as lights, motors and heating coils). To accomplish this, the PLC-5 processor performs two primary operations:

- program scanning—where
  - logic is executed
  - housekeeping is performed
- I/O scanning—where input data is read and output levels are set



Program Scanning

The program scan cycle is the time it takes the processor to execute the logic scan once, perform housekeeping tasks, and then start executing logic again.

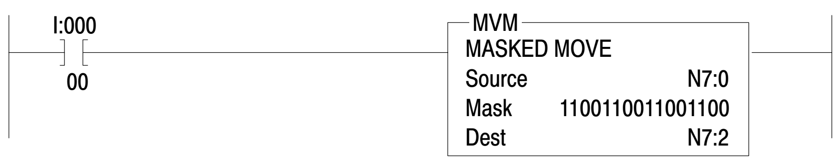
The processor continually performs logic scanning and housekeeping. You can monitor the program scan time using the processor status screen. Housekeeping activities for most PLC-5 processors includes:

- processor internal checks
- updating the input image table with processor-resident I/O input status
- updating processor-resident local I/O output modules with data from the output image table
- updating the input image table with remote I/O input status as contained in the remote I/O buffer
- updating the remote I/O buffer with output data from the output image table

If no change in input status occurs and the processor continues to execute the same logic instructions, the program scan cycle remains consistent (in our example, at 25 ms). In real systems, however, the program scan cycle fluctuates due to the following factors:

- false logic executes faster than true logic
- different instructions execute at different rates
- different input states cause different sections of logic to be executed
- interrupt programs affect program scan times

Effects of False versus True Logic on Scan Time
The rung below—which changes states from one program scan to the next—changes your program scan time by about .25 ms.



If I:000/00 is:	Then the rung is:
On	True and the processor executes the masked-move instruction. A masked-move instruction takes 258 µs to execute (see appendix A of the PLC-5 Programming Software Instruction Set Reference, publication 6200-6.4.11).
Off	False and the processor scans the rung but does not execute it. It takes only 1.4 microseconds to just scan the rung.

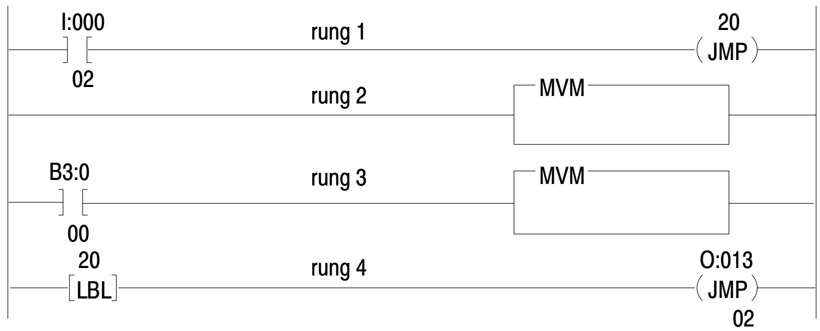
Other instructions may have a greater or lesser effect.

Effects of Different Instructions on Logic Scan Time
Some instructions have a much greater effect on logic scan time than others based on the time it takes to execute that instruction.

Program scan time is also affected by the basic construction of your ladder rungs. The sizes of rungs and the number of branches in each can cause the scan time to fluctuate greatly.

Effects of Different Input States on Logic Scan Time

You can write your logic so that it executes different rungs at different times, based on input conditions. The different amounts of logic executed in the logic scans causes differences in program scan times. For example, the simple differences in rung execution in the following example cause the logic scan times to vary.



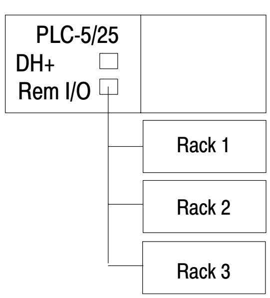
If I:000/02 is:	Rungs 2 and 3 are:
On	Skipped
Off	Executed

If you use subroutines, program scan times can vary by the scan time of entire logic files.

I/O Scanning

The remote I/O scan cycle is the time that it takes for the processor (configured as a scanner) to communicate with all of the entries in its rack scan-list once. The remote I/O scan is independent of and asynchronous to the program scan.

The scanner processor keeps a list of all of the devices connected to each remote I/O link. An example system would look like this:



I/O Status

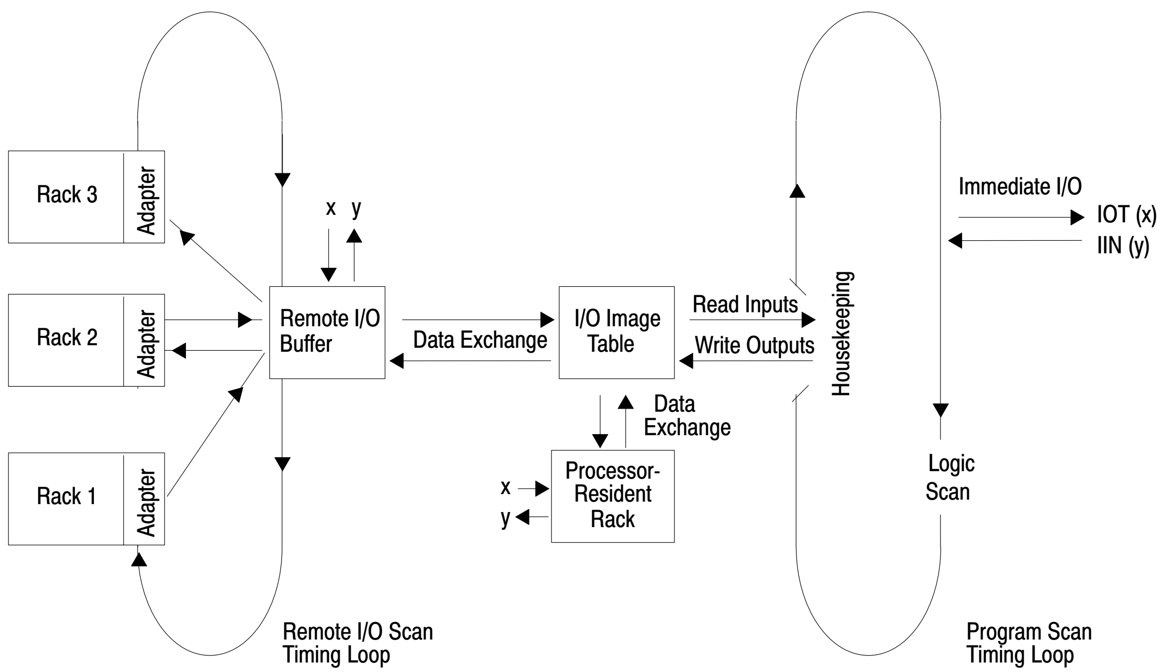
Rack Address	Rack Size	I/O Range
1	Full	IO 010/00 to 017/17
2	1/2	IO 020/00 to 023/17
3	Full	IO 030/00 to 037/17

In this example, the remote I/O channel continually scans the three racks in its scan list and places the data in the remote I/O buffer in the processor. The processor updates its own buffer and the I/O image table. During housekeeping, the two buffers are updated by exchanging the input and output data with each other.

I/O placed in the same chassis as the processor is called “processor-resident” local I/O. These inputs and outputs are not updated during the remote I/O scan—they are updated during the housekeeping portion of the program scan. During housekeeping, the processor reads and writes the I/O across the chassis backplane. Thus, the update of processor-resident local I/O is synchronous to the program scan.

Figure 9.1 shows timing loops for discrete data transfer in a PLC-5 processor.

Figure 9.1
Remote I/O Scan and Program Scan Timing Loops



During the housekeeping portion of the program scan, both the remote I/O buffer and the processor-resident rack are updated. Remember that the I/O scanner is constantly updating the remote I/O buffer asynchronously to the program scan.

I/O Scanning—Discrete and Block Transfer

A Classic PLC-5 processor can transfer discrete data and block data to/from processor-resident local I/O, extended-local I/O chassis, and remote I/O chassis.

Transferring Discrete Data

The remote I/O system is scanned in a separate and asynchronous scan to the program scan. The remote I/O scan takes output data from the remote I/O buffer to output modules and puts input data into the remote I/O buffer from input modules. The remote I/O scan time can take 3, 6, or 10 ms per one rack in a chassis on the remote I/O link, depending on baud rate. The PLC-5 processor then exchanges the input and output image table data with the remote I/O buffer during the I/O-update portion of housekeeping.

Immediate I/O

The processor responds to immediate input (IIN) and immediate output (IOT) requests during the logic scan. The logic scan is suspended at the request for immediate input/output data. The logic scan resumes after obtaining the data and fulfilling the request.

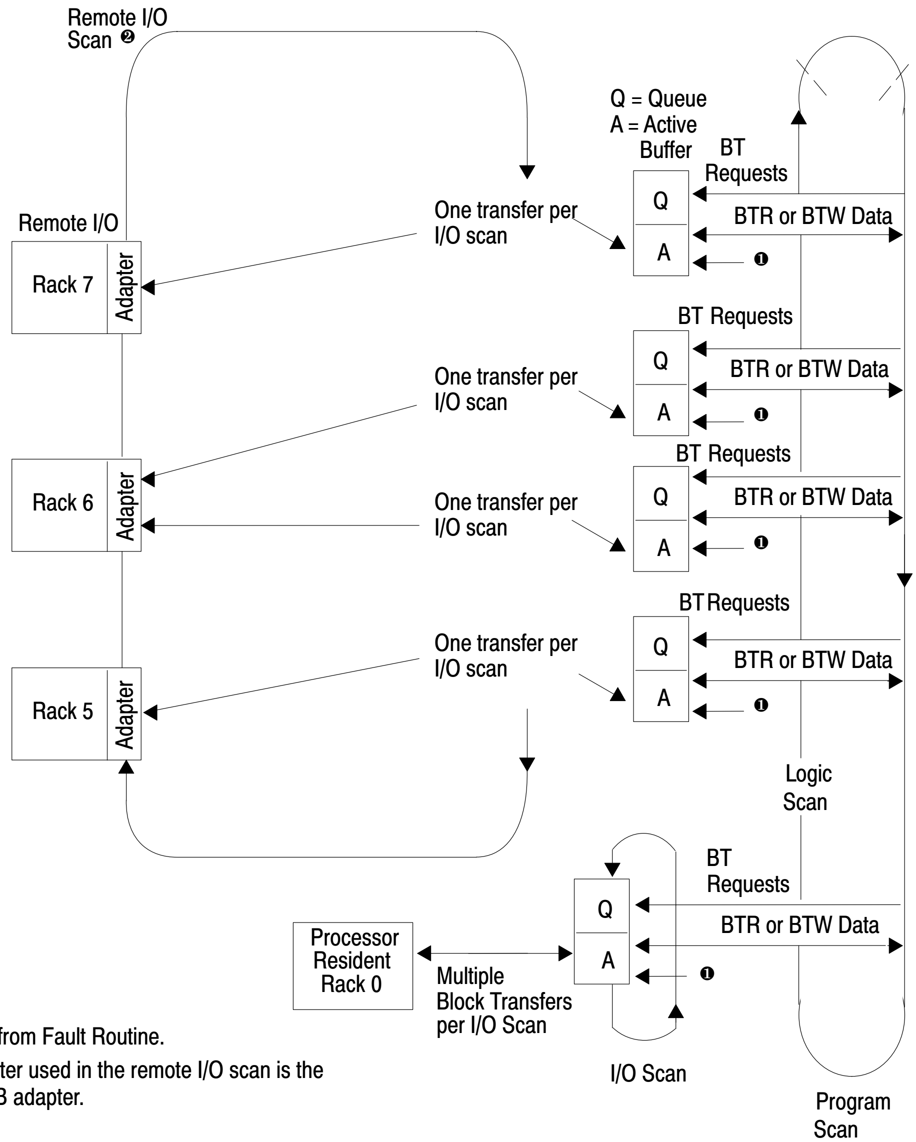
IIN and IOT data transfers directly to and from I/O modules in processor-resident I/O and extended-local I/O chassis. With remote I/O, only the remote I/O buffer is updated.

Transferring Block Data

The exchange of block-transfer data and the logic scan run independently and concurrently. The following paragraphs explain block transfer for extended-local I/O and then for processor-resident I/O and remote I/O.

Remote I/O and Processor-Resident I/O
The processor performs block transfers asynchronously to the program scan. The processor also interrupts the program scan asynchronously to momentarily access BTW and BTR data files. The processor performs one remote block transfer per addressed rack and per remote I/O scan. Figure 9.2 shows timing loops for block transfer from a Classic PLC-5 processor.

Figure 9.2
Transferring Block Data to Local and Remote I/O



❶ Interrupt from Fault Routine.
❷ The adapter used in the remote I/O scan is the 1771-ASB adapter.

Instruction Timing and Memory Requirements

The time it takes for a processor to scan an instruction depends on the type of instruction, the type of addressing, the type of data, whether the instruction has to convert data, and whether the instruction is true or false.

This section provides timing and memory requirements estimates with these assumptions:

- direct addressing
- integer data, except where noted
- no data-type conversions
- addresses within the first 4096 words of the data table of a PLC-5/10, -5/12, -5/15, or -5/25 processor
- execution times shown in μs

Memory requirements refer to the number of words the instruction uses. In some cases, an instruction may have a range of memory requirements. The range of words exists because the instruction can use different types of data.

Important: The tables are divided into instruction times and memory requirements that are specific to PLC-5/10, -5/12, -5/15, and -5/25 processors.

Bit and Word Instructions for PLC-5/10, -5/12, -5/15, and -5/25 Processors

Table 9.N shows timing and memory requirements for bit and word instructions for PLC-5/10, -5/12, -5/15, and -5/25 processors.

Table 9.N
Timing and Memory Requirements for Bit and Word Instructions for PLC-5/10, -5/12, -5/15, and -5/25 Processors

Category	Code	Title	Execution Time (μs) Integer		Execution Time (μs) Floating Point		Words of Memory[2]
			True	False	True	False	
Relay	XIC	examine if closed	1.3	0.8			1[1]
	XIO	examine if open					
	OTL	output latch	1.6				
	OTU	output unlatch					
	OTE	output energize	1.6	1.6			
Branch		branch end	0.8	0.8			1
		next branch					
		branch start					
Timer and Counter	TON	timer on (0.01 base) (1.0 base)	39 44	27 28			2-3
	TOF	timer off (0.01 base) (1.0 base)	30 30	43 51			
	RTO	retentive timer on (0.01 base) (1.0 base)	39 44	24 24			
	CTU	count up	32	34			
	CTD	count down	34				
	RES	reset	30	14			

1 For every bit address above the first 256 words of memory in the data table, add 0.8 μs to the execution time and 1 wordd of memory to the requirements.

2 Use the smaller number if all addresses are below word 4096; use the larger number if all addresses are above 4096.

Category	Code	Title	Execution Time (µs) Integer		Execution Time (µs) Floating Point		Words of Memory[2]
			True	**False**	**True**	**False**	
Arithmetic	ADD	add	36	14	92	14	4-7
	SUB	subtract					
	MUL	multiply	41	14	98		
	DIV	divide	49	14	172		
	SQR	square root	82	14	212	14	3-5
	NEG	negate	28	14	36		
	CLE	clear	18	14	23	14	2-3
	TOD	convert to BCD	52	14			3-5
	FRD	convert from BCD	44				
Logic	AND	and	36	14			4-7
	OR	or					
	XOR	exclusive or					
	NOT	not	27	14			3-5
Move	MOV	move	26	14	35	14	
	MVM	masked move	55	14			6-9
Comparison	EQU	equal	32	14	42	14	3-5
	NEQ	not equal					
	LES	less than					
	LEQ	less than or equal					
	GRT	greater than					
	GEQ	greater than or equal					
	LIM	limit test	42	14	60	14	4-7
	MEQ	mask compare if equal	41	14			

[2] Use the smaller number if all addresses are below word 4096; use the larger number if all addresses are above 4096.

Category	Code	Title	Execution Time (µs) Integer		Execution Time (µs) Floating Point		Words of Memory[2]
			True	False	True	False	
Compute	CPT	add	67	34	124	34	6-9
		subtract					
		multiply	73	34	130		
		divide	80	34	204		
		square root	113	33	244	34	5-7
		negate	59	33	68		
		clear	49	30	55	34	4-5
		move	58	33			5-7
		convert to BCD	84				
		convert from BCD	75				
		AND	68	34			6-9
		OR					
		XOR					
		NOT	59	34			5-7
Compare	CMP	equal	63	34	73	34	
		not equal					
		less than					
		less than or equal					
		greater than					
		greater than or equal					

[2] Use the smaller number if all addresses are below word 4096; use the larger number if all addresses are above 4096.

File Instructions

The instruction timing for file instructions depends on the data type, number of files acted on per scan, number of elements acted on per scan, and whether the instruction converts data between integer and floating point formats.

Table 9.O shows PLC-5/10, -5/12, -5/15, and -5/25 processors. When you use these tables, keep these guidelines in mind:

- for integer to floating point conversion, add:

  8 µs for each element address
  10 µs for each file address (# prefix)

- for floating point to integer conversion add:

  33 µs for each element address
  44 µs for each file address (# prefix)

Table 9.O
Timing and Memory Requirements for File Instructions

Category	Code	Title	Time (µs) Integer True	Time (µs) Floating point True	Time (µs) Integer or Floating Point False	Words of Memory[1]
File Arithmetic and Logic	FAL	add	98 + W[36.7 + N]	98 + W[95.1 + N]	54	7-12
		subtract				
		multiply	98 + W[42.5 + N]	98 + W[101.2 + N]		
		divide	98 + W[51.1 + N]	98 + W[180.3 + N]		
		square root	98 + W[84.7 + N]	98 + W[220.5 + N]	54	6-10
		negate	98 + W[29.2 + N]	98 + W[37.2 + N]		
		clear	98 + W[18.4 + N]	98 + W[24.0 + N]	54	5-8
		move	98 + W[27.3 + N]	98 + W[36.2 + N]	54	6-10
		convert to BCD	98 + W[54.3 + N]			
		convert from BCD	98 + W[45.4 + N]			
		AND	98 + W[37.2 + N]		54	7-12
		OR	98 + W[37.2 + N]			
		XOR	98 + W[37.2 + N]			
		NOT	98 + W[28.2 + N]		54	6-10
File Search and Compare	FSC	all comparisons	93 + W[32.7 +N]	93 + W[43.3 +N]		

W = number of elements acted on per scan

N = 2 x (number of integer file addresses) + 8 x (number of floating-point file addresses) + 6 x (number of timer, counter, or control file addresses) + (number of conversions between integer and floating point formats)

[1] Use the smaller number if all addresses are below word 4096; use the larger number if all addresses are above 4096.

Category	Code	Title	Time (μs) Integer True	Time (μs) Floating point True	Time (μs) Integer or Floating Point False	Words of Memory[1]
File	COP	copy counter, timer, and control	88 + 2.7W 98 + 5.8W	104 + 3.8W	20	4-7
	FLL	fill counter, timer, and control	81 + 2/.1 W 97 + 4.4W	100 + 3.1W	15	
Shift Register	BSL	bit shift left	74 + 3.4W		57	
	BSR	bit shift right	78 + 3.0W			
	FFL	FIFO load	54		44	
	FFU	FIFO unload	68 + 3.2W		46	
Diagnostic	FBC	file bit compare				6-11
		0 mismatch	75 + 6W		31	
		1 mismatch	130 + 6W			
		2 mismatches	151 + 6W			
	DDT	diagnostic detect				
		0 mismatch	71 + 6W		31	
		1 mismatch	150 + 6W			
		2 mismatches	161 + 6W			
Zone Control	MCR	master control	12		18	1
Immediate I/O	IIN	immediate input				2-3
		local	196		16	
		remote	204			
	IOT	immediate output				
		local	202		16	
		remote	166			
Sequencer	SQI	sequencer input	57		14	5-9
	SQL	sequencer load	55		42	4-7
	SQO	sequencer output	77		42	5-9
Jump and Subroutine	JMP	jump	45		15	2-3
	JSR	jump to subroutine				
	SBR	0 parameters	56		15	2-3
		1 parameter	91			3-5
		add per parameter	21			
	RET	return from sub.				
		0 parameters	48		13	1
		1 parameter	70			2-3
		add per parameter	21			
	LBL	label	12		12	3

W = number of elements acted on per scan
[1] Use the smaller number if all addresses are below word 4096; use the larger number if all addresses are above 4096.

Category	Code	Title	Time (μs) Integer True	Time (μs) Floating point True	Time (μs) Integer or Floating Point False	Words of Memory[1]
Miscellaneous	END	end	negligible		negligible	1
	TND	temporary end	negligible		15	
	AFI	always false	15		13	
	ONS	one shot	28		30	2-3
	DTR	data transitional	41		41	4-7
	BTD	bit distributor	77		14	6-11
	PID	PID loop control	608		34	5-9
	BTR	block transfer read	See Block Transfer BTR, BTW chapter			
	BTW	block transfer write				
	MSG	message	See Message MSG chapter			

W = number of elements acted on per scan

N = 2 x (number of integer file addresses) + 8 x (number of floating-point file addresses) + 6 x (number of timer, counter, or control file addresses) + (number of conversions between integer and floating point formats)

[1] Use the smaller number if all addresses are below word 4096; use the larger number if all addresses are above 4096.

Program Constants

Use program constants in compare, compute, and file instructions to improve instruction execution times. Integer constants and floating-point constants execute in less than 1 μs. Note that if you program using constants you must edit the program to change the constants. If you program using data table addresses, you can change the values by simply changing the value in the data table.

Direct and Indirect Elements

Directly Addressed Elements

Additional execution time for directly addressed elements depends on:

- data types
- location in memory, referenced to the beginning of all data files (output file, word 0)
- whether data is stored at the source or destination address
- whether the instruction converts data

Table 9.P lists times to add to instruction execution times.

Table 9.P
Additional Execution Time Based on Source and Destination Addresses

Data Type	Source (integer to floating point)			Destination (floating point to integer)		
	0-2K	2-4K	4K+	0-2K	2-4K	4K+
Integer	0	1	2	0	1	2
Floating point	0	3	4	0	3	4
Data conversion	8	9	10	33	34	35

When file addresses (# prefix) in the expression or destination address contain indirect addresses for file numbers, add:

- 45 µs when the indirect address is integer type
- 48 µs when the indirect address is floating point type
- 48 µs when the indirect address is timer, counter, or control type

When file addresses in the expression or destination contain indirect addresses for element numbers, add:

- 45 µs when the indirect address is integer type
- 46 µs when the indirect address is floating point type
- 46 µs when the indirect address is timer, counter, or control type

If the file address contains two indirect addresses, add only one value (the largest). For example, for #F[N7:20][N7:30], add 48 µs (indirect floating point file address).

Multiply the additional time by the number of elements in the file for any type of file or file address. For one FAL example:

Expression:	#N[N7:100]/10 * F8:20 add 10 for converting to floating point add 45 for indirect address
Destination:	#N7:30 add 35 for converting to integer
FAL multiply:	98 + W[42.5 + N + indirect addressing] N = 2(2) +8 (1) + 6(0) + 10 + 35 = 57 W = 16
Execution time in ALL mode:	98 + 16[42.5 + 57 +45] 2410 µs

Indirect Bit or Elements Addresses

Additional execution times for indirectly addressed bits and elements depends on the number of variable (indirect) addresses in the overall address. Table 9.Q lists the additional times.

Table 9.Q
Additional Execution Times for Indirectly Addressed Bits and Elements

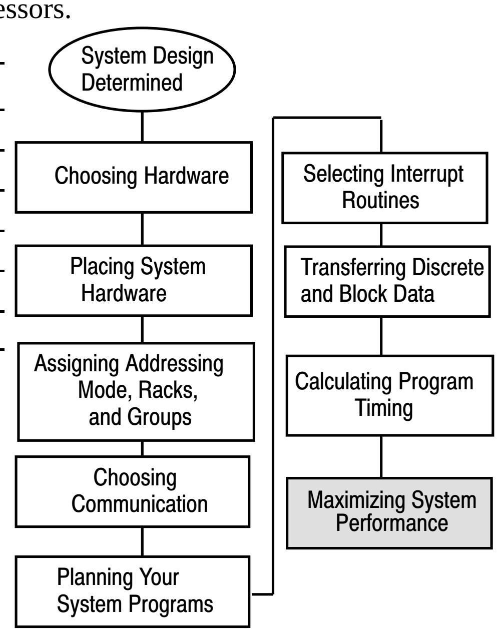
Data Type	Time (μs) for Variable File or Element	Time (μs) for Variable File and Element
Bit in binary file	57	60
Bit in integer file	60	63
Bit in timer, counter, or control file	64	66
Integer (N)	42	42
Timer, counter, or control file	43	44
Floating point (F)	61	64
Converting integer to floating point	71	77
Converting timer, counter, or control to floating point	85	81

Chapter 10

Maximizing System Performance

Chapter Objectives

This chapter explains how to calculate throughput and provides methods for optimizing remote I/O scan time in PLC-5/11, -5/20, -5/30, -5/40, -5/40L, -5/60, -5/60L, and -5/80 processors.



For information on the time that it takes the processor to execute a specific instruction, see chapter 9.

Components of Throughput

Throughput is the time that it takes for an output to be energized after its associated input has been energized. You need to consider the following components when evaluating throughput:

- input and output module delay
- I/O backplane transfer
- remote I/O scan time
- processor time

Input and Output Modules Delay

All input and output modules have a "delay time"—the time that it takes the module to transfer information to/from the I/O backplane through the I/O module to/from the field device. Depending on the type of modules you are using, these delay times vary; but the times must be taken into account when calculating system throughput. Choose modules that perform the function that you need with the lowest possible delay times.

I/O Backplane Transfer

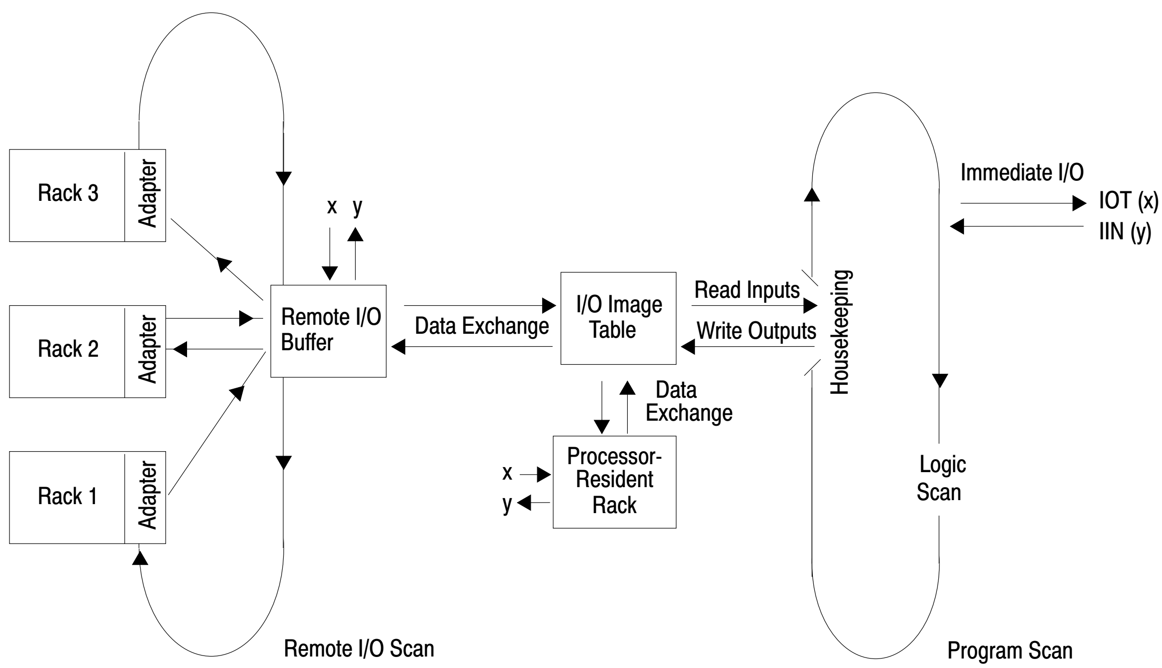
The I/O backplane transfer time is the time it takes for the 1771-ASB adapter module to exchange data with the I/O modules in the same chassis, generally 1-2 ms for a full I/O rack.

This time is fairly insignificant compared to total system throughput but can be optimized in situations where there are empty slots or modules that only use backplane power in the chassis. For example, if the last four slots of a rack contain a 1785-KA module and power supply (with 2 empty slots), the 1771-ASB can be configured to ignore those last four slots.

For more information on configuring your ASB module, refer to the 1771 Remote I/O Adapter Module User Manual, publication 1771-6.5.83.

Remote I/O Scan Time

The remote I/O scan time is the time it takes for the scanner to communicate with each device in the remote I/O system.



Three factors that affect the remote I/O scan time are:

- communication rate
- number of chassis
- block transfers

Communication Rate

The baud rate determines the time it takes for the scanner to communicate with each individual entry in its scan list. Table 10.A lists the amount of time required to communicate to a device at each communication rate.

Table 10.A
Communication Times at Different Communication Rates

Communication Rate (kbps)	Time (ms)
57.6	10
115.2	7
230.4	3

Note that these are full rack times. Smaller racks will decrease this time.

If there are four full-rack entries in the scan list, the I/O scan for that channel at 57.6 kbps is 4 x 10 = 40 ms. If you change the baud rate to 230.4 kbps, the I/O scan decreases to 4 x 3 = 12 ms.

Important: All devices on the network must support the baud rate you chose and must be within the required cable lengths.

Number of Rack Entries

You determine the total remote I/O scan time in the remote I/O system by taking the number of rack entries in the scan list and multiplying by the time per rack at the baud rate that you are using (see Table 10.A). If one channel has twice as many racks as another, for example, the scan time for the first channel is twice as long.

To optimize this scan time, divide your I/O racks between multiple channels. Place your most time-critical I/O on one channel, and non-time-critical I/O on the other channel. Since all I/O channels are independent, a long remote I/O scan on one channel will not affect the remote I/O scan on another channel.

Block Transfers

A block transfer is an interruption of the normal remote I/O scan in order to transfer a block of data to a specific I/O module. Most of the time that the processor spends in performing the block transfer is for the handshaking that occurs between the processor and the block-transfer module; this handshaking is embedded in the discrete I/O transfer and has no effect on the remote I/O scan. The remote I/O scan is affected when the actual transferring of data is occurring.

The amount of time that the block transfer interrupts the remote I/O scan depends on the number of words being transferred and the baud rate:

Communication Rate (kbps)	ms/Word	Overhead (ms)
57.6	.28	3
115.2	.14	2.5
230.4	.07	2

If the communication rate is 115.2 kbps and you want to block transfer 10 words, for example, the interruption of the remote I/O scan is:

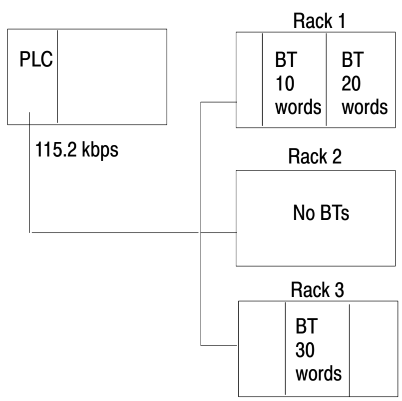
(10 x .14) + 2.5 = 1.4 + 2.5 = 3.9 ms

For the particular remote I/O scan in which the block transfer takes place, 3.9 ms will be added to the remote I/O scan time.

Calculating Worst-Case Remote I/O Scan Time

Since it is impossible to predict which remote I/O scan a block transfer will occur within, you can only calculate the worst-case remote I/O scan time. To calculate the worst case, determine the normal I/O (without block transfers) then add the time of the longest block transfer to each entry in the scan list. (The processor can only perform 1 block transfer per entry in the scan list per I/O scan.)

For example, if your system is:



Worst-case I/O scan:

(3 x 6)	3 racks at 115.2 kbps—normal I/O scan
+ (20 x .14) + 2.5	longest BT in rack 1
+ 0	no BTs in rack 2
+ (30 x .14) + 2.5	longest BT in rack 2

18 + 5.3 + 0 + 6.7 = 30 ms

Optimizing Remote I/O Scan Time

The best way to optimize your scan time is to place your most time-critical I/O on a separate channel from non-critical I/O. If you only have one channel available for I/O, however, you can still optimize the scanning by using the processor's configurable scan list.

In a normal 4-rack system, the scan list would be:

rack 1
rack 2
rack 3
rack 4

If you are using 57.6 kbps, the normal I/O scan is 4 racks x 10ms—40ms. Each entry is of equal priority, so each rack is scanned every 40 ms.

However, if rack 2 has the most time-critical I/O, use the configurable scan list to specify:

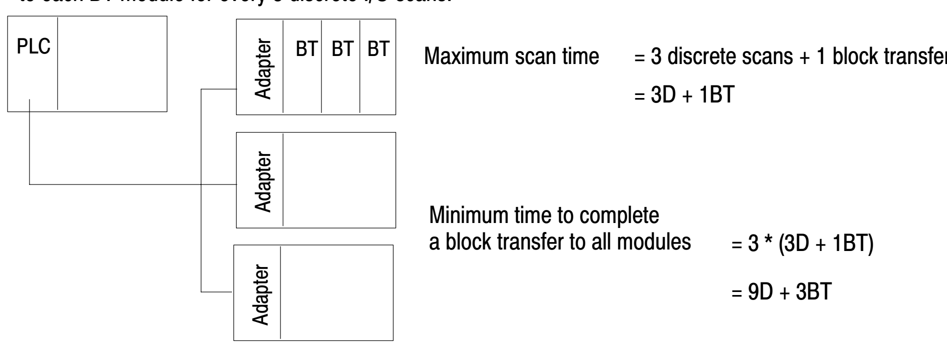
rack 1
rack 2
rack 3
rack 2
rack 4
rack 2

Using this scan list, rack 2 is scanned every other rack. There are 6 entries, so the normal I/O scan time is 6 x 10 ms = 60 ms. Since rack 2 is scanned every other rack, however, the rack 2 **effective** scan time is 2 x 10 ms = 20 ms. The remaining racks are scanned every 60ms. Thus, the tradeoff for the more frequent scanning of rack 2 (every 20 ms) means that the other racks are scanned only every 60 ms.

You can also optimize block transfers within the channel. You block transfer to only one block transfer module per entry in the scan list per I/O scan. If you have three block-transfer modules in one I/O rack, it takes a minimum of three I/O scans to complete the block transfers to all of the modules:

System Optimized for Discrete-Data Transfer

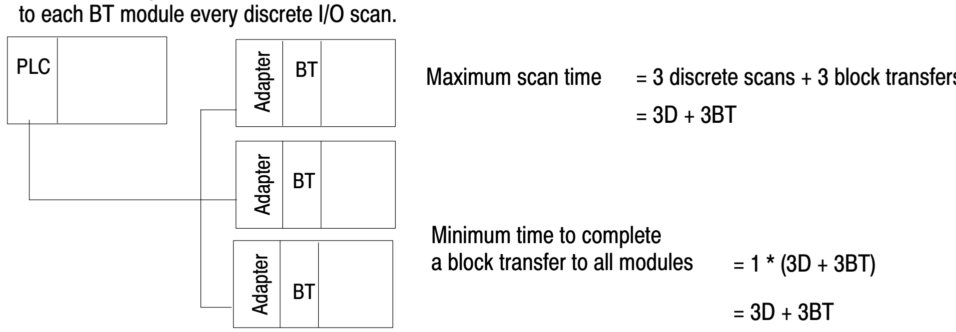
With this arrangement, there is only one block transfer to each BT module for every 3 discrete I/O scans.



If you place the three block-transfer modules in different racks, however, you can block transfer to all three modules in one I/O scan. To optimize your system layout for **block-data transfers**, use an arrangement similar to the following:

System Optimized for Block-Data Transfer

With this arrangement, there is a block transfer to each BT module every discrete I/O scan.



Processor Time

The processor time is the time needed to process the inputs and set the corresponding outputs. This processor time varies for different processors and is based on input buffering, program scan, etc.

In a PLC-5 system, both inputs and outputs are buffered between the I/O image table and the I/O scanner. The movement of inputs from the scanner to the input buffer is asynchronous to the movement of data from the input buffer to the input image table. The worst-case processor time is:

Periodic input buffer update	= 10 ms
One program scan to guarantee inputs received	= *xx* ms
One program scan to guarantee outputs received	= *xx* ms
.18 ms times the number of racks	= *xx* ms

For a 3-rack system with a 20 ms program scan, the worst-case processor time is: 10 + 20 + 20 + .54 = 50.54 ms.

Calculating Throughput

To calculate throughput, use the following equation:

Input Card Delay + I/O Backplane + Worst-Case Remote I/O Scan Time + Worst-Case Processor Time + Worst-Case Remote I/O Scan Time + I/O Backplane + Output Card Delay

An example of a worst-case update time calculation:

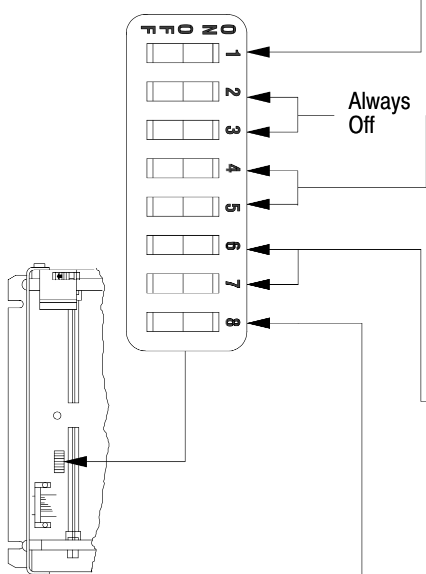
Input card delay	= 20 ms (typical)
I/O backplane	= 1 ms
Worst-case remote I/O scan time	= 30 ms
Worst-case processor time	= 50.54 ms
Worst-case remote I/O scan time	= 30 ms
I/O backplane	= 1 ms
Output card delay	= 8.8 ms (typical)
Total	141.34 ms

Appendix A

Selecting Switch Settings

Chassis Backplane with Classic PLC-5 Processor

Make the following switch selections for Classic PLC-5 processors.



Switch 1	Last State
ON	Outputs of this I/O chassis remain in their last state when a hardware failure occurs. [1]
OFF	Outputs of this I/O chassis are turned off when a hardware failure occurs. [1]

Switches 4	Switches 5	Addressing
OFF	OFF	2 – slot
OFF	ON	1 – slot
ON	OFF	1/2 – slot
ON	ON	Not allowed

Switches 6	Switches 7	EEPROM Transfer
OFF	OFF	EEPROM memory transfer to processor memory at powerup. [2]
ON	ON	EEPROM memory transfers to processor memory if processor memory not valid.
ON	OFF	EEPROM memory does not transfer to processor memory. [3]

Switch 8	RAM Memory Protection
OFF	RAM memory protection disabled.
ON	RAM memory protection enabled. [4]

Pressed in at top ON (closed)

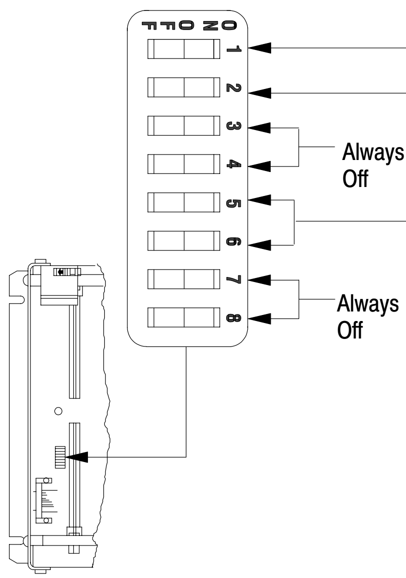
Pressed in at bottom OFF (open)

1. Regardless of this switch setting, outputs are reset when either of the following occurs:
   - processor detects a runtime error
   - an I/O chassis backplane fault occurs
   - you select program or test mode
   - you set a status file bit to reset a local rack
2. If an EEPROM module is not installed and processor memory is valid, the processor's PROC LED indicator blinks, and the processor sets S:11/9, bit 9 in the major fault status word.
3. A processor fault occurs if processor memory (solid red PROC LED) is not valid.
4. You cannot clear processor memory when this switch is on

19309

Chassis Backplane with Adapter Module

Make the following switch selections for a 1771-AS, -ASB, or -ALX adapter module.



Switch 1	Last State
ON	Outputs of this I/O chassis remain in their last state when a communication fault is detected by this I/O adapter. [1]
OFF	Outputs of this I/O chassis are turned off when a comunication fault is detected by this I/O adapter.

Switch 2	Processor Restart Lockout
ON	Processor can restart the I/O chassis after a communication fault. [2]
OFF	You must manually restart the I/O chassis with a switch wired to the 1771-AS or -ASB.

Switches		Addressing
5	6	
OFF	OFF	2-slot
ON	OFF	1-slot [3]
OFF	ON	1/2-slot [3]
ON	ON	Not allowed

Pressed in at top Closed (ON)

Pressed in at bottom Open (OFF)

19308

1. ATTENTION: If you set this switch to the ON position, when a communication fault is detected, outputs connected to this chassis remain in their last state to allow machine motion to continue. We recommed that you set switch 1 to the OFF position to de-energize outputs wired to this chassis when a fault is detected.

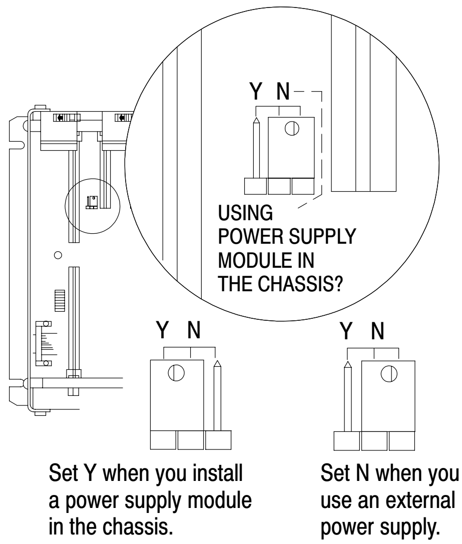
   Also, if outputs are controlled by inputs in a different rack and a remote I/O rack fault occurs (in the inputs rack), the inputs are left in their last non-faulted state. The outputs may not be properly controlled and potential personnel and machine damage may result. If you want your inputs to be anything other than their last non-faulted state, then you need to program a fault routine.

2. Set this switch to ON if you plan to use I/O rack auto-configuration.

3. The 1771-ASB series A adapter does not support 1/2-slot addressing.

Chassis Configuration Plug for Power Supply

Position the configuration plug for the power supply you add to your chassis.



1. Locate the chassis configuration plug (between the first two left most slots of the chassis).
2. Set the I/O chassis configuration plug.

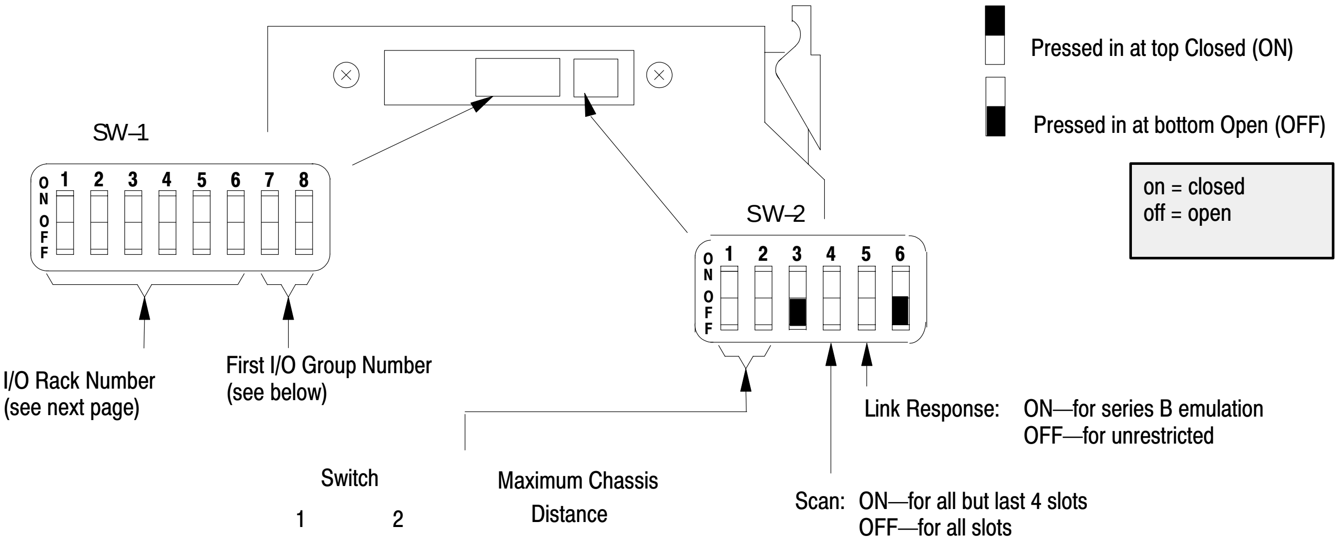
   The default setting is N (not using a power supply module in the chassis).

IMPORTANT: You cannot power a single I/O chassis with both a power supply module and an external power supply.

17075

Remote I/O Adapter Module 1771-ASB Series C without Complementary I/O

Select the switches to determine I/O rack, group, transmission rate, link response, and scan for your adapter module without complementary I/O.



Switch 1	Switch 2	Maximum Chassis Distance
ON	OFF	57.6 Kbps—10,000 ft
OFF	OFF	115.2 Kbps—5,000 ft
OFF	ON	230.4 Kbps—2,500 ft
ON	ON	Not used

First I/O Group Number	7	8
0	on	on
2	on	off
4	off	on
6	off	off

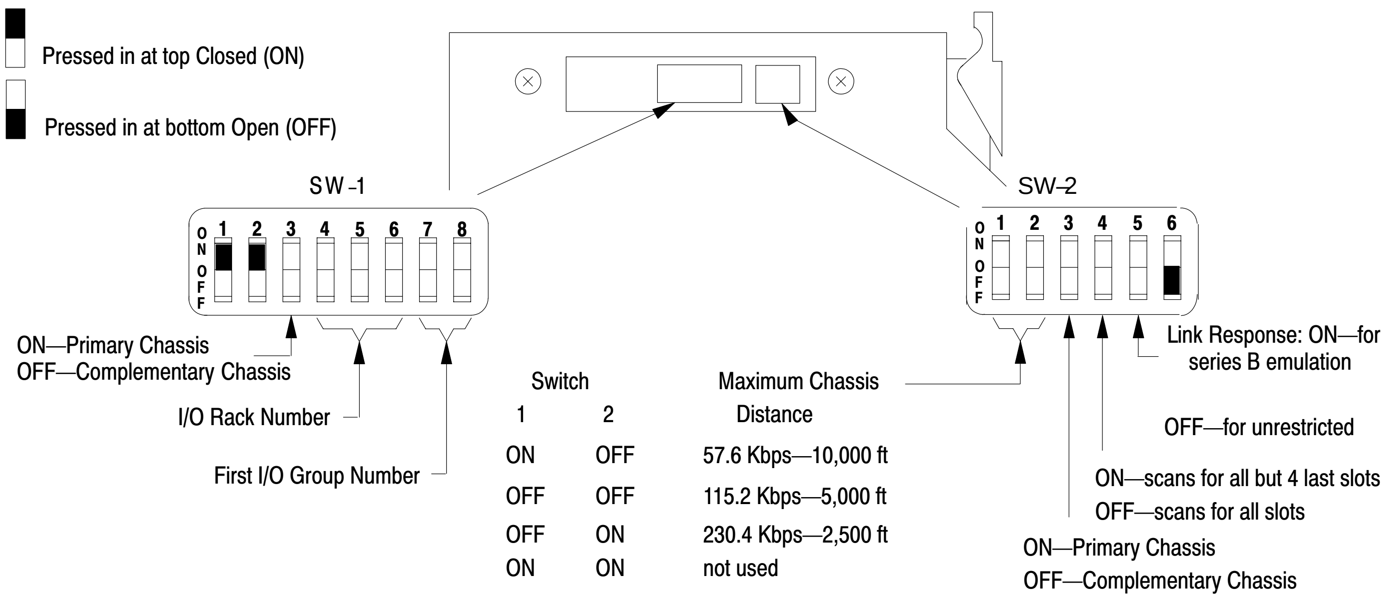
Remote I/O Adapter Module (1771-ASB Series C) I/O Rack Number—without Complementary I/O

Rack	1	2	3	4	5	6
01	on	on	on	on	on	off
02	on	on	on	on	off	on
03	on	on	on	on	off	off
04	on	on	on	off	on	on
05	on	on	on	off	on	off
06	on	on	on	off	off	on
07	on	on	on	off	off	off
10	on	on	off	on	on	on
11	on	on	off	on	on	off
12	on	on	off	on	off	on
13	on	on	off	on	off	off
14	on	on	off	off	on	on
15	on	on	off	off	on	off
16	on	on	off	off	off	on
17	on	on	off	off	off	off
20	on	off	on	on	on	on
21	on	off	on	on	on	off
22	on	off	on	on	off	on
23	on	off	on	on	off	off
24	on	off	on	off	on	on
25	on	off	on	off	on	off
26	on	off	on	off	off	on
27	on	off	on	off	off	off

PLC-5/15 processors—racks 01-03;
PLC-5/25 processors—racks 01-07;

Remote I/O Adapter Module 1771-ASB Series C with Complementary I/O

Select the switches to determine I/O rack, group, transmission rate, link response, and scan for your adapter module using complementary I/O.



Switch 1	Switch 2	Maximum Chassis Distance
ON	OFF	57.6 Kbps—10,000 ft
OFF	OFF	115.2 Kbps—5,000 ft
OFF	ON	230.4 Kbps—2,500 ft
ON	ON	not used

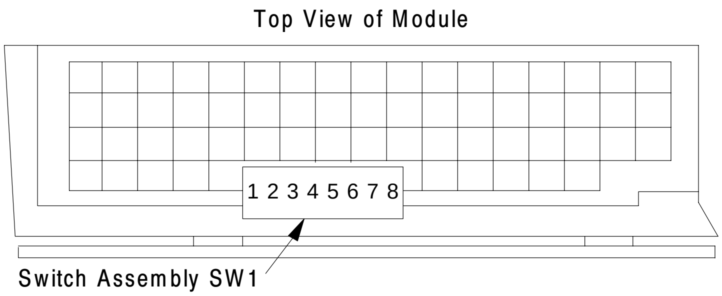
I/O Rack Number	4	5	6
1	on	on	off
2	on	off	on
3	on	off	off
4[1]	off	on	on
5[1]	off	on	off
6[1]	off	off	on
7[1]	off	off	off

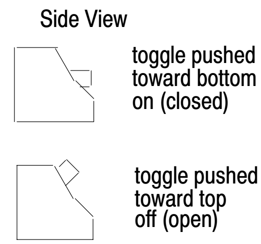
[1] Valid for PLC-5/25 processors only. Only seven racks can be complemented in a PLC-5 system.

For First I/O Group Number	7	8
0	on	on
2	on	off
4	off	on
6	off	off

SW1

Set SW1 switch assembly switches 1 through 6 for the DH+ station number. Switch 7 is not used. Set switch 8 for scanner or adapter mode.





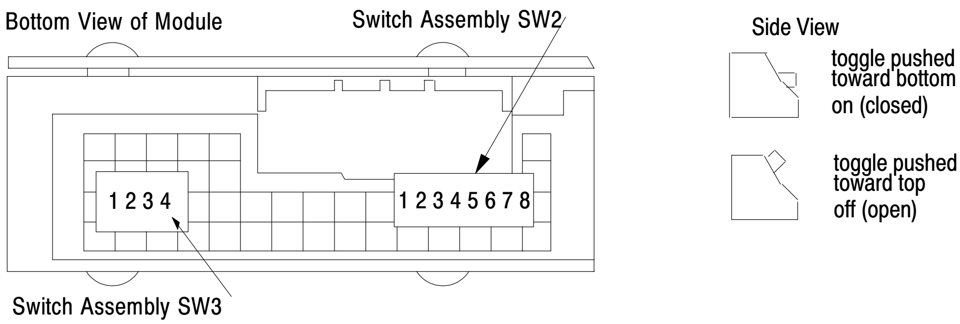
To select:	Set switch:	To:
DH+ Station Number	1 through 6	(see below)
Switch 7 not used	7	off
Scanner mode	8	off
Adapter	8	on

DH+ Station Number	Switch					
	1	2	3	4	5	6
0	on	on	on	on	on	on
1	off	on	on	on	on	on
2	on	off	on	on	on	on
3	off	off	on	on	on	on
4	on	on	off	on	on	on
5	off	on	off	on	on	on
6	on	off	off	on	on	on
7	off	off	off	on	on	on
10	on	on	on	off	on	on
11	off	on	on	off	on	on
12	on	off	on	off	on	on
13	off	off	on	off	on	on
14	on	on	off	off	on	on
15	off	on	off	off	on	on
16	on	off	off	off	on	on
17	off	off	off	off	on	on
20	on	on	on	on	off	on
21	off	on	on	on	off	on
22	on	off	on	on	off	on
23	off	off	on	on	off	on
24	on	on	off	on	off	on
25	off	on	off	on	off	on
26	on	off	off	on	off	on
27	off	off	off	on	off	on
30	on	on	on	off	off	on
31	off	on	on	off	off	on
32	on	off	on	off	off	on
33	off	off	on	off	off	on
34	on	on	off	off	off	on
35	off	on	off	off	off	on
36	on	off	off	off	off	on
37	off	off	off	off	off	on

DH+ Station Number	Switch					
	1	2	3	4	5	6
40	on	on	on	on	on	off
41	off	on	on	on	on	off
42	on	off	on	on	on	off
43	off	off	on	on	on	off
44	on	on	off	on	on	off
45	off	on	off	on	on	off
46	on	off	off	on	on	off
47	off	off	off	on	on	off
50	on	on	on	off	on	off
51	off	on	on	off	on	off
52	on	off	on	off	on	off
53	off	off	on	off	on	off
54	on	on	off	off	on	off
55	off	on	off	off	on	off
56	on	off	off	off	on	off
57	off	off	off	off	on	off
60	on	on	on	on	off	off
61	off	on	on	on	off	off
62	on	off	on	on	off	off
63	off	off	on	on	off	off
64	on	on	off	on	off	off
65	off	on	off	on	off	off
66	on	off	off	on	off	off
67	off	off	off	on	off	off
70	on	on	on	off	off	off
71	off	on	on	off	off	off
72	on	off	on	off	off	off
73	off	off	on	off	off	off
74	on	on	off	off	off	off
75	off	on	off	off	off	off
76	on	off	off	off	off	off
77	off	off	off	off	off	off

Adapter-Mode Processors—SW2 in a PLC-5 or Scanner Module

Set SW2 switch assembly switches for an adapter-mode PLC-5 processor in a PLC-5 processor or scanner module. Set switches 2 through 8 for number of words communicated from the host processor to the adapter processor, for the I/O group, and for the rack number of the I/O group of the adapter processor, respectively. Switch 1 is unused.

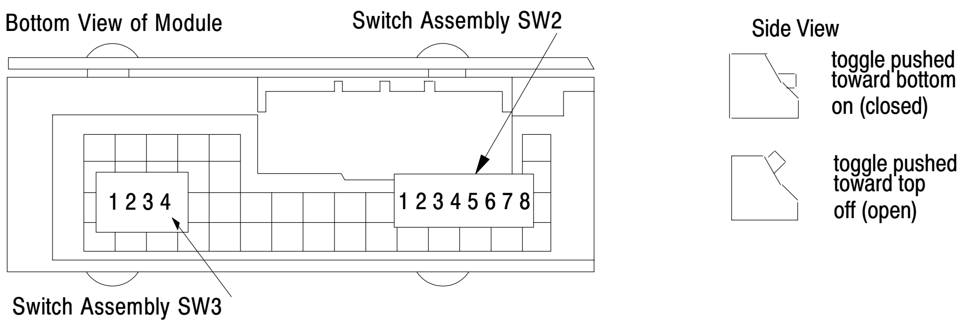


If you want:	Set switch:	To:
Switch 1 is always unused.	1	off
The host processor to use **8 words** to communicate with the adapter PLC-5 processor	2	off
The host processor to use **4 words** to communicate with the adapter PLC-5 processor	2	on
First I/O group to be 0	3	on
First I/O group to be 4	3	off
Select the I/O rack number of the adapter PLC-5 processor	4 through 8	see table below

Rack	4	5	6	7	8	Rack	4	5	6	7	8
01	on	on	on	on	off	15	on	off	off	on	off
02	on	on	on	off	on	16	on	off	off	off	on
03	on	on	on	off	off	17	on	off	off	off	off
04	on	on	off	on	on	20	off	on	on	on	on
05	on	on	off	on	off	21	off	on	on	on	off
06	on	on	off	off	on	22	off	on	on	off	on
07	on	on	off	off	off	23	off	on	on	off	off
10	on	off	on	on	on	24	off	on	off	on	on
11	on	off	on	on	off	25	off	on	off	on	off
12	on	off	on	off	on	26	off	on	off	off	on
13	on	off	on	off	off	27	off	on	off	off	off
14	on	off	off	on	on						

Adapter-Mode Processors—SW2 in a PLC-2/20, -2/30, or Sub I/O Scanner Module System

Set SW2 switch assembly switches for an adapter-mode PLC-5 processor in a PLC-2/20 or -2/30 processor system or sub I/O scanner module system. Set switches 2 through 8 for the number of words communicated from host processor to adapter processor, for the I/O group, and for the rack number of the I/O group of the adapter processor, respectively. Switch 1 is unused.

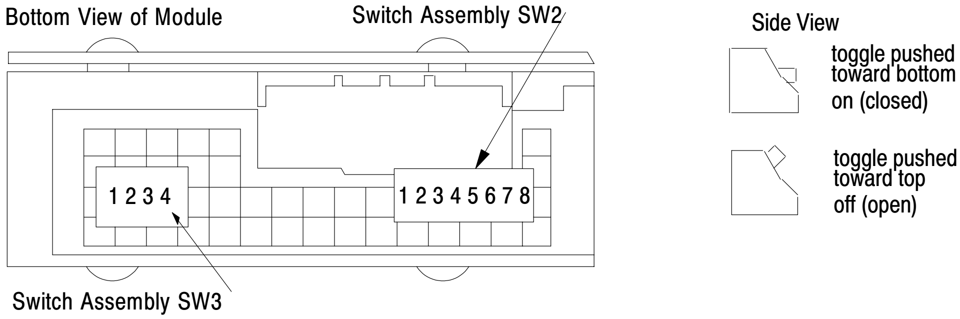


If you want:	Set switch:	To:
Switch 1 is always unused.	1	off
The host processor to use **8 words** to communicate with the adapter PLC-5	2	off
The host processor to use **4 words** to communicate with the adapter PLC-5	2	on
First I/O group to be 0	3	on
First I/O group to be 4	3	off
Select the I/O rack number of the adapter PLC-5 processor	4 through 8	see below

Rack	4	5	6	7	8
01	on	on	on	on	on
02	on	on	on	on	off
03	on	on	on	off	on
04	on	on	on	off	off
05	on	on	off	on	on
06	on	on	off	on	off
07	on	on	off	off	on

Adapter-Mode Processors—SW2 in a PLC-3 or PLC-5/250 System with 8-Word Groups

Set SW2 switch assembly switches for an adapter-mode PLC-5 processor in a PLC-3 or PLC-5/250 processor system. Set switch 2 for the number of words communicated from host processor to the adapter processor. Set switches 3 through 8 for the I/O rack number of the adapter processor. Switch 1 is unused.



If you want:	Set switch:	To:
Switch 1 is always unused.	1	off
The host processor to use **8 words** to communicate with the adapter PLC-5 processor	2	off
Select the I/O rack number of the adapter PLC-5 processor	3 through 8	see below

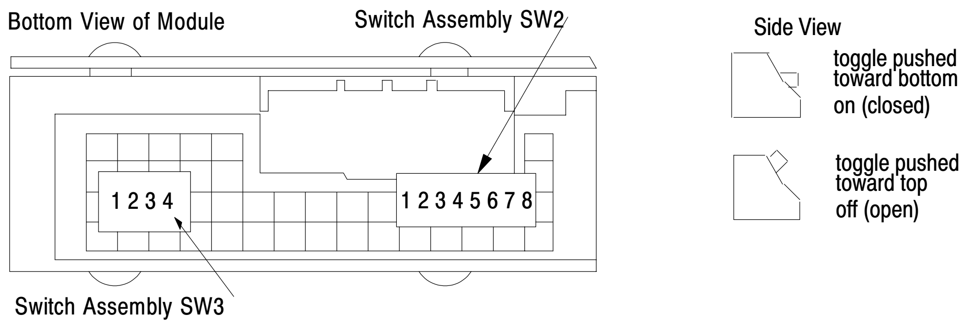
I/O Rack No.	Switch 3	Switch 4	Switch 5	Switch 6	Switch 7	Switch 8
0	on	on	on	on	on	on
1	on	on	on	on	on	off
2	on	on	on	on	off	on
3	on	on	on	on	off	off
4	on	on	on	off	on	on
5	on	on	on	off	on	off
6	on	on	on	off	off	on
7	on	on	on	off	off	off
10	on	on	off	on	on	on
11	on	on	off	on	on	off
12	on	on	off	on	off	on
13	on	on	off	on	off	off
14	on	on	off	off	on	on
15	on	on	off	off	on	off
16	on	on	off	off	off	on
17	on	on	off	off	off	off
20	on	off	on	on	on	on
21	on	off	on	on	on	off
22	on	off	on	on	off	on
23	on	off	on	on	off	off
24	on	off	on	off	on	on
25	on	off	on	off	on	off

I/O Rack No.	Switch 3	Switch 4	Switch 5	Switch 6	Switch 7	Switch 8
26	on	off	on	off	off	on
27	on	off	on	off	off	off
30	on	off	off	on	on	on
31	on	off	off	on	on	off
32	on	off	off	on	off	on
33	on	off	off	on	off	off
34	on	off	off	off	on	on
35	on	off	off	off	on	off
36	on	off	off	off	off	on
37	on	off	off	off	off	off
40	off	on	on	on	on	on
41	off	on	on	on	on	off
42	off	on	on	on	off	on
43	off	on	on	on	off	off
44	off	on	on	off	on	on
45	off	on	on	off	on	off
46	off	on	on	off	off	on
47	off	on	on	off	off	off
50	off	on	off	on	on	on
51	off	on	off	on	on	off
52	off	on	off	on	off	on

I/O Rack No.	Switch 3	Switch 4	Switch 5	Switch 6	Switch 7	Switch 8
53	off	on	off	on	off	off
54	off	on	off	off	on	on
55	off	on	off	off	on	off
56	off	on	off	off	off	on
57	off	on	off	off	off	off
60	off	off	on	on	on	on
61	off	off	on	on	on	off
62	off	off	on	on	off	on
63	off	off	on	on	off	off
64	off	off	on	off	on	on
65	off	off	on	off	on	off
66	off	off	on	off	off	on
67	off	off	on	off	off	off
70	off	off	off	on	on	on
71	off	off	off	on	on	off
72	off	off	off	on	off	on
73	off	off	off	on	off	off
74	off	off	off	off	on	on
75	off	off	off	off	on	off
76	off	off	off	off	off	on

Adapter-Mode Processors—SW2 in a PLC-3 or PLC-5/250 System with 4-Word Groups

Set SW2 switch assembly switches for an adapter-mode PLC-5 processor in a PLC-3 or PLC-5/250 processor system. Set switch 2 for the number of words communicated from the host processor to the adapter processor. Set switch 3 for I/O group. Set switches 4 through 8 for the I/O rack number of the adapter processor. Switch 1 is unused.



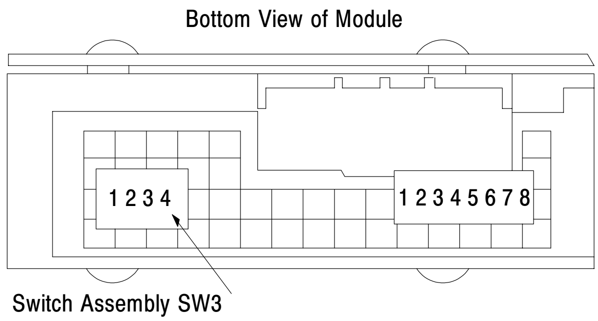
If you want:	Set switch:	To:
Switch 1 is always unused.	1	off
The host processor to use **4 words** to communicate with the adapter PLC-5 processor	2	on
First I/O group to be 0	3	on
First I/O group to be 4	3	off
Select the I/O rack number of the adapter PLC-5 processor	4 through 8	see below

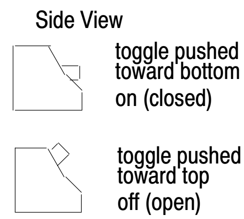
I/O Rack Number	Switch 4	Switch 5	Switch 6	Switch 7	Switch 8
0	on	on	on	on	on
1	on	on	on	on	off
2	on	on	on	off	on
3	on	on	on	off	off
4	on	on	off	on	on
5	on	on	off	on	off
6	on	on	off	off	on
7	on	on	off	off	off
10	on	off	on	on	on
11	on	off	on	on	off
12	on	off	on	off	on
13	on	off	on	off	off
14	on	off	off	on	on
15	on	off	off	on	off
16	on	off	off	off	on
17	on	off	off	off	off

I/O Rack Number	Switch 4	Switch 5	Switch 6	Switch 7	Switch 8
20	off	on	on	on	on
21	off	on	on	on	off
22	off	on	on	off	on
23	off	on	on	off	off
24	off	on	off	on	on
25	off	on	off	on	off
26	off	on	off	off	on
27	off	on	off	off	off
30	off	off	on	on	on
31	off	off	on	on	off
32	off	off	on	off	on
33	off	off	on	off	off
34	off	off	off	on	on
35	off	off	off	on	off
36	off	off	off	off	on
37	off	off	off	off	off

SW3

Set SW3 switch assembly switches to terminate either a DH+ link or a remote I/O link. Switches 3 and 4 are unused.



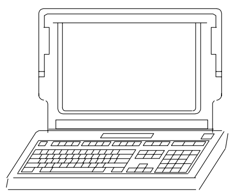


If the processor is:	Set switch:	To:
An end device on the remote I/O link	1	on
Not an end device on the remote I/O link	1	off
An end device on the DH+ link	2	on
Not an end device on the DH+ link	2	off
Switch 3 unused	3	off
Switch 4 unused	4	off

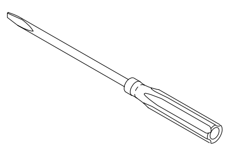
Design Worksheets

Conventions Used in These Worksheets

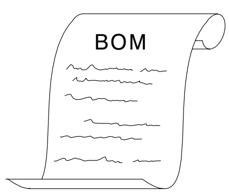
The following symbols are printed in the top left corner of the worksheets. The symbols indicate whether programmers or installers will need the completed worksheets. Use the symbols as a way of organizing the completed worksheets for the appropriate user.



indicates a worksheet that provides information for a **programmer**

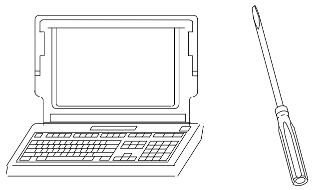


indicates a worksheet that provides information for an **installer**



indicates a worksheet that provides information for a **bill of materials (BOM)**

Important: You may need to make multiple copies of some worksheets to record all of your system requirements.



Prepare a Functional Specification

For more information on:	See:
Functional specifications	Classic 1785 PLC-5 Programmable Controllers User Manual, publication 1785-6.2.1, Chapter 1: Designing Systems Preparing Your Functional Specification

1. Divide your manufacturing process into functional areas.

2. Make a copy of the reverse side of this worksheet for each functional area.

3. For each functional area, document the following:

Information to Document	Example(s)
Inputs	Actions and signals, ranges, quantities, timing, tolerance, units of measure, validation requirements, possible errors, and error responses
Outputs	Quantities, units of measure, timing, tolerances, ranges, validation methods, method or reporting invalid outputs, locations, methods of data output, physical requirements
Performance requirements	Accuracy; maximum and minimum transition times; interface timing; operator response timing; adherence to standards such as IEEE, ANSI
Interfaces	Operator, software, hardware
Failure modes and recovery methods	Fault routines
Security requirements	Operator access, alarming, etc.
Maintenance requirements	Documentation, spare parts, accessibility

4. Use the information on these worksheets to develop a complete functional specification.

Functional Area:

Inputs:

Outputs:

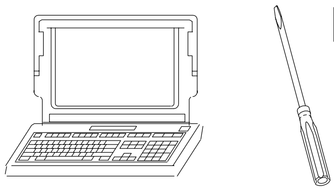
Performance Requirements:

Interfaces:

Failure Modes and Recovery Methods:

Security Requirements:

Maintenance Requirements:



Determine Control Strategy

For more information on:	See:
Control strategy Remote I/O scanner mode Extended-local I/O scanner mode Remote I/O adapter mode	Classic 1785 PLC-5 Programmable Controllers User Manual, publication 1785-6.2.1, Chapter 1: Designing Systems Using the Classic PLC-5 Processor as a Remote I/O Scanner Using the Classic PLC-5 Processor as a Remote I/O Adapter
Choosing communication	Classic 1785 PLC-5 Programmable Controllers Design Manual, publication 1785-6.2.1, Chapter 5

1. Answer the following questions when you begin planning your control strategy:

What will be controlled together?

What will be controlled separately?

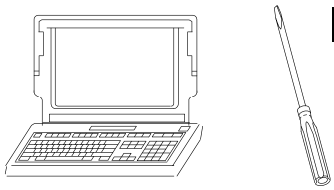
Will control devices communicate as peers (network) or in a hierarchy (master/slave)?

What will be controlled via a remote I/O link?

Which processes will be controlled by a classic 1785 PLC-5 processor?

What are the environmental and safety concerns for your system?

2. Use text and illustrations to lay out and describe your strategy.



Identify Chassis Locations

1. Make a copy of this worksheet for each of your functional areas.

2. For each functional area, determine the number of chassis by using the table below.

Functional Area:	
Category	**Number of Chassis**
Each functional area requires at least one chassis.	1
Add an additional chassis for each case where I/O at this functional area require different:	
power disconnects	
ac phases	
logical or functional groupings	
Total chassis for this functional area:	

3. Assign a unique chassis number to each of the chassis, and record all assigned numbers below.

Chassis _______________ Chassis _______________

Chassis _______________ Chassis _______________

Chassis _______________ Chassis _______________

Chassis _______________ Chassis _______________

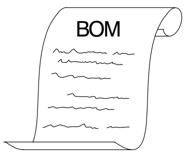
Chassis _______________ Chassis _______________

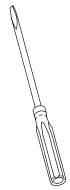
Chassis _______________ Chassis _______________

Chassis _______________ Chassis _______________

Chassis _______________ Chassis _______________

Chassis _______________ Chassis _______________





Select Module Types and List I/O Points

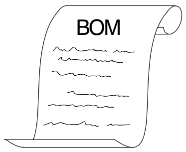
For more information on:	See:
Selecting I/O modules Selecting I/O points	Classic 1785 PLC-5 Programmable Controllers User Manual, publication 1785-6.2.1, Chapter 2: Selecting I/O Modules Selecting I/O Adapter Modules
I/O module catalog numbers	Automation Products Catalog, publication AP 100, Section 3: Input/Output

1. Make a copy of the reverse side of this worksheet for each of your chassis.
2. For each chassis, list each discrete, analog, and specialty I/O module and its electrical characteristics. Use the table below to determine which characteristics to list.

If you choose this type of I/O module:	List these characteristics:
Discrete input module	Voltage Special requirements: – Isolation – Proximity switch – Source or sink – Fast response – TTL
Discrete output module	Voltage Current Special requirements: – Isolation – Protection (detection of failed triacs) – TTL – High current switching
Analog input module	Voltage or current range Resolution required Single-ended or differential Special requirements – Thermocouple – RTD – Isolation
Analog output module	Voltage or current range Resolution required Special requirements: – PID – Isolation
Specialty or communication I/O modules (including block I/O modules)	Voltage or current range Resolution required Noise/distance limits Single-ended or differential Special requirements – Thermocouple – RTD – PID – Isolation

Chassis Number: ____________________

Input or Output	I/O Module Type	I/O Module Catalog Number	Time Critical Yes / No	Voltage or Range	Current or Range	Number of Points	Required Resolution (analog only)	Single-Ended or Differential (analog only)	Special Requirements



Total I/O Module Requirements

1. Make a copy of this worksheet for each of your chassis.

2. For each of the I/O modules identified on the chassis' Select Module Types and List I/O Points worksheet on pages 7 and 8, perform the following steps to determine the total number that you need.

   3. In column **A**, list the module's catalog number.

   4. In column **B**, total the number of I/O points for the module.

   5. In column **C**, enter the maximum number of I/O points available per module.

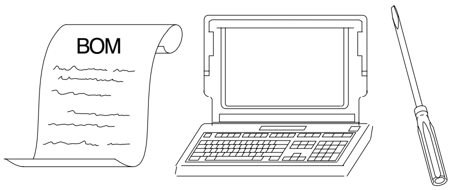
   6. In column **D**, calculate the total number of these modules that you will need for this chassis by dividing column B by column C.

   7. In column **E**, enter the number of spare modules that you may need for future expansion of this chassis.

   8. In column **F**, enter the total number of these modules that you need for this chassis.

Chassis Number: _________________

A I/O Module Cat. No.	B Total I/O Points	C I/O Points / Module	D I/O Modules Required (B / C)	E Spare I/O modules	F Total I/O modules



Assign I/O Modules to Chassis and Assign Addresses

For more information on:	See:
Assigning I/O modules to chassis Selecting addressing mode Assigning addresses Addressing complementary I/O	Classic 1785 PLC-5 Programmable Controllers User Manual, publication 1785-6.2.1, Chapter 4: Placing I/O Modules in Chassis Choosing the Addressing Mode Assigning Racks Addressing Complementary I/O
Selecting the I/O chassis	Classic 1785 PLC-5 Programmable Controllers Design Manual, publication 1785-6.2.1, Chapter 2: Selecting I/O Chassis
Current requirements to I/O modules	Automation Products Catalog, publication AP 100, Section 3: Input/Output

9. Make a copy of the reverse side of this worksheet for each of your chassis, and use it to record your responses to items 2 through 8.

10. Indicate the addressing mode for each chassis. Use the table below to guide your selection.

If the densest I/O module in the chassis is:	And you want to:	Then choose:
8-point		2-slot addressing
16-point	Assign any mix of modules in adjacent module slots	1-slot addressing
	Make full use of I/O capacity	2-slot addressing
32-point	Assign any mix of modules in adjacent module slots	1/2-slot addressing
	Make full use of I/O capacity	1-slot addressing

11. Indicate the chassis size. Use the table below to guide your selection.

If you need to:	And are:	Then consider:
Reduce spare parts	Expanding your system	Standard size you now use
	Installing a new system	One size using the guidelines listed below
Fit space requirements	Limited to 9 inches	4-slot chassis
	Limited to 14 inches	8-slot chassis
	Limited to 19 inches	12-slot chassis
	Limited to 24 inches	16-slot chassis
Minimize scan time		Largest chassis containing processor
Minimize cost per slot		Largest chassis consistent with decisions above
Accommodate expansion		

12. Indicate whether a processor or an adapter is in the left-most slot.

13. Indicate whether you are using this chassis for complementary I/O.

14. Write the module type in each available slot on the chassis diagram.

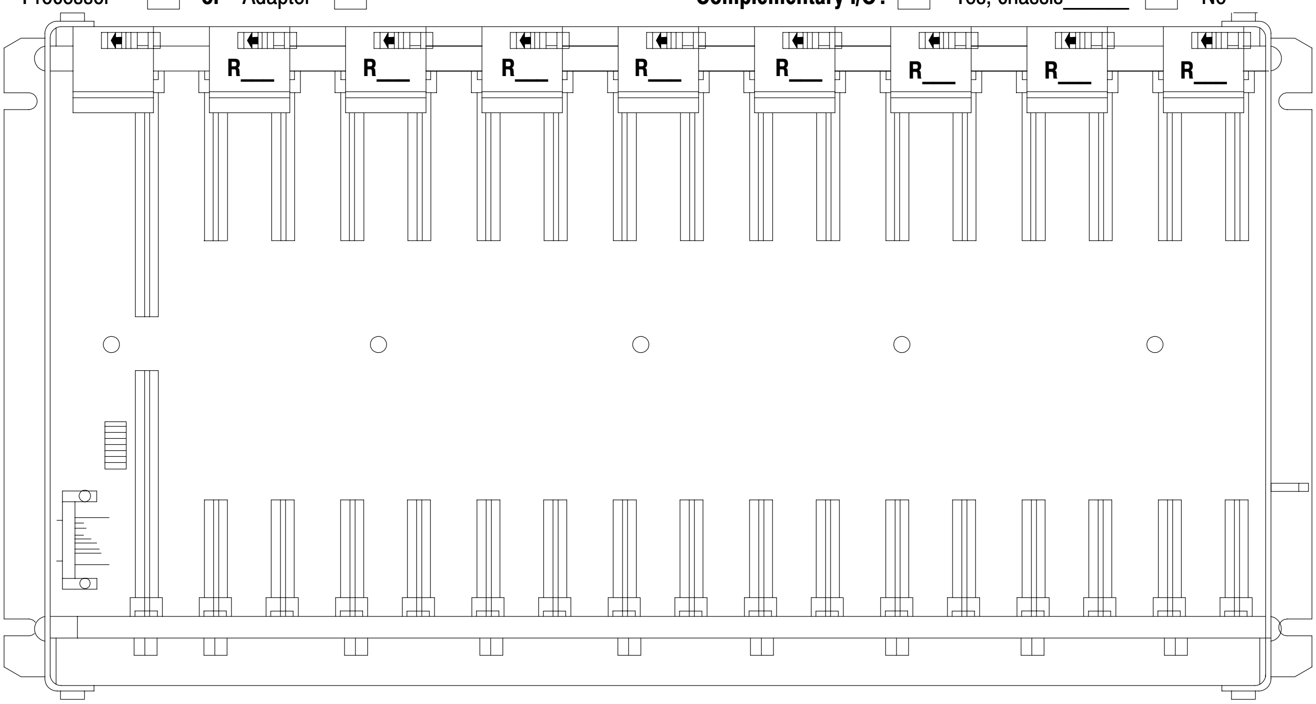
15. Assign rack numbers, group numbers, and number of points per group.

16. Indicate the current requirement for each module.

Chassis Number: ________________ **Addressing Mode:** ☐ 2-slot ☐ 1-slot ☐ 1/2-slot

Indicate Chassis Size:	☐ 1771-A1B 4-slot chassis	☐ 1771-A2B 8-slot chassis	☐ 1771-A3B, or 1771-A3B1 12-slot chassis	☐ 1771-A4B 16-slot chassis

Processor ☐ **or** Adapter ☐ **Complementary I/O?** ☐ Yes, chassis______ ☐ No



Identify Groups and I/O Points:

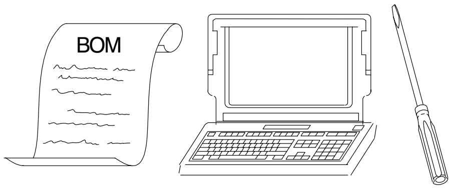
Slot 1	**Slot 2**	**Slot 3**	**Slot 4**	**Slot 5**	**Slot 6**	**Slot 7**	**Slot 8**
G ___	G ___	G ___	G ___	G ___	G ___	G ___	G ___
☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07
☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17
G __	G __	G __	G __	G __	G __	G __	G __
☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07
☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17

Slot 9	**Slot 10**	**Slot 11**	**Slot 12**	**Slot 13**	**Slot 14**	**Slot 15**	**Slot 16**
G ___	G ___	G ___	G ___	G ___	G ___	G ___	G ___
☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07
☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17
G __	G __	G __	G __	G __	G __	G __	G __
☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07	☐ 00–07
☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17	☐ 10–17

List Current Required for Each Module in this Chassis:

____	____	____	____	____	____	____	____
____	____	____	____	____	____	____	____

Total Current Draw for I/O Modules in this Chassis = _____



Select Adapter Modules

For more information on:	See:
Selecting the I/O adapter modules	Classic 1785 PLC-5 Programmable Controllers User Manual, publication 1785-6.2.1, Chapter 2: Selecting I/O Adapter Modules

17. In column **A** of the table below, indicate the chassis number in which you will place each adapter module.

18. In column **B**, indicate the name/type of each adapter module.

19. In column **C**, list any special requirements for the adapter module.

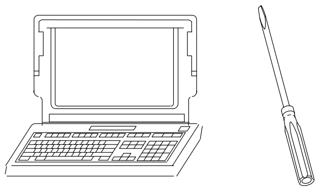
20. In column **D**, indicate the current draw requirements for the adapter module.

A Chassis Number	B Adapter Module	C Special Requirements	D Current Draw

21. In the table below, list any device other than an adapter module that you are connecting to a remote I/O link, list its chassis number, and list any special requirements. For example:

- PLC-5 processor in adapter mode
- PLC-5/250 remote scanner in adapter mode
- PLC interface module for digital ac and dc drives
- Remote I/O adapter for Bulletin 1336 drives
- RediPANEL pushbutton and keypad modules
- Dataliner
- PanelView (see operator interface)
- Option module (for T30 plant-floor terminal)
- 8600 CNC with remote I/O adapter option
- CVIM in adapter mode
- Pro-Spec 6000 Fastening System with remote I/O adapter option
- 1747-DCM module (to SLC-500 rack)
- 1771-DCM module
- 1771-GMF robot (remote I/O interface module)

Device	Attached to Chassis Number	Special Requirements

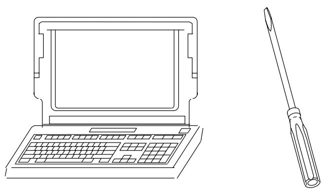


Place System Hardware

For more information on:	See:
	Classic PLC-5 Programmable Controllers User Manual, publication 1785-6.2.1, Chapter 3:
Determining proper environment	Determining the Proper Environment
Enclosures	Protecting Your Processor
Raceway layout	Laying Out Your Cable Raceway
Cabling and wiring	Planning Cabling
Mounting	Laying Out the Backpanel Spacing
Grounding	Grounding Configuration

Sketch a layout of your system that indicates the following:

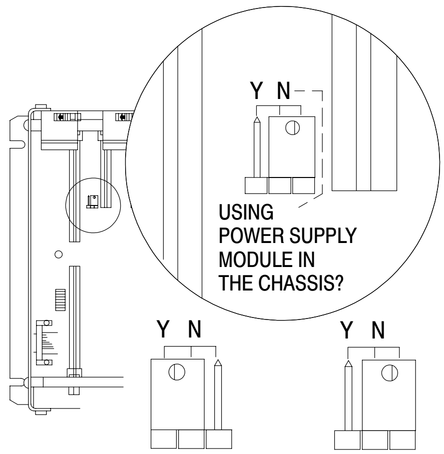
- proper environment
- enclosures
- mounting
- raceway layout
- cabling and wiring
- grounding



Configure Switch Settings

For more information on:	See:
Configuring switches	Classic PLC-5 Programmable Controllers User Manual, publication 1785-6.2.1, Appendix A

Record switch setting choices on a copy of this worksheet. You might need to return to the worksheet several times as you complete your system design.



Set Y when you install a power supply module in the chassis.

Set N when you use an external power supply.

1. Locate the chassis configuration plug (between the first two left most slots of the chassis).

2. Set the I/O chassis configuration plug.

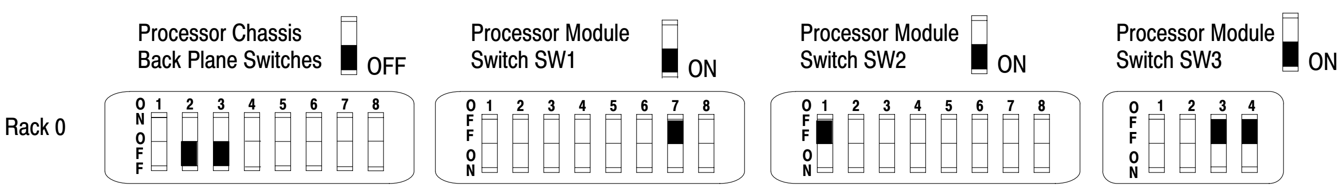
   The default setting is N (not using a power supply module in the chassis).

IMPORTANT: You cannot power a single I/O chassis with both a power supply module and an external power supply.

17075

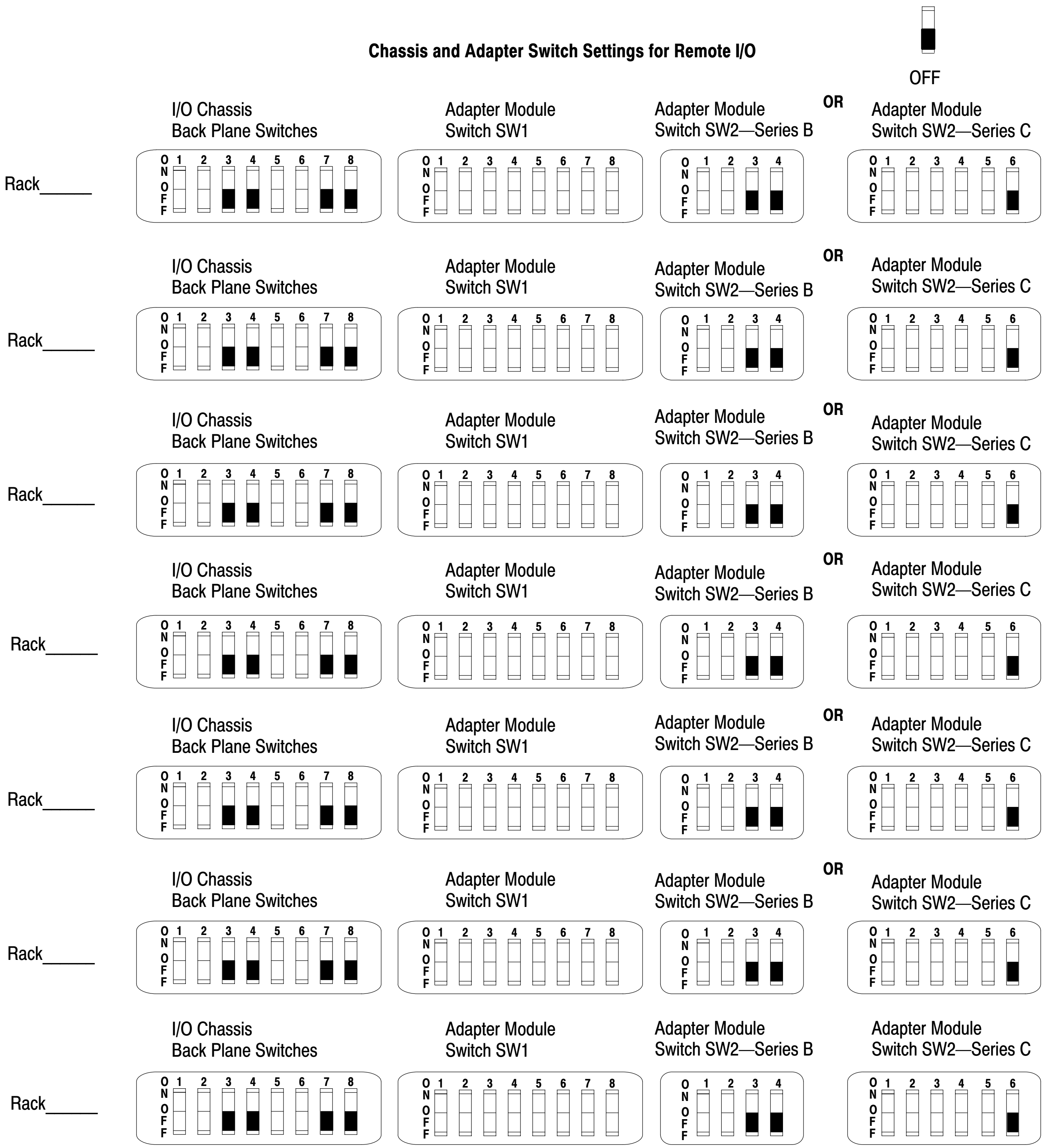
Rack 0 Chassis and Processor Switch Settings

For PLC-5/10, -5/12, -5/15, -5/25 processors:

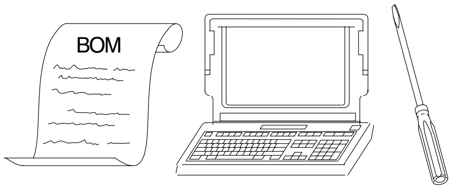


Notes: **Switches shown in black are not used, but they must be set to the positions indicated.**

Chassis and Adapter Switch Settings for Remote I/O



Note: Switches previously marked in black are not used, but must be set to the positions indicated.



Determine Communication Requirements

For more information on:	See:
Identifying processor connectors/channels Configuring Channel 0 (serial ASCII port) Choosing a DH+ link	Classic 1785 PLC-5 Programmable Controllers User Manual, publication 1785-6.2.1, Chapter 5: Identifying Classic PLC-5 Processor Channels/Connectors Using Channel 0 Configuring a DH+ Link
Selecting DH+ cabling, layout Selecting processor connectors/channels cabling	Classic 1785 PLC-5 Programmable Controllers Design Manual, publication 1785-6.2.1, Chapter 3: Laying Out Your Cable Raceway Planning Cabling
Selecting termination resistors	Classic PLC-5 Programmable Controllers Design Manual, publication 1785-6.2.1, Chapter 2: Selecting Link Terminators
Defining DH+ station addresses	Classic PLC-5 Family Programmable Controller Hardware Installation Manual, publication 1785-6.6.1

1. Make a copy of appropriate pages of this worksheet for each of your processors.

2. Identify communication modes and network selections.

3. Indicate channel configurations and DH+ station addresses.

4. List racks attached to each channel/connector configured for remote I/O scanner or adapter mode.

5. Identify DH+ link cable layout (daisy chain or trunkline/dropline).

6. Select your data link cables. Circle or highlight your selections in the following tables.

for remote I/O link

With this transmission rate:	Select this maximum cable length (1770-CD cable):
57.6 kbaud	10,000 ft
115.2 kbaud	5,000 ft
230.4 kbaud	2,500 ft

for Ethernet link

If you need this:	Select this catalog number:
Thick-wire 2.0 m (6.5 ft) transceiver cable	5810-TC02/A
Thick-wire 15.0 m (49.2 ft) transceiver cable	5810-TC15/A
Thin-wire transceiver, and 2.0 m (6.5 ft) cable	5810-TAS/A (kit)
Thin-wire transceiver, and 15.0 m (49.2 ft) cable	5810-TAM/A (kit)
Thick-wire transceiver, and 2.0 m (6.5 ft) cable	5810-TBS/A (kit)
Thick-wire transceiver, and 15.0 m (49.2 ft) cable	5810-TBM/A (kit)

7. Terminate a DH+ or remote I/O link by setting switch assembly 3.

PLC-5/10 Processor

List information:

List local rack numbers:

DH+ Station Address ______________

Record any additional information about your communication mode and network selection.

PLC-5/12, -5/15, or -5/25 Processor

SW1 settings

☐ Scanner

☐ Adapter

SW2 rack address setting ______

First I/O group ______

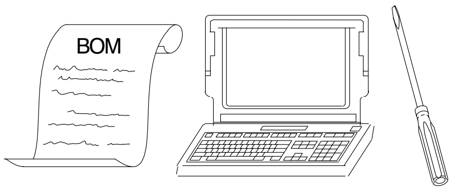
Number of words to transfer ______

List information:

List rack numbers (if configured for scanner):

DH+ Station Address ______

Record any additional information about your communication mode and network selection.



Select a Classic PLC-5 Processor

For more information on:	See:
Selecting a processor Selecting optional memory modules Selecting a replacement battery	Classic 1785 PLC-5 Programmable Controllers User Manual, publication 1785-6.2.1, Chapter 2: Choosing a Classic PLC-5 Processor for Your Application Selecting Memory Modules Selecting a Replacement Battery
Selecting a backup system	PLC-5 Backup Communication Module User Manual, publication 1785-6.5.4

8. Make copies of both sides of this worksheet for each chassis that requires a processor.

9. For each chassis that requires a processor, use the table below to help you determine which processor to use.

Total Memory Required	Total Number of Racks	Total Number of Chassis	Need a Serial Port?	Required Program Scan Time	Total Number of DH+ Ports	Total Number of Remote I/O Ports

10. Record your Classic PLC-5 processor choice below.

Classic PLC-5 processor is: ______________________________

It will reside in chassis no.: ______________________________

The current required is: ______________________________

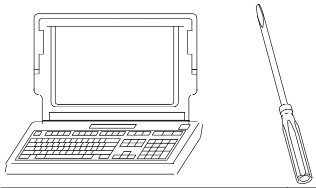
11. Select additional memory for your classic PLC-5 processor. Circle or highlight your selection in the table below.

Nonvolatile Memory Backup (EEPROM)		RAM Memory (CMOS)	
Words	**Catalog Number (and Processor)**	**Words**	**Catalog Number (and Processor)**
8 K	1785-MJ (PLC-5/10, -5/12, -5/15, -5/25)	4 K	1785-MR (PLC-5/15 and -5/25)
16 K	1785-MK (PLC-5/25)	8 K	1785-MS (PLC-5/15 and -5/25)

12. Select a 1770-XY, AA lithium replacement battery for your classic PLC-5 processor.

13. Select a backup system for your classic PLC-5 processor if required. A classic PLC-5 processor backup system contains **two** of each of the following hardware components. Indicate your selections below.

- PLC-5 processor module (PLC-5/15 or -5/25 only)
- 1785-BCM module (for two channels)
- 1785-BEM module (for two additional channels)
- power supply
- local chassis



Select Power Supplies

For more information on:	See:
Selecting power supplies	Classic 1785 PLC-5 Programmable Controllers User Manual, publication 1785-6.2.1, Chapter 2: Selecting Power Supplies
Selecting power supply cables	Automation Products Catalog, publication AP 100

1. Make a copy of this worksheet for each of your chassis.

2. Refer to the following worksheets for the values that you need to complete the formula to select a power supply.

- Assign I/O Modules to Chassis and Assign Addresses worksheet for total I/O current draw
- Select Adapter Module **o**r Select Classic PLC-5 Processor worksheet for current draw

3. Perform the following steps to calculate the total current required for chassis number ____________ and to select a power supply.

On line **A** below, record the total backplane current draw for all I/O modules in the chassis. If you leave slots available in your chassis for future expansion, add current draw for future I/O modules.

On line **B** below, record the current draw required for the classic PLC-5 processor or the adapter module in the chassis.

On line **C** below, record the total current required from a power supply for that chassis.

A—Total I/O backplane current ____________________
B—PLC-5 processor/adapter module current + ____________________
C—Total backplane current required = ____________________

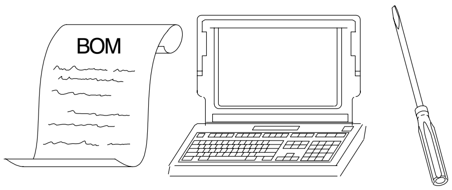
4. Choose your power supply dependent on the input voltage requirement and the total backplane current required (line C above). Two types of power supplies are:

- power supply modules—located in the same chassis as the PLC-5 processor or adapter module
- power supplies—located external to the chassis containing the PLC-5 processor or adapter module

5. Record your power supply and cable choice below.

Power supply for this chassis is: ________________________________
Cable for this power supply is: ________________________________

Important: You cannot use an external power supply and a power supply module to power the same chassis; they are not compatible.



Choose a Programming Terminal

For more information on:	See:
Selecting a programming terminal Selecting cables for a programming terminal	Classic 1785 PLC-5 Programmable Controllers User Manual, publication 1785-6.2.1, Chapter 2: Choosing Programming Terminals Choosing Cables

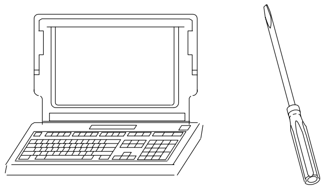
6. Make a copy of this worksheet for each of your PLC-5 processors.

7. Select a programming terminal for your Classic PLC-5 processor __________ located in chassis number _____________. Circle or highlight your selection in the table below.

Programming Terminal	Operating System
• 6160-T53 • 6160-T60 • 6160-T70	• DOS 3.2, 3.3, 4.x, 5.0, or 6.0

8. Select a communication device and cables. Circle or highlight your selection in the table below.

If you have this device:	With this communication device:	Use this cable:
PLC-5/10, -5/12, -5/15, or -5/25	1784-KT, -KT2 1784-KL, -KL/B	1784-CP
	1784-KTK1	1784-CP5
	1784-PCMK	1784-PCM5
6160-T60, 6160-T70, 6121 IBM PC/AT (or compatible)	1785-KE	1784-CAK
1784-T47, 6123, 6124 IBM PC/XT (or compatible)	1785-KE	1784-CXK
6120, 6122	1785-KE	1784-CYK



Select Programming Terminal Configuration

1. Make a copy of this worksheet for each of your programming terminals.

2. Document your software configuration decisions below for your Classic PLC-5 processor ___________ located in chassis number ______________.

DH+ connections:

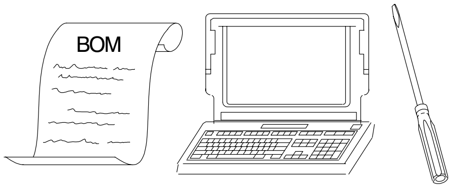
Local access or remote network access ___________________________________

Direct or multi-drop ___________________________________

Type of interface card ___________________________________

Unique station address assigned to terminal ___________________________________

Bit address of KT board in programming terminal ___________________________________



Select Operator Interface

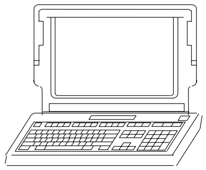
For more information on:	See:
Selecting an operator interface	Classic 1785 PLC-5 Programmable Controllers User Manual, publication 1785-6.2.1, Chapter 2: Selecting an Operator Interface

Select your operator interface using the table that follows.

3. In column **A**, list the operator interface station.
4. In column **B**, list the required operator interface screens per station.
5. In column **C**, describe the information and control requirements for each screen.
6. In column **D**, list the reports that you want to generate.

A **Operator Interface Station**	**B** **Operator Interface Screen(s)**	**C** **Information and Control Requirements**	**D** **Reports**

A Operator Interface Station	B Operator Interface Screen(s)	C Information and Control Requirements	D Reports



Develop Programming Specifications

1. Use the following table to guide you in developing a programming design specification.

Design Specification Options	Definition
Will you use SFCs?	
What fault routines will you use?	

2. Lay out your data table memory map.

3. Plan your ladder-logic program.

4. What testing will you perform?

Numbers

A

B

C

D

E

R

S

World Headquarters, Allen-Bradley, 1201 South Second Street, Milwaukee, WI 53204 USA, Tel: (1) 414 382-2000 Fax: (1) 414 382-4444

Publication 1785-6.2.1—April 1996
Supersedes Publication 1785–6.2.1—May 1995, Publication 1785-5.2–February 1994

PN 955125-29