Allen-Bradley

Backup Scanner Module

(Catalog Number 1747-BSN)

User Manual

Rockwell Automation

Important User Information

Because of the variety of uses for the products described in this publication, those responsible for the application and use of these products 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. In no event will Rockwell Automation be responsible or liable for indirect or consequential damage resulting from the use or application of these products.

Any illustrations, charts, sample programs, and layout examples shown in this publication are intended solely for purposes of example. Since there are many variables and requirements associated with any particular installation, Rockwell Automation does not assume responsibility or liability (to include intellectual property liability) for actual use based upon 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 Rockwell Automation 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 publication, notes may be used to make you aware of safety considerations. The following annotations and their accompanying statements help you to identify a potential hazard, avoid a potential hazard, and recognize the consequences of a potential hazard:

WARNING Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.

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

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

Summary of Changes

The information below summarizes the changes to this manual since the last printing. To help you find new and updated information in this release of the manual, we have included change bars as shown to the right of this paragraph.

The table below lists the pages or sections containing new or revised information.

For this information:	See
clarification of the user identification DIP switch operation	page 3-3
expanded wiring section with simplified examples for RIO, DH+, and RS-232 network wiring	pages 3-6 through 3-9
G-file configuration using RSLogix 500	pages 5-3 through 5-5
revised ladder logic example for monitoring the System Status Word and Module Status Word	pages 6-10 through 6-12
new information on using block transfer instructions (BTR and BTW)	pages 7-9 through 7-21
revised ladder logic example for transferring data table values to the secondary processor	pages 9-2 through 9-13
updated certifications	page A-1

Preface

Chapter 1

Overview

Chapter 2

Quick Start for Experienced Users

Chapter 3

Installation and Wiring

Chapter 4

Operating Your SLC 500 Backup System

Chapter 5

Configuration and Programming

Chapter 6

Module Control and Status Word

Chapter 7

RIO Block Transfer

Chapter 8

Switchover Considerations

Chapter 9

Programming Techniques

Appendix A

Specifications

Appendix B
M0-M1 Files and G Files

Appendix C
RIO Configuration Worksheet

Glossary

Index

Read this preface to familiarize yourself with the rest of the manual. This preface covers the following topics:

- who should use this manual
- how to use this manual
- related publications
- conventions used in this manual
- Allen-Bradley support

Who Should Use This Manual

Use this manual if you are responsible for designing, installing, programming, or troubleshooting control systems that use Allen-Bradley small logic controllers.

How to Use This Manual

As much as possible, we organized this manual to explain, in a task-by-task manner, how to install, configure, program, operate and troubleshoot a control system using the 1747-BSN Backup Scanner.

Manual Contents

If you want...	See
An overview of the system including backup scanner/SLC interaction, compatible devices, and features	Chapter 1
A quick start guide for experienced users	Chapter 2
Installation and wiring guidelines	Chapter 3
Operating your SLC 500 backup system	Chapter 4
Backup scanner configuration information, I/O file information, and G and M file descriptions	Chapter 5
Information on module control and status word	Chapter 6
RIO block transfer information, M file block transfer buffer layout, block transfer examples	Chapter 7
Switchover considerations	Chapter 8
Programming techniques	Chapter 9
Backup scanner and system specifications	Appendix A
Information and usage of M and G files	Appendix B
Worksheets for configuring the scanner’s I/O image	Appendix C
Definitions of terms used in this manual	Glossary

Related Documentation

The table below provides a listing of publications that contain important information about Allen-Bradley SLC products.

For	Read this document	Document number
An overview of the SLC 500 family of products	SLC 500 System Overview	1747-SO001
A description on how to install and use your *Modular* SLC 500 programmable controller	Modular Hardware Style Programmable Controllers	1747-UM011
A reference manual that contains status file data and instruction set information for the SLC 500 processors.	SLC 500™ Instruction Set Reference Manual	1747-RM001
Installation guide for the Backup Scanner Module	Backup Scanner Module Installation Instructions	1747-5.38
An article on wire sizes and types for grounding electrical equipment	National Electrical Code	Published by the National Fire Protection Association of Boston, MA.
A glossary of industrial automation terms and abbreviations	Allen-Bradley Industrial Automation Glossary	AG-7.1
In-depth information on grounding and wiring Allen-Bradley programmable controllers	Allen-Bradley Programmable Controller Grounding and Wiring Guidelines	1770-4.1

If you would like a manual, you can:

- download a free electronic version from the internet at **www.theautomationbookstore.com**
- purchase a printed manual by:
  - contacting your local distributor or Rockwell Automation representative
  - visiting **www.theautomationbookstore.com** and placing your order
  - calling 1.800.963.9548 (USA/Canada) or 001.330.725.1574 (Outside USA/Canada)

Conventions Used in This Manual

The following conventions are used throughout this manual:

- Bulleted lists (like this one) provide information not procedural steps.
- Numbered lists provide sequential steps or hierarchical information.
- *Italic* type is used for emphasis.

Rockwell Automation Support

Rockwell Automation tests all of our products to ensure that they are fully operational when shipped from the manufacturing facility.

If you are experiencing installation or startup problems, please review the troubleshooting information contained in this publication first. If you need technical assistance to get your module up and running, please contact Customer Support (see the table below); our trained technical specialists are available to help.

If the product is not functioning and needs to be returned, contact your distributor. You must provide a Customer Support case number to your distributor in order to complete the return process.

Phone	United States/Canada	1.440.646.5800
	Outside United States/Canada	You can access the phone number for your country via the Internet: 1. Go to http://support.rockwellautomation.com/ 2. Under *Contacting Customer Support* and Other Countries, click on *Click here*
Internet	Worldwide	Go to http://support.rockwellautomation.com/

Your Questions or Comments on the Manual

If you find a problem with this manual or you have suggestions for how this manual could be made more useful to you, please contact us at the address below:

Rockwell Automation
Automation Control and Information Group
Technical Communication, Dept. A602V
P.O. Box 2086
Milwaukee, WI 53201-2086

Overview

This chapter contains the following information:

- system overview
- how the backup scanner interacts with the SLC processor
- how the backup scanner interacts with adapter modules
- scanner I/O image concepts
- extended node capability
- complementary I/O
- scanner features
- compatible network devices
- backup concepts for the SLC-500 system

System Overview

The 1747-BSN Backup Scanner provides redundancy for:

- Remote I/O (RIO)
- RS232 channel switchover for communications to devices such as operator interface
- DH+

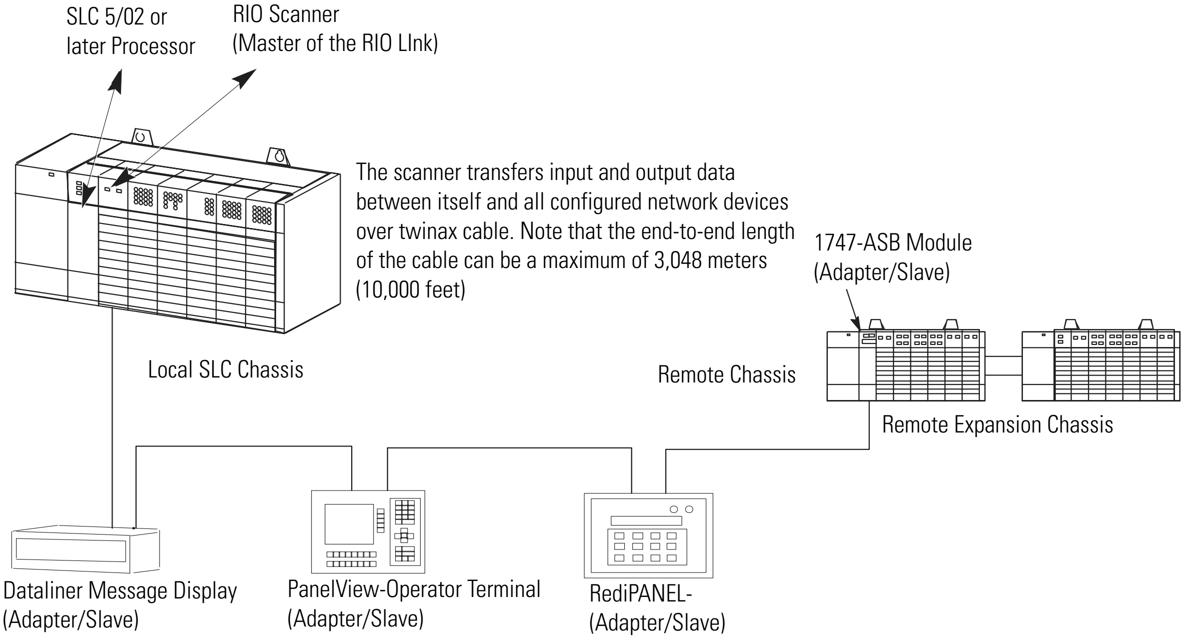
The backup scanner also provides a High Speed Serial Link (HSSL) to write any retentative data from a primary to a secondary processor. In addition, the module has a Local Serial Link (LSL) to pass status information between multiple 1747-BSN modules located in the same chassis (optional).

The 1747-BSN is a complementary set of modules, with one module residing in the primary system and another module in the secondary or backup system. The primary module controls the operation of remote I/O, while the secondary module is available to take over control in the event of a fault in the primary.

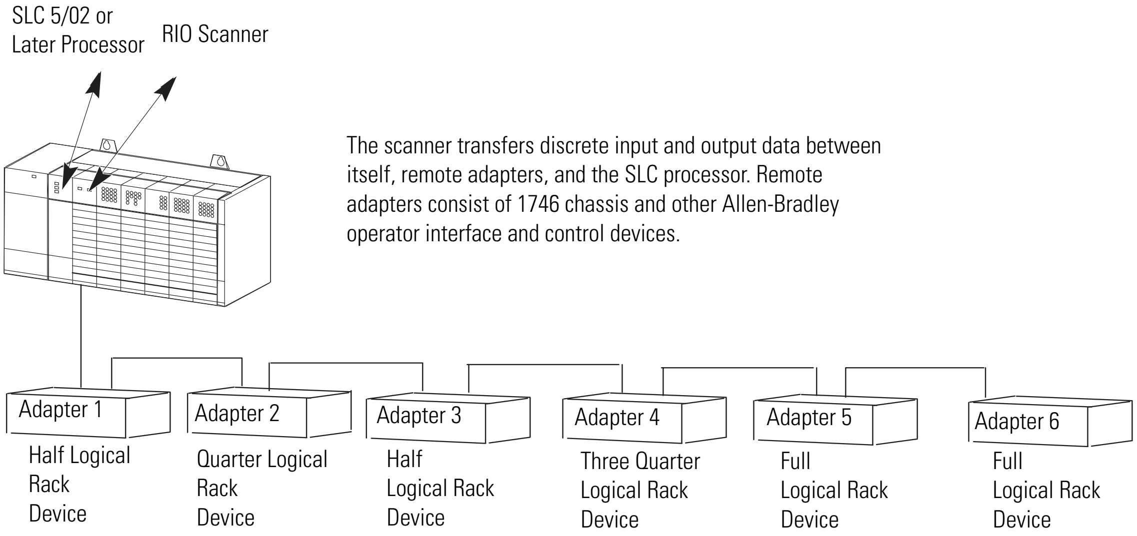
The backup scanner has the capability to switch between two communication channels. The first channel is configurable as RIO or DH+. The second channel is used to switch the RS232/485 channel in order to provide connection for electronic operator interfaces. The DH+/RIO and the RS232/485 channels can be used simultaneously.

The Remote I/O (RIO) scanner, Catalog Number 1747-SN, enables communication between an SLC processor (SLC 5/02 or later) and remotely located (3,048 meters [10,000 feet] maximum) 1746 I/O chassis and other RIO-compatible Allen-Bradley operator interface and control devices. The 1747-SN scanner communicates with remotely located devices using the A-B Remote I/O link. The RIO link consists of a single master (scanner) and multiple slaves (adapters).

Communication between devices occurs over twinax cable with the devices daisy-chained together. The scanner can reside in any slot of the local SLC chassis except for slot 0. The 1747-BSN includes the same functionality as the 1747-SN scanner, but adds the backup features.



The backup scanner can be configured for and transfer a maximum of 4 logical racks of discrete data on the RIO link. The backup scanner provides discrete I/O and block transfers. Configurations allowed are any combination of quarter, half, three quarter, or full logical rack devices.



The scanner transfers discrete input and output data between itself, remote adapters, and the SLC processor. Remote adapters consist of 1746 chassis and other Allen-Bradley operator interface and control devices.

The SLC processor transfers the scanner's 4 logical racks (32 input image and 32 output image words) of discrete remote I/O image data into the SLC input and output image files. You can adjust the size of the scanner input and output image file during configuration of your SLC system so that the scanner only transfers the discrete I/O data your application program requires. Configuration is done through the configuration file (G file). Refer to Chapter 5, Configuration and Programming, for more information.

IMPORTANT The SLC 500 processor (SLC 5/02 or later) supports multiple backup scanners in its local I/O chassis. The maximum number is dependent on the following:

- backplane power requirements (power supply dependent)
- SLC 500 processor I/O data table limit (4,096 I/O)

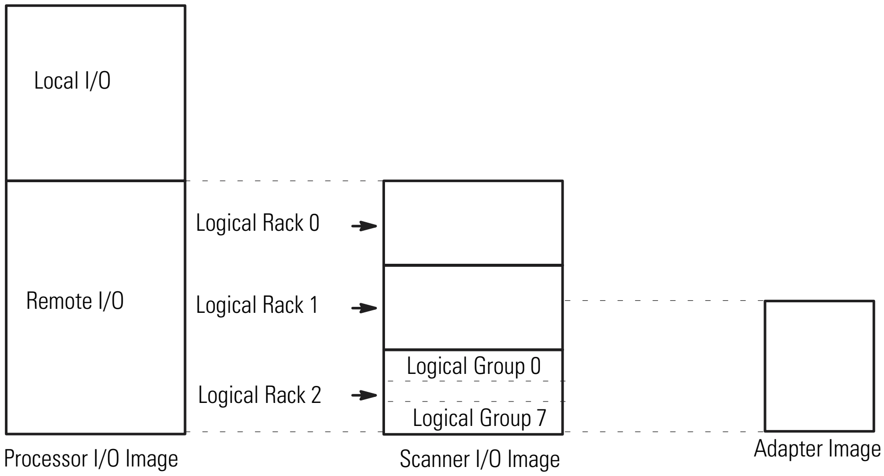
  Based on SLC processor I/O capacity only, a maximum of eight scanners may be used when no local I/O exists.

- processor memory to support the application (SLC 500 processor dependent)

Backup Scanner I/O Image Division

The backup scanner allows each adapter to use a fixed amount (user-defined) of the scanner's input and output image. Part of the SLC processor's image is used by local I/O; the other portion is used by the scanner for remote I/O.

The scanner's remote I/O image is divided into logical racks and further divided into logical groups. A full logical rack consists of eight input and eight output image words. A logical group consists of one input and one output word in a logical rack. Each logical group is assigned a number from 0 to 7.



The scanner image contains the image of each adapter on the RIO link. The adapter is assigned a portion of the scanner image, which is referred to as the adapter image.

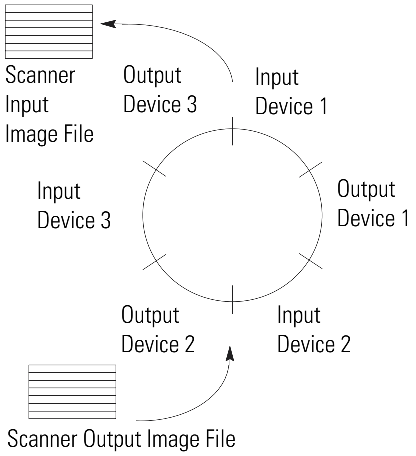
How the Backup Scanner Scans Remote I/O

The scanner communicates with each logical device in a sequential fashion. First, the scanner initiates communication with a device by sending output data to the device. The device responds by sending its input data back to the scanner, as illustrated below. You refer to this exchange as a discrete I/O transfer. After the scanner completes its discrete I/O transfer with the last configured network device, it begins another discrete I/O transfer with the first device.

IMPORTANT The scanner transfers RIO data on a logical device basis not on an adapter basis. A logical device is a full logical rack or portion of a logical rack assigned to an adapter.

Backup Scanner Scan

The scanner updates its input image file each time it scans a logical device.

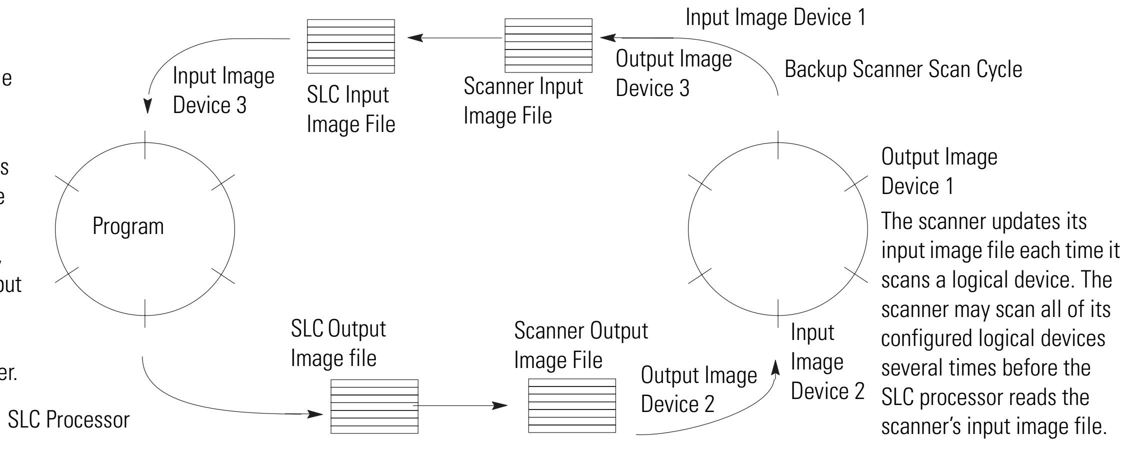


SLC and Scanner Asynchronous Operation

The SLC processor scan and Backup scanner scan are independent (asynchronous) of each other. The SLC processor reads the scanner input image file during its input scan and writes the output image file to the scanner during its output scan. The scanner continues reading inputs and writing outputs to the scanner I/O image file, independent of the SLC processor scan cycle.

Depending on your SLC processor, RIO link configuration, and application program size, the scanner may complete multiple scans before the SLC processor reads the scanner's input image file. The backup scanner updates its I/O files on a per logical rack basis.

The figure below illustrates the asynchronous operation of the SLC processor and backup scanner.



SLC Processor Scan Cycle

The SLC processor reads the scanner input image file into the SLC input image file, processes it, and creates an SLC output image file. The SLC processor transfers its output file to the scanner.

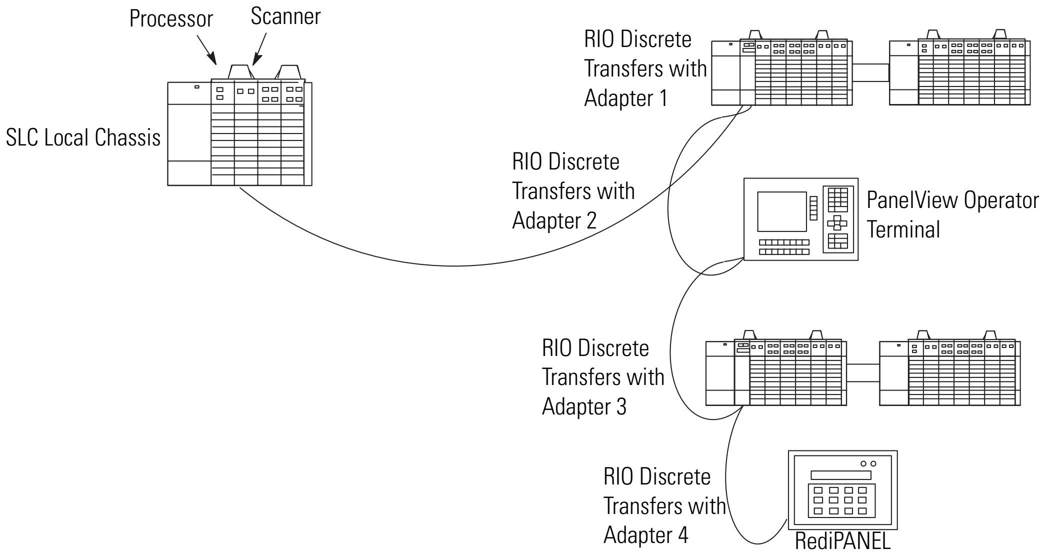
The scanner updates its input image file each time it scans a logical device. The scanner may scan all of its configured logical devices several times before the SLC processor reads the scanner's input image file.

IMPORTANT The outputs of the RIO are updated after the end of the first SLC processor scan.

How the Scanner Interacts with Adapters

The scanner's function is to continuously scan the adapters on the RIO link in a consecutive manner. This scan consists of one or more RIO discrete transfers to each adapter on the RIO link.

RIO discrete transfers consist of the scanner sending output image data and communication commands to the adapter that instruct the adapter on how to control its output. (These include run, adapter reset, and reset decide commands.) The adapter responds by sending input data to the scanner. The scanner performs as many RIO discrete transfers as necessary to update the entire adapter image. If RIO discrete transfers do not occur, data is not exchanged between the scanner and adapter. RIO discrete transfers are asynchronous to the processor scan.



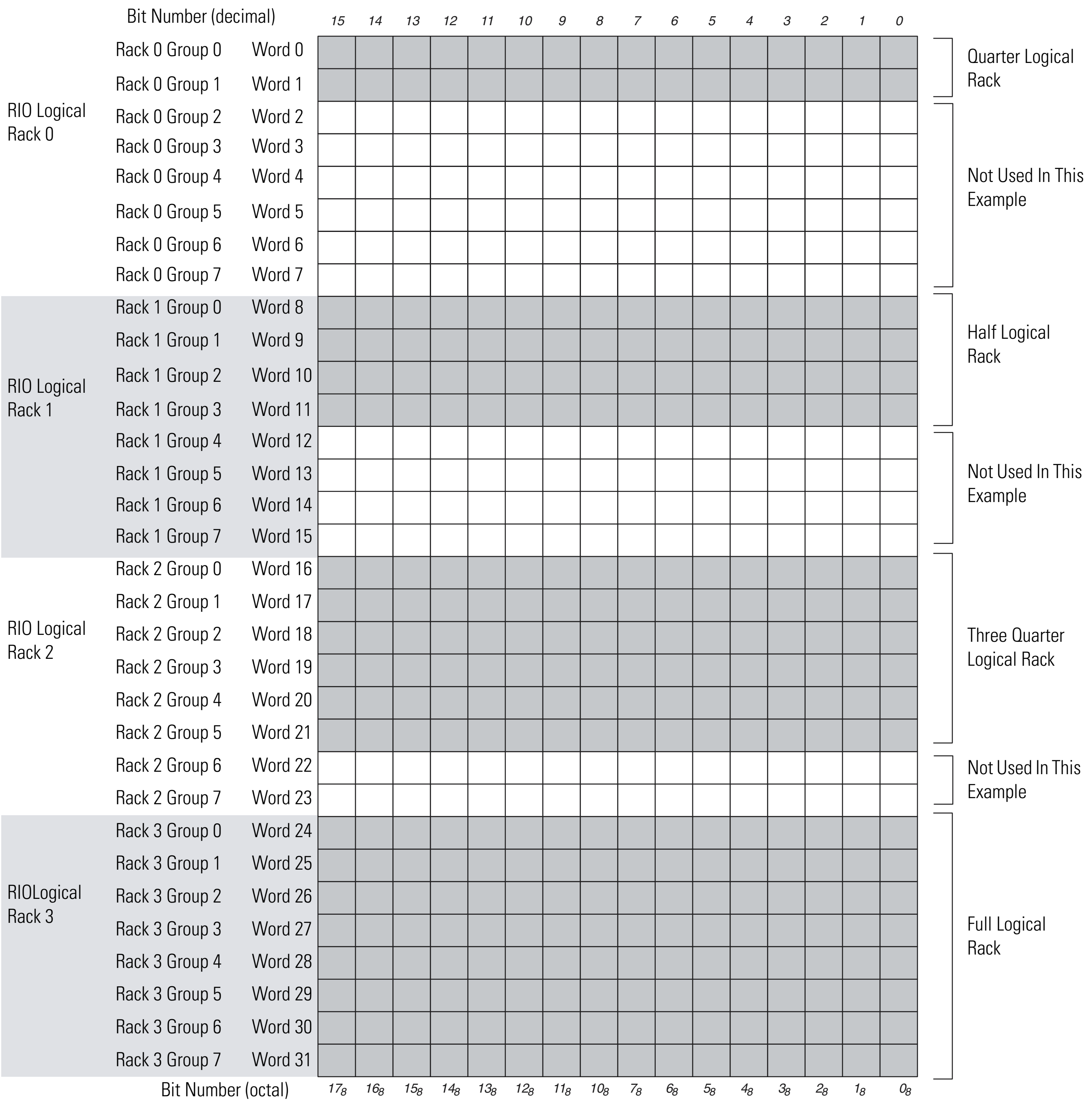
Scanner I/O Image Concepts

The backup scanner's I/O image consists of RIO logical racks and I/O groups. A full RIO logical rack consists of eight input image and eight output image words. (A word consists of 16 bits of data.) Each word within an RIO logical rack is assigned an I/O group number from 0 to 7.

You assign devices on the RIO link a portion of the scanner's image. Devices can occupy a quarter logical rack (2 input and output words), half logical rack (4 I/O words), three quarter logical rack (6 I/O words), or full logical rack. You may configure devices to start at any even I/O group number within an RIO logical rack. More than one physical device's (adapter) I/O information can reside in a single logical rack. Also, by crossing logical rack boundaries a device can consist of more than one logical rack.

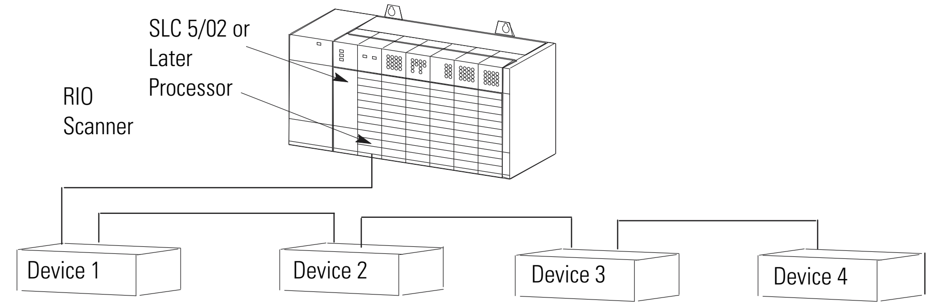
IMPORTANT The illustration below shows only the input image configuration of the scanner's I/O image. The output image configuration is the same.

Input Image Half of a Scanner's I/O Image



Example Scanner I/O Image

The illustration below shows a scanner's input image of 4 RIO link devices.



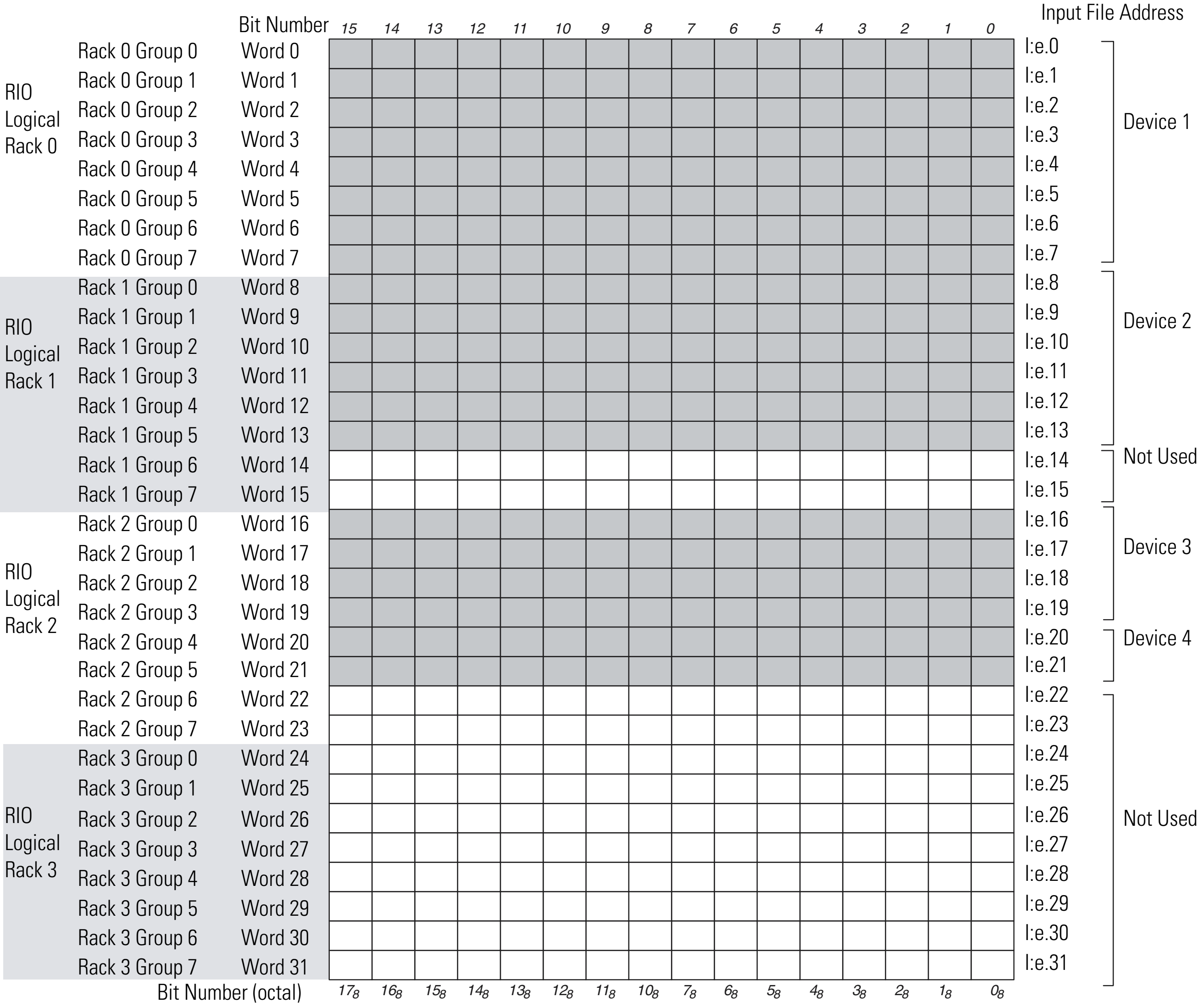
Full Logical Rack Device Begins at Logical Rack 0, Group 0

Three-Quarter Logical Rack Device Begins at Logical Rack 1, Group 0

Half Logical Rack Device Begins at Logical Rack 2, Group 0

Quarter Logical Rack Device Begins at Logical Rack 2, Group 4

IMPORTANT The illustration on the following page shows how the 1747-BSN scanner's input image translates to the SLC 500 processor image. The output image looks the same.



		Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Input File Address	
RIO Logical Rack 0	Rack 0 Group 0	Word 0	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.0	Device 1
	Rack 0 Group 1	Word 1	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.1	
	Rack 0 Group 2	Word 2	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.2	
	Rack 0 Group 3	Word 3	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.3	
	Rack 0 Group 4	Word 4	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.4	
	Rack 0 Group 5	Word 5	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.5	
	Rack 0 Group 6	Word 6	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.6	
	Rack 0 Group 7	Word 7	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.7	
RIO Logical Rack 1	Rack 1 Group 0	Word 8	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.8	Device 2
	Rack 1 Group 1	Word 9	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.9	
	Rack 1 Group 2	Word 10	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.10	
	Rack 1 Group 3	Word 11	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.11	
	Rack 1 Group 4	Word 12	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.12	
	Rack 1 Group 5	Word 13	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.13	
	Rack 1 Group 6	Word 14																	I:e.14	Not Used
	Rack 1 Group 7	Word 15																	I:e.15	
RIO Logical Rack 2	Rack 2 Group 0	Word 16	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.16	Device 3
	Rack 2 Group 1	Word 17	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.17	
	Rack 2 Group 2	Word 18	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.18	
	Rack 2 Group 3	Word 19	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.19	
	Rack 2 Group 4	Word 20	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.20	Device 4
	Rack 2 Group 5	Word 21	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.21	
	Rack 2 Group 6	Word 22																	I:e.22	Not Used
	Rack 2 Group 7	Word 23																	I:e.23	
RIO Logical Rack 3	Rack 3 Group 0	Word 24																	I:e.24	
	Rack 3 Group 1	Word 25																	I:e.25	
	Rack 3 Group 2	Word 26																	I:e.26	
	Rack 3 Group 3	Word 27																	I:e.27	
	Rack 3 Group 4	Word 28																	I:e.28	
	Rack 3 Group 5	Word 29																	I:e.29	
	Rack 3 Group 6	Word 30																	I:e.30	
	Rack 3 Group 7	Word 31																	I:e.31	
		Bit Number (octal)	17_8	16_8	15_8	14_8	13_8	12_8	11_8	10_8	7_8	6_8	5_8	4_8	3_8	2_8	1_8	0_8		

e = slot number of the SLC chassis containing the 1747-BSN scanner

Transferring Data with RIO Discrete and Block Transfers

Input and output image data and command information are quickly exchanged between a scanner and adapter using RIO discrete transfers. RIO discrete transfers are the simplest and fastest way a scanner and adapter communicate with each other. RIO discrete transfers, which are transparent to the user, consist of the scanner sending the output image data to the adapter, and the adapter transmitting input data to the scanner. Each RIO discrete transfer also contains scanner commands for the adapter.

Through your control program you command the SLC processor to initiate RIO block transfers, which directs the scanner to exchange large amounts of data to/from an adapter. Block Transfers (BTs) use the basic RIO discrete transfer mechanism of the RIO link. However, the actual transfer of data occurs asynchronous to the discrete transfers. It is possible for several discrete transfers to occur before the scanner processes a block transfer. Refer to Chapter 7, RIO Block Transfer for more details.

Physical and Logical RIO Link Specifications

The maximum number of adapters with which your scanner can communicate is determined by the scanner's and adapter's physical and logical specifications, as described below:

- *Physical Specifications* are the maximum number of adapters that can be connected to the scanner. For more information, see Extended Node Capability below.
- *Logical Specifications* for the scanner are the maximum number of logical racks the scanner can address, how the logical racks can be assigned, and whether the scanner can perform block transfers.

Extended Node Capability

Extended node functionality allows you to connect up to 32 physical devices on an RIO link. You must use 82 Ohm termination resistors in an extended node configuration. You can only use extended node if *all* RIO link devices have extended node capability. (Refer to the Compatible Devices table at the end of this chapter, or to the specifications of your device.) The 1747-BSN scanner has extended node capability. However, the smallest logical rack division is 1/4 logical rack and the scanner image size is 4 logical racks. Therefore, the scanner is limited to 16 devices unless complementary I/O is used. Refer to the following section for more information on complementary I/O.

Complementary I/O

Complementary I/O is very useful when portions of your input and output images are unused because it allows the images of two adapters to overlap each other in the scanner's I/O image. To use complementary I/O, the I/O image from one adapter must be the mirror (complement) of the other. This means that if there is an input module in the primary chassis, there must be an output module in the same slot of the complementary chassis. This enables total use of the scanner's 32 input and 32 output word image for I/O addressing of up to 1024 discrete points.

ATTENTION

Because the primary and complementary chassis images overlap, input and specialty combination I/O modules must never share the same image location. Inputs received by the scanner may be incorrect and RIO block transfers will not be serviced properly.

If an output module shares its output image with another output module, both output modules receive the same output information.

If you want to use complementary I/O, two adapters that support this function are required (e.g., 1747-ASB modules). One adapter is configured (via its DIP switches) as a primary chassis, the other as a complementary chassis. If a primary chassis exists, it is scanned first.

Primary and complementary chassis cannot have the same logical rack number. The logical rack numbers must be assigned to the primary and complementary racks as shown below:

Primary Chassis	Complementary Chassis Logical Rack Number	
Logical Rack Number	Decimal	Octal
0	8	10_8
1	9	11_8
2	10	12_8
3	11	13_8

ATTENTION

If the logical rack numbers are not properly assigned, unpredictable operation of both ASB modules results. No ASB module errors occur. Refer to your ASB module user manual for specific information on setting the address of the complementary chassis. (For example, in the 1771-ASB manual the addresses for the complementary chassis are referred to as complementary chassis 0-3.)

Guidelines for Configuring Complementary I/O

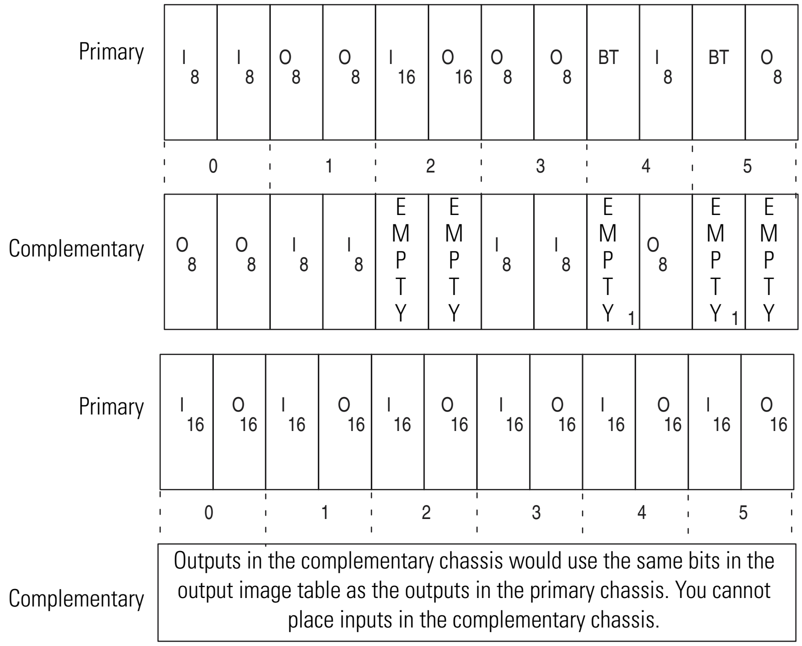
When you configure your remote system for complementary I/O, follow these guidelines:

- You can place an output module in the primary chassis opposite another output module in the complementary chassis; they use the same bits in the output image table. However, we *do not* recommend this placement of modules for redundant I/O.
- You cannot use complementary I/O with a chassis that uses 32-point I/O modules and 1-slot addressing or 16-point I/O modules with 2-slot addressing.
- Do not place an input module in the primary chassis opposite an input module in the complementary chassis because they use the same bits in the input image table.

2-Slot Addressing

The figures below illustrate a possible module placement to configure complementary I/O using 2-slot addressing.

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



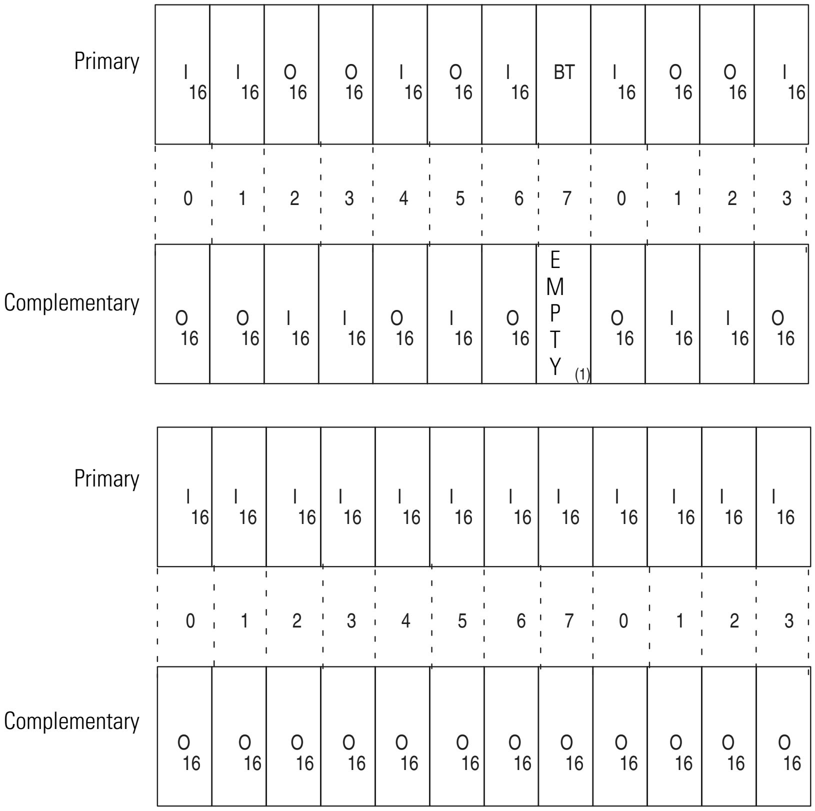
Outputs in the complementary chassis would use the same bits in the output image table as the outputs in the primary chassis. You cannot place inputs in the complementary chassis.

(1) Must be empty if corresponding primary slot is a block transfer module.

IMPORTANT With 2-slot addressing, if an input module resides in either slot associated with a logical group of the primary chassis, an input module cannot reside in that logical group's complementary chassis.

1-Slot Addressing

The figure below illustrates a possible module placement to configure complementary I/O using 1-slot addressing.

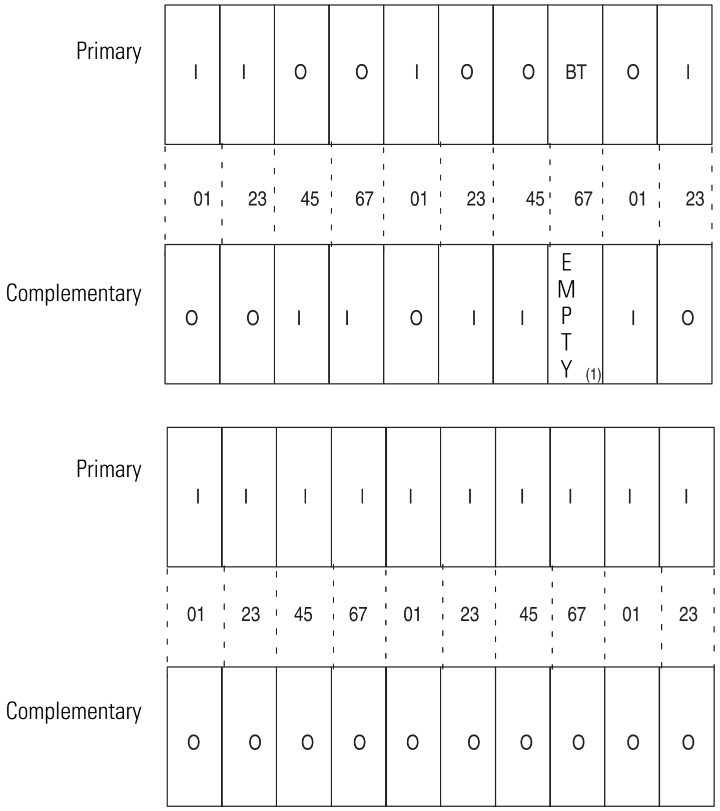


I = Input Module (8- or 16-point)
O = Output Module (8- or 16-point)
BT = Block Transfer Module

(1) Must be empty if corresponding primary slot is a block transfer module.

1/2-Slot Addressing

The figure below illustrates a possible module placement to configure complementary I/O using 1/2-slot addressing.



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

(1) Must be empty if corresponding primary slot is a block transfer module.

Summary for Placing Modules Used In Complementary I/O

Discrete Modules

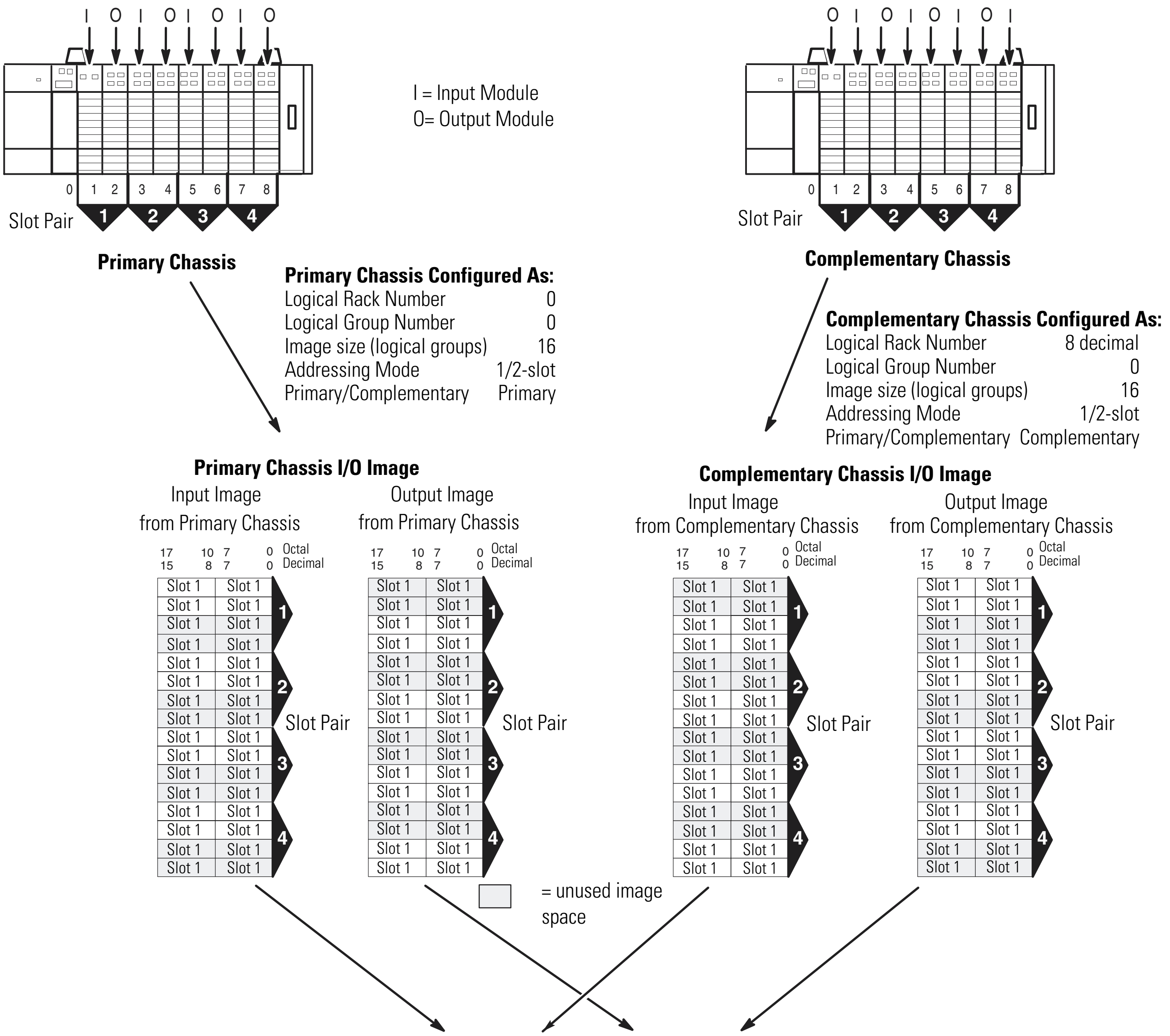
Addressing Method	Types of Modules used	Placement
2-slot	8-point	Install input modules opposite output modules, and output modules opposite input modules.[1]
1-slot	8-point, 16-point	Install input modules opposite output modules, and output modules opposite input modules.
1/2-slot	8-point, 16-point, 32-point	Install input modules opposite output modules, and output modules opposite input modules.

(1) If an input module resides in either slot, associated with a logical group, of the primary chassis, an input module cannot reside in that logical group's complimentary chassis.

Block Transfer Modules

Addressing Method	Placement
2-slot	The right slot of the primary I/O group can be another 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 block transfer module.
1-slot	Leave the corresponding I/O group in the complementary chassis empty.
1/2-slot	Leave the corresponding I/O group in the complementary chassis empty.

The following example illustrates how I/O modules requiring two words of the input or output image can leave unused image space.



I = Input Module
O= Output Module

Primary Chassis Configured As:

Logical Rack Number	0
Logical Group Number	0
Image size (logical groups)	16
Addressing Mode	1/2-slot
Primary/Complementary	Primary

Complementary Chassis Configured As:

Logical Rack Number	8 decimal
Logical Group Number	0
Image size (logical groups)	16
Addressing Mode	1/2-slot
Primary/Complementary	Complementary

Primary Chassis I/O Image / **Complementary Chassis I/O Image**

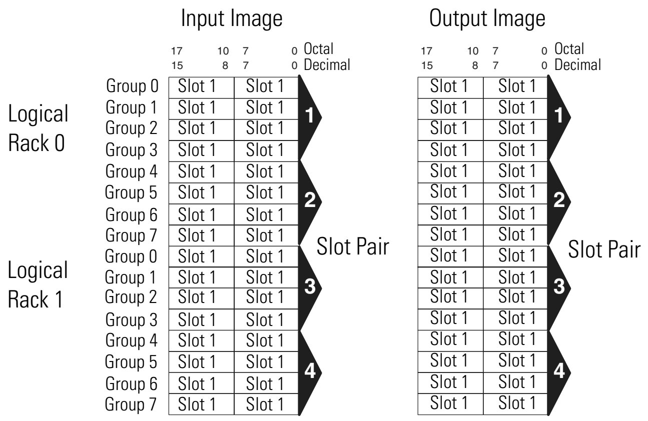
Slot Pair	Input Image from Primary Chassis 17 15 10 8	7 7 0 0 Octal Decimal	Output Image from Primary Chassis 17 15 10 8	7 7 0 0 Octal Decimal	Input Image from Complementary Chassis 17 15 10 8	7 7 0 0 Octal Decimal	Output Image from Complementary Chassis 17 15 10 8	7 7 0 0 Octal Decimal
1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1
	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1
	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1
	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1
2	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1
	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1
	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1
	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1
3	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1
	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1
	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1
	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1
4	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1
	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1
	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1
	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1	Slot 1

▨ = unused image space

Scanner's I/O Image

Both images are overlapped in the scanner. The overlapped image appears where the primary chassis image is configured to reside.

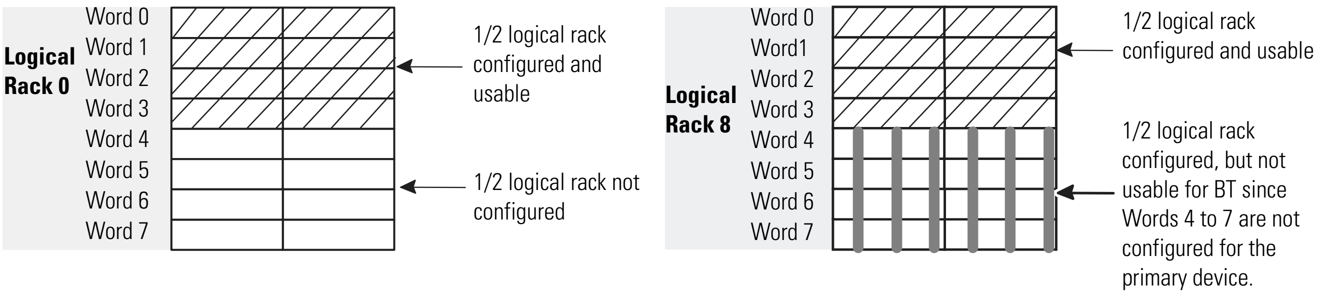
In this case, the primary chassis image is configured as starting logical rack 0 and starting logical group 0.



		Input Image 17 15 10 8	7 7 0 0 Octal Decimal	Output Image 17 15 10 8	7 7 0 0 Octal Decimal	Slot Pair
Logical Rack 0	Group 0	Slot 1	Slot 1	Slot 1	Slot 1	1
	Group 1	Slot 1	Slot 1	Slot 1	Slot 1	
	Group 2	Slot 1	Slot 1	Slot 1	Slot 1	
	Group 3	Slot 1	Slot 1	Slot 1	Slot 1	
	Group 4	Slot 1	Slot 1	Slot 1	Slot 1	2
	Group 5	Slot 1	Slot 1	Slot 1	Slot 1	
	Group 6	Slot 1	Slot 1	Slot 1	Slot 1	
	Group 7	Slot 1	Slot 1	Slot 1	Slot 1	
Logical Rack 1	Group 0	Slot 1	Slot 1	Slot 1	Slot 1	3
	Group 1	Slot 1	Slot 1	Slot 1	Slot 1	
	Group 2	Slot 1	Slot 1	Slot 1	Slot 1	
	Group 3	Slot 1	Slot 1	Slot 1	Slot 1	
	Group 4	Slot 1	Slot 1	Slot 1	Slot 1	4
	Group 5	Slot 1	Slot 1	Slot 1	Slot 1	
	Group 6	Slot 1	Slot 1	Slot 1	Slot 1	
	Group 7	Slot 1	Slot 1	Slot 1	Slot 1	

Application Considerations

If you configure a complementary device to use more I/O image space than an associated primary device, then block transfers can only be performed to locations in the complementary device that have associated I/O image space in the primary device. For example, if a primary device is 1/2 logical rack and a complementary device is a full logical rack, block transfers can be performed only in the first 1/2 logical rack of the complementary device. Attempting block transfers in the last half of the complementary device results in a BT error (error 11 - device not configured).



Complementary 1771 I/O Module Details

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

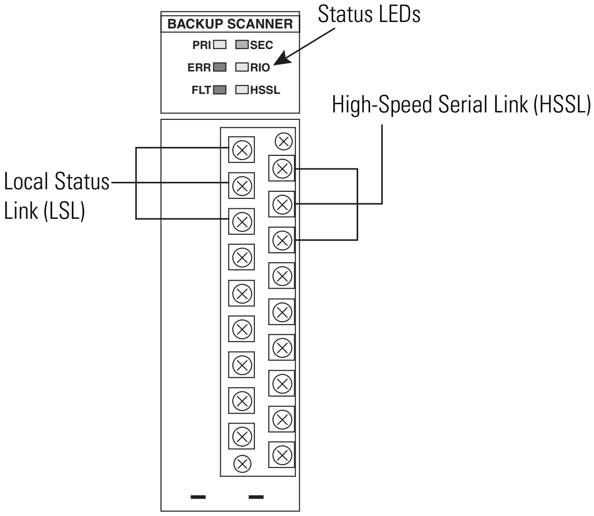
- Communication Adapter Module (1771-KA2)
- Communication Controller Module (1771-KE)
- PLC-2 Family/RS-232-C Interface Module (1771-KG)
- Fiber Optics Converter Module (1771-AF)
- DH/DH+ Communication Adapter Module (1785-KA)
- DH+/RS-232C Communications Interface Module (1785-KE)

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:

- 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)

Hardware Features

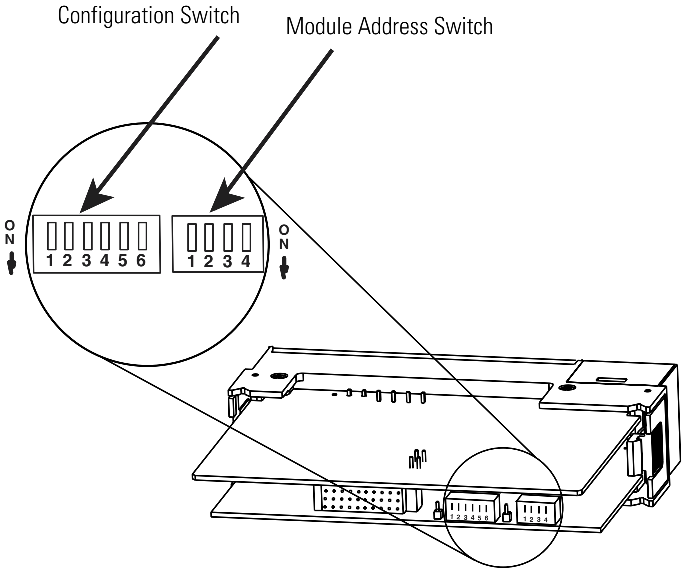
Note the backup scanner's hardware features in the following illustration.



Status LEDs

The table below describes the six LEDs located on the module's front panel. To ensure that they are operating correctly, all LEDs illuminate during power up.

LED	Definition	Status & Color	Indication
PRI	Primary	Steady Green	The module is in the primary mode.
SEC	Secondary	Steady Amber	The module is in the secondary mode.
RIO	RIO Communication	Steady Green	The RIO link is working properly.
		Flashing Green	A remote device is not configured or connected correctly, or is faulted.
		Flashing Red	The RIO link has a fault. The scanner is connected incorrectly, or all devices are configured improperly, have no power, or are faulted.
		Steady Red	There is a configuration error.
		Off	The communication channel is not configured as RIO.
ERR	Backup Module Error	Flashing Red	The module is not ready for switchover.
		Off	The module is ready for switchover.
HSSL	High-Speed Serial Link Communication	Flashing Green	The link is operating with no errors.
		Off	A communications error has been detected on the HSSL.
FLT	Fault	Steady Red	A hardware fault has occurred.
		Flashing Red	The module is not configured properly.



Configuration Dip Switch Settings

The six-position Configuration DIP Switch is used to select the baud rate, configure the communication channel and identify each individual BSN module and the last BSN module. The tables below define the DIP switch configuration settings.

DIP Switch Position	Definition	Setting
1 and 2	Set the communication channel baud rate	See the table on page 1-20
3	Channel configuration	DH+ = ON RIO = OFF
4	User Identification Switch. This switch can be used to differentiate one 1747-BSN module from its counterpart and help identify when a switchover occurs. This switch is user-configurable and does not affect the operation of the module.	User selectable
5	Reserved	
6	Identifies the last module in the local status link, if multiple BSN modules are used in each chassis. If only one module is used in each chassis, turn this switch to ON.	Last module = ON All others = OFF

Baud Rate Settings

Position 1	Position 2	Baud Rate
ON	ON	57.6K
ON	OFF	115.2K
OFF	ON	230.4K
OFF	OFF	Disabled

Module Address Switch

The four-position Module Address DIP switch configures the BSN address in the LSL. The following table shows the address that corresponds to each setting.

Switch Position				1747-BSN Address
1	2	3	4	
OFF	OFF	OFF	Not Used	1
ON	OFF	OFF		2
OFF	ON	OFF		3
ON	ON	OFF		4
OFF	OFF	ON		5
ON	OFF	ON		6
OFF	ON	ON		7
ON	ON	ON		8

IMPORTANT The module address must equal the actual slot number or the module will fault. For example, the 1747-BSN module in slot 1 must have an address equal to 1.

Compatible Devices

The 1747-BSN is compatible with all Remote I/O adapter devices.

Backup Concepts for the SLC 500 System

Why Use a Backup System?

The objective of any redundant system (backup system) is to improve the amount of up-time of a machine or process by ensuring consistent availability of that machine, and by reducing costs associated with equipment failure. By using this backup system, you can guard your application against shutdowns caused by the programmable controller.

ATTENTION Backup does not protect you from faults caused by programming errors or system timeouts because such an error or timeout also occurs in the secondary processor.

The backup option is used when you must transfer the control of the process to a secondary system, without interrupting the machine/process operation.

To guard against system shutdown, a backup system must provide:

- equipment with exceptional reliability
- automatic fault isolation
- minimal disturbance of the process when switching from the primary to the secondary system

Applying 1747-BSN Backup Scanner Modules to the SLC-500 Programmable Controller

An SLC 500 system configured with 1747-BSN modules provides high-speed backup communication and switchover of the Data Highway Plus, remote I/O links, and RS232 channel (on 5/03 and later processors). In this section, we

- show a typical SLC 500 backup configuration,
- explain how the backup system works, and
- describe the role of the 1747-BSN module

A Typical SLC 500 Backup Configuration

An SLC 500 backup system contains a minimum of two each of the following hardware components:

- SLC 500 processor module

Processor	Catalog Number
SLC 5/02	1747-L524
SLC 5/03	1747-L531, -L532
SLC 5/04	1747-L541, -L542, -L543
SLC 5/05	1747-L551, -L552, -L553

- 1747-BSN module
- power supply
- local chassis

How the SLC 500 Backup System Works

In the SLC 500 backup configuration, one system (consisting of one SLC 500 processor, 1747-BSN module, power supply, and chassis) controls the operation of the remote I/O. This system is referred to as the primary system. The other system is ready to take control of the remote I/O in the event of a fault in the primary system. This is referred to as the secondary system. The SLC 500 backup system does not back up local I/O; therefore, do not install I/O in the local chassis.

The system provides high-speed data table transfer from the primary system to the secondary system. This guarantees that the data tables of the two processors track each other. To provide the possibility of backing up a large number of I/O points, a backup system supports up to eight 1747-BSN modules in each processor chassis linked by a local status link (LSL).

The **high-speed serial link** (HSSL) has the following functionality:

- Sends the network commands from the primary to the secondary 1747-BSN and the replies from the secondary to the primary when the communication channel is configured as DH+.

- Transfers a limited number of SLC 5/0x data table files from primary to the secondary processor. This data table transfer is done through an application program in the SLC 5/0x processor.

Although data transfer between the primary and secondary processors is done as fast as possible, there is no guarantee that a switchover from the primary to the secondary will be bumpless. The data rate is 2 Mbits/second.

The **local status link** (LSL) is a 57.6 KBaud serial link provided for exchanging status between the 1747-BSNs that are in the same chassis.

Data Transfer

During normal operation, the primary system sends remote input and retentative data table data to the secondary system so that in the event of a switchover, the secondary system (which becomes the new primary system) has the same data.

Remote I/O data is automatically transferred over the High-Speed Serial Link. This transfer is independent of the application program.

Each 1747-BSN is capable of transferring up to 2 KWords of the SLC 500 data table. This capacity is increased as the number of 1747-BSN modules per local chassis is increased in a redundant system. For example, a system with eight 1747-BSNs in each chassis is capable of transferring up to 16 KWord of synchronizing data table information.

Data table values are transferred from the primary to the secondary system with M0 and M1 files and the HSSL. You copy the data to and from M files via your ladder program.

You do not have to transfer data table values if it is not necessary for your application.

For detailed information about data transfer from the primary to the secondary system, refer to Chapter 7.

Switchover

Should a fault occur in the primary processor, control switches to the secondary system in less than 50 ms (maximum). When a switchover occurs, the outputs in the remote I/O maintain their last state until they come under the control of the secondary processor.

The program scans of the two processors are not synchronized. This means that the secondary processor may be in a different place in the scan cycle than the primary processor. This manual explains the switchover process and provides guidelines for developing programs for your SLC 500 backup system. (For more information about switchover, refer to Chapter 8, Switchover Considerations.)

Role of the 1747-BSN Module

As an integral part of the backup system, the 1747-BSN modules enable high-speed communication between the two SLC processors and permit the secondary processor to assume control of the process. In addition, the 1747-BSN module provides:

- high-speed transfer of the data table values from the primary to the secondary system (HSSL), to ensure that the secondary system's data table is a copy of the primary system's (2K words maximum per BSN module pair)
- exchange of information on the status of the primary and secondary systems (LSL)
- automatic transfer to the secondary system of the remote input
- transfer of control from the primary processor to the secondary processor when one of the following conditions occur:
  - power failure or power loss
  - major processor fault
  - 1747-BSN module fault
- change in the primary processor's mode from:
  - RUN to PROGRAM (manual switchover)
  - REM RUN to REM PROG
  - REM RUN to REM TEST
- transfer of control from the primary processor to the secondary when one of the following conditions is detected by the secondary processor:
  - communication timeout in the HSSL between the two 1747-BSN modules and primary system is not updating the remote I/O
  - transfer of control command from the primary 1747-BSN
- substitution of equipment without interruption of the process; that is, the faulted system can be repaired while the other system is controlling the process
- connections for remote I/O and Data Highway Plus network (the 1747-BSN module routes the remote I/O network and the Data Highway Plus network to the primary processor)
- isolation of the systems, in order to guarantee that a fault in one system does not affect the other
- diagnostics information
- remote programming capability for secondary processor. The secondary processor is in the DH+ network
- the capability of switching two communication channels (one configurable as RIO or DH+ and one RS-232/485 for DH485)
- minimal user programming impact
- use of standard SLC 5/02 or later and 1746 platform

Quick Start for Experienced Users

This chapter helps you to get started using the Backup Scanner. We base the procedures here on the assumption that you have a basic understanding of SLC 500 products. You must:

- understand electronic process control
- be able to interpret the ladder logic instructions for generating the electronic signals that control your application

Because it is a start-up guide for experienced users, this chapter *does not* contain detailed explanations about the procedures listed. It does, however, reference other chapters in this book where you can get more detailed information. It also references other documentation that may be helpful if you are unfamiliar with programming techniques or system installation requirements.

If you have any questions, or are unfamiliar with the terms used or concepts presented in the procedural steps, *always read the referenced chapters* and other recommended documentation before trying to apply the information.

This chapter:

- tells you what tools and equipment you need
- lists preliminary considerations
- describes when to address, configure and program the module
- explains how to install and wire the module
- discusses system power-up procedures

Required Tools and Equipment

Have the following tools and equipment ready:

- medium blade screwdriver
- programming equipment
- termination kit (package of resistors and ring lug included with the scanner)
- an adequate length of RIO communication cable (Belden 9463) for your specific application

Procedures

1. Ensure your chassis supports placement of the 1747-BSN module.

   Review the power requirements of your system to see that your chassis supports placement of the scanner module. The scanner consumes 800 mA at 5V dc.

2. Configure the module using the DIP switches. (Refer to Chapter 3, Installation and Wiring, to configure the switches.)

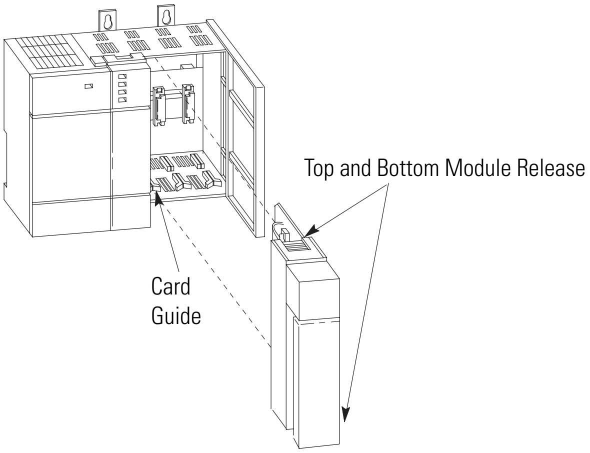
   Set the DIP switches located on the printed circuit board. The six-position Configuration DIP Switch is used to select the baud rate, configure the communication channel, and identify each BSN module and the last BSN module.

IMPORTANT All RIO/DH+ devices must be configured for the same baud rate.

3. Make sure system power is off; then insert the 1747-BSN scanner module into your 1746 chassis.

ATTENTION Never install, remove, or wire modules with power applied to the chassis or devices wired to the module.

In this example procedure, local slot 1 is selected.



4. Connect all RIO/DH+ link devices. Ensure that you:
   - Daisy chain each RIO/DH+ link device.
   - Connect the appropriate termination resistors on each end of the link.

5. Configure the system.

   Set up your system I/O configuration for the particular slot in which you installed the scanner (slot 1 in this example). If the 1747-BSN is not yet listed in your version of programming software, select “Other” and type in a module ID code of 13609.

6. Enter the number of scanned words.

   Enter the number of Scanned Input and Output Words using the Specialty I/O and Advanced Setup menus. The default value is 32 I/O words. You can specify less than 32 and reduce the processor scan time by transferring only the part of the input and output image that your application requires. If you have configured the BSN for DH+ mode, you may configure the scanned I/O words for 1. It is important that you do not set either of these values to 0. If you do, the BSN will not work correctly.

7. Set the M0-M1 and G file sizes.

   Using the Specialty I/O Configuration menu, set the M1 and M0 file sizes to at least 32 words (48 words if using complementary I/O). 32 words is the minimum required for operation. If you do not set the M1 and M0 file sizes to at least 32 words, the programming device does not allow you to access the M files in the SLC control program. The maximum M file sizes are 5548 words for complete capability. There is no penalty to set the M file lengths to this maximum.

   Set the G file size to 3 (5 if using complementary I/O) using the Specialty I/O Configuration menu.

   Write your ladder code to address discrete I/O from the 32 input and output image words and the M files for control and status of the RIO network as well as block transfer and the HSSL, if needed. Refer to Appendix B, M0-M1 Files and G Files before completing this selection.

8. Go through the system start-up procedure.

   a. Apply power.
   b. Download your program to the primary and secondary SLC.
   c. Place the primary SLC in Run mode.
   d. Place the secondary SLC in Run mode.

The backup scanner's FAULT and ERROR LEDs are off, the RIO LED is green, the HSSL LED is blinking and the two PRI and SEC shows each processor Mode at this time. (This is the valid LED pattern when in Run mode or after a Run mode to Program mode transition.)

Installation and Wiring

This chapter contains the information necessary to:

- select the baud rate
- configure the Backup Scanner
- insert the Backup Scanner into the SLC chassis
- wire the RIO and communication links

Compliance to European Union Directives

If this product has the CE mark it is approved for installation within the European Union and EEA regions. It has been designed and tested to meet the following directives.

EMC Directive

This product is tested to meet Council Directive 89/336/EEC Electromagnetic Compatibility (EMC) and the following standards, in whole or in part, documented in a technical construction file:

- EN 50081-2
  EMC - Generic Emission Standard, Part 2 - Industrial Environment
- EN 50082-2
  EMC - Generic Immunity Standard, Part 2 - Industrial Environment

This product is intended for use in an industrial environment.

Low Voltage Directive

This product is tested to meet Council Directive 73/23/EEC Low Voltage, by applying the safety requirements of EN 61131-2 Programmable Controllers, Part 2 – Equipment Requirements and Tests.

For specific information required by EN61131-2, see the appropriate sections in this publication, as well as the following Allen-Bradley publications:

- *Industrial Automation, Wiring and Grounding Guidelines for Noise Immunity*, publication 1770-4.1
- Automation Systems Catalog

Safety Considerations

This equipment is UL certified for ordinary locations only. The module is C-UL certified for use in Class I, Division 2, Groups A, B, C, D, or non-hazardous locations only. The following attention statement applies to use in hazardous locations.

ATTENTION

Explosion Hazard

- Substitution of components may impair suitability for Class I, Division 2.

- Do not replace components or disconnect equipment unless power has been switched off, and the area is known to be non-hazardous.

- Do not connect or disconnect connectors or operate switches while circuit is live unless the area is known to be non-hazardous.

- All wiring must comply with N.E.C. article 501-4(b).

TIP

The temperature code rating is marked on the product label.

Configuration Selection

The six-position Configuration DIP Switch is used to select the baud rate, configure the communication channel, and identify each individual BSN module and the last BSN module. The tables below define the DIP switch configuration settings.

DIP Switch Position	Definition	Setting
1 and 2	Set the communication channel baud rate.	See the table on page 2
3	Channel configuration.	DH+ = ON RIO = OFF
4	User Identification Switch. This switch can be used to diferentiate one 1747-BSN from its counterpart and identify when a switchover occurs. This switch is user-configurable and does not affect the operation of the module.	User selectable
5	Reserved.	
6	Identifies the last module in the local status link.	Last module = ON All others = OFF

Baud Rate Settings

Position 1	Position 2	Baud Rate
ON	ON	57.6K
ON	OFF	115.2K
OFF	ON	230.4K
OFF	OFF	Disabled

Module Address Switch

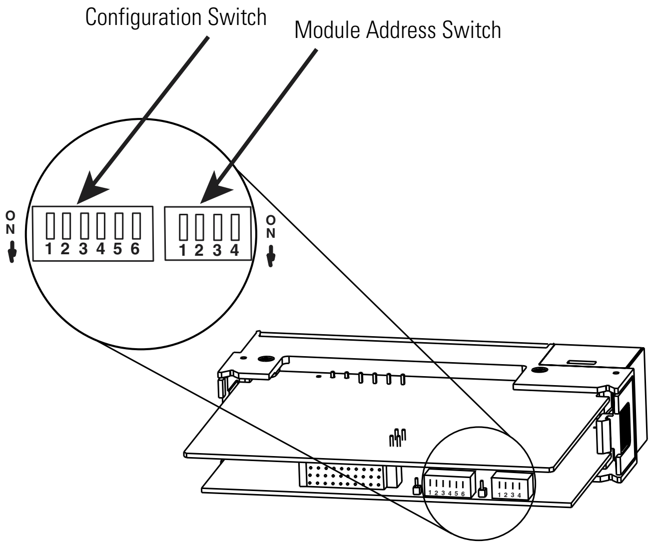
The four-position Module Address DIP switch configures the BSN address in the LSL. The table below shows the address that corresponds to each setting.

Switch Position[1]			1747-BSN Address
1	2	3	
OFF	OFF	OFF	1
ON	OFF	OFF	2
OFF	ON	OFF	3
ON	ON	OFF	4
OFF	OFF	ON	5
ON	OFF	ON	6
OFF	ON	ON	7
ON	ON	ON	8

(1) Switch position 4 is not used.

The figure below shows the location of the DIP switches on the Backup Scanner.

IMPORTANT For proper RIO link system operation, all devices must be configured for the same baud rate.



Backup Scanner Installation

Installation procedures for this module are the same as for any other discrete I/O or specialty module. Refer to the illustration on page 3-5 to identify chassis and module components listed in the procedures below.

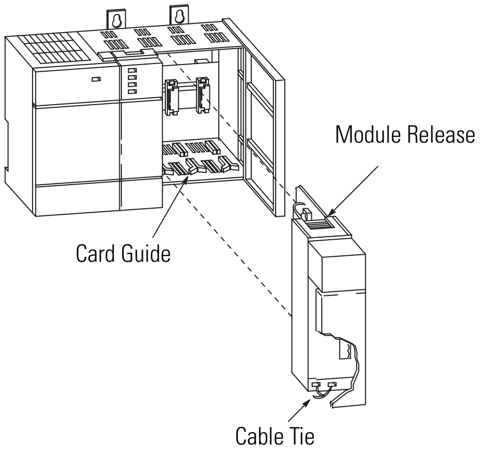
ATTENTION Disconnect system power before attempting to install, remove, or wire the backup scanner.

IMPORTANT Make sure you have set the DIP switches properly before installing the scanner.

IMPORTANT Before installation, ensure that your modular SLC power supply has adequate reserve current capacity. The scanner requires 800 mA at 5V dc.

Insertion

1. Disconnect power.
2. Align the full-sized circuit board with the chassis card guides. The first slot (slot 0) of the first rack is reserved for the SLC 500 processor.
3. Slide the module into the chassis until the top and bottom latches catch.
4. Install terminal wiring. See page 3-6.
5. Insert the cable tie in the slots.
6. Route the cable down and away from module, securing it with the cable tie.
7. Cover all unused slots with the Card Slot Filler, Catalog Number 1746-N2.



Removal

1. Disconnect power.
2. Remove all cabling.
3. Press the releases at the top and bottom of the module and slide the module out of the chassis slot.
4. Cover all unused slots with the Card Slot Filler, Catalog Number 1746-N2.

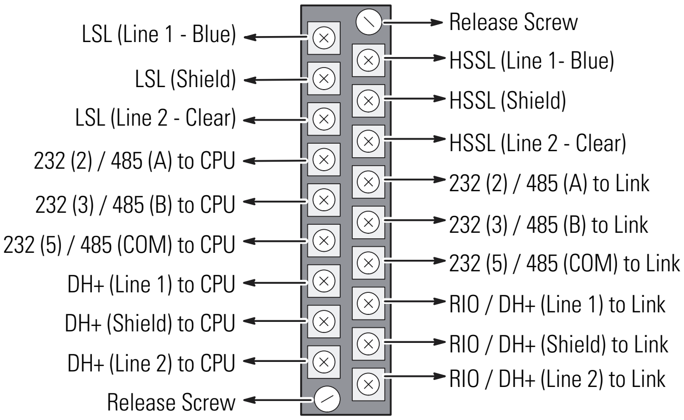
Wiring

Terminal Wiring

The backup scanner module contains a green removable terminal block. The terminal pinout is shown on the following page.

ATTENTION

Disconnect power to the SLC before attempting to install, remove or wire the removable terminal wiring block.



Terminal screws accept a maximum of two #14 AWG (2mm 2) wires. Tighten terminal screws only tight enough to immobilize wires. Maximum torque on terminal screws is 0.9 Nm (8 in-lbs.).

Wiring Considerations

The system examples on pages 3-7 through 3-9 have been simplified to show only the type of wiring described. Keep the following considerations in mind when planning your system:

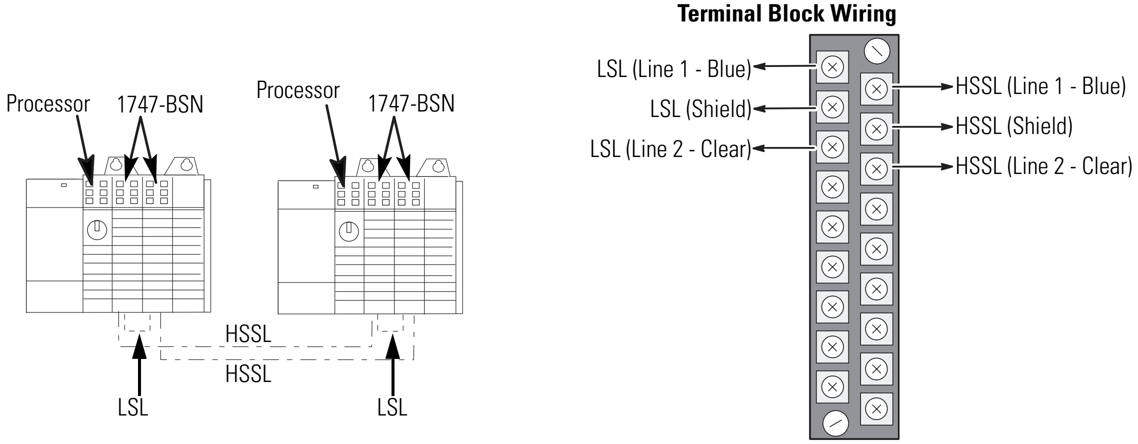
- When wiring a system using 1747-BSN backup scanners, you must connect the High-Speed Serial Link (HSSL) between the primary and secondary backup scanners. The Local Status Link (LSL) is required only when more than one 1747-BSN module per chassis is used.
- RIO/DH+ connections are dependent upon your system setup and are mutually exclusive.
- RS-232 connections are also optional, dependent upon your system setup.

Local Status Link and High-Speed Serial Link Wiring

The High-Speed Serial Link connects complementary 1747-BSN modules. The HSSL allows network commands to be transferred between BSN modules using DH+. Retentive data can be transferred between the primary and secondary processors via the HSSL via ladder logic supporting handshake data transfer. The complementary BSN modules are connected using Belden™ 9463 cable.

The Local Status Link exchanges status information between 1747-BSN modules in the same chassis without programming. The BSN modules are connected using Belden 9463 cable.

Figure 3.1 HSSL and LSL Connections



RIO Network and RIO Link Wiring

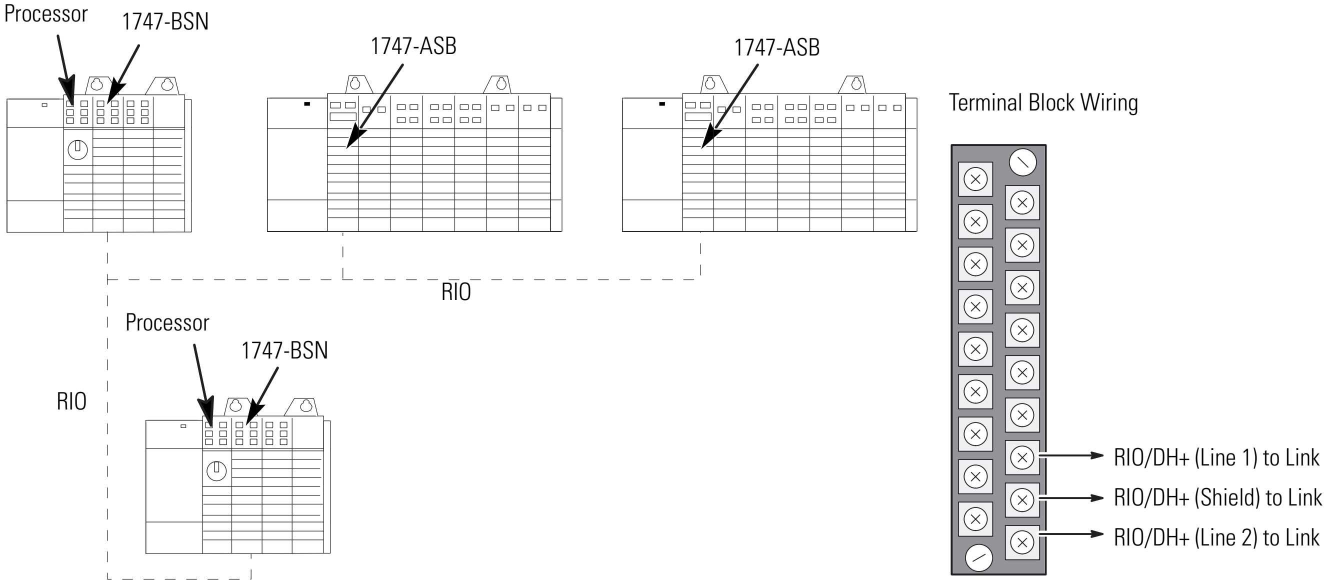
In primary mode, the 1747-BSN module can act as a RIO scanner, supporting discrete I/O control, RIO block transfers, and RIO passthru. In secondary mode, the BSN module monitors the primary BSN. If the primary BSN module fails, the secondary module becomes primary.

The scanner is connected to other devices using the RIO link. There are no restrictions governing the space between each device, provided the maximum cable distance is not exceeded.

A 1/2 Watt terminating resistor (included with the module) must be attached across line 1 and line 2 of the connectors at *each* end (scanner and last physical device) of the RIO link. The value of the resistor depends on the baud rate and extended node capability, as shown in the table on A-2.

IMPORTANT To use extended node, all devices on the RIO link must support it. Refer to each device's user manual.

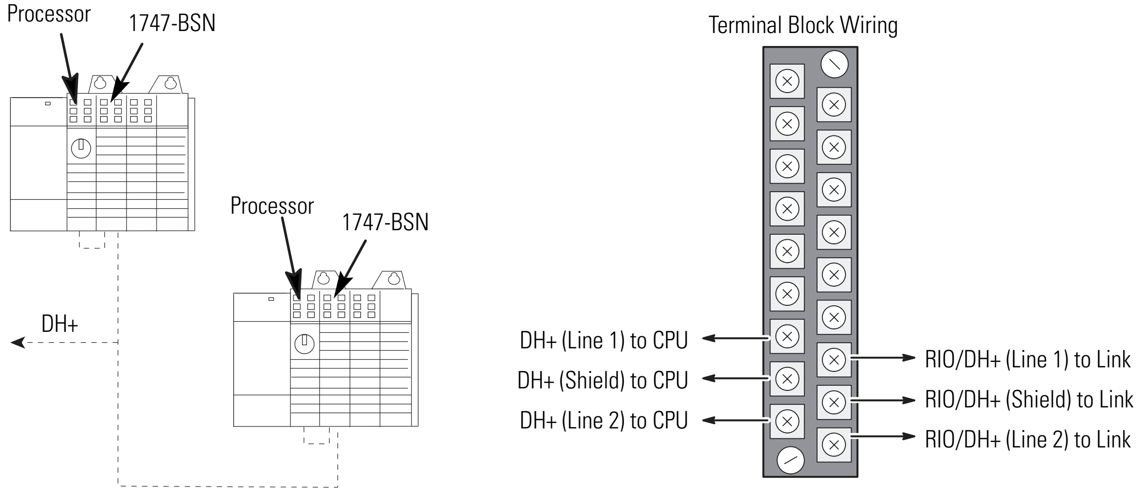
Figure 3.2 RIO System



DH+ Network Wiring

The 1747-BSN module acts as a smart switch, selectively allowing both processors to be addressed on the DH+ network at node addresses *n* (primary) and *n+1* (secondary), where *n* is the DH+ node address used in the control program.

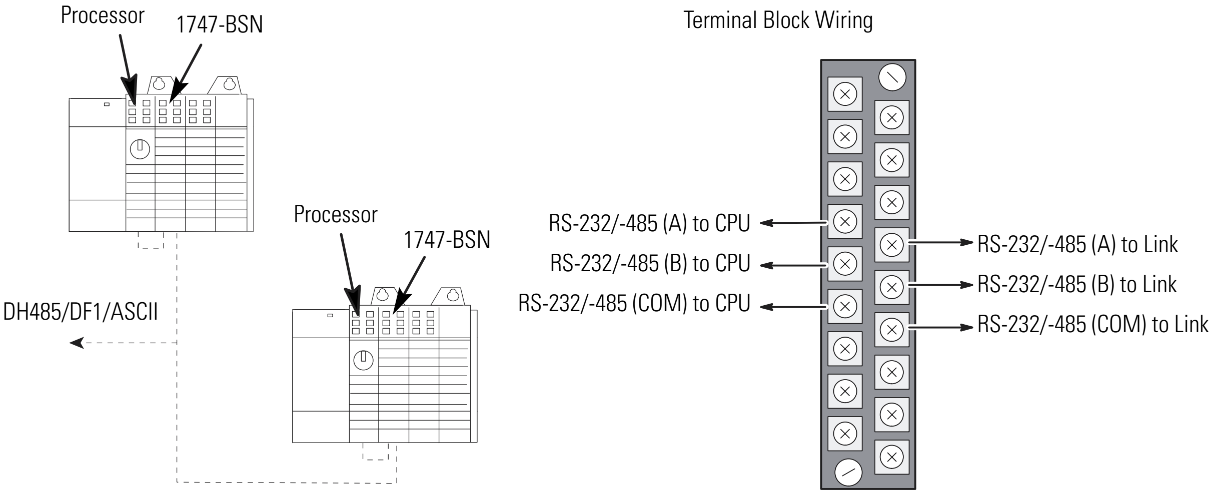
Figure 3.3 DH+ System



RS-232/RS-485 Network Wiring

The 1747-BSN module supports RS-232 or RS-485 communications for the primary processor only. The RS-232 or RS-485 network can use the same BSN module as the DH+ or RIO network.

Figure 3.4 RS-232/485 System



Status LEDs

The table below describes the six LEDs located on the module's front panel. To ensure that they are operating correctly, all LEDs are illuminated during power-up.

LED	Definition	Status & Color	Indication
PRI	Primary	Steady Green	The module is in the primary mode.
SEC	Secondary	Steady Amber	The module is in the secondary mode.
RIO	RIO Communication	Steady Green	The RIO link is working properly.
		Flashing Green	A remote device is not configured or connected correctly, or is faulted.
		Flashing Red	The RIO link has a fault. The scanner is connected incorrectly, or all devices are configured improperly, have no power, or are faulted.
		Steady Red	There is a configuration error.
		Off	The communication channel is not configured as RIO.
ERR	Back-up Module Error	Flashing Red	The module is not ready for switchover.
		Off	The module is ready for switchover.
HSSL	High-Speed Serial Link Communication	Flashing Green	The link is operating with no errors.
		Off	A communications error has been detected on the HSSL.
FLT	Fault	Steady Red	A hardware fault has occurred.
		Flashing Red	The module is not configured properly.

Operating Your SLC 500 Backup System

Chapter Objectives

In this chapter we describe how the primary system transfers data to the secondary system. We do this by describing the operation of the:

- 1747-BSN module
- SLC 500 backup system

In addition, we provide procedures for:

- starting your system
- powering up your system
- disconnecting a faulted system
- restarting a repaired system
- changing the processor's operating mode
- editing a program on-line

How the 1747-BSN Module Operates

In a SLC 500 backup system, the 1747-BSN module performs three distinct functions:

- automatic transfer of remote input data over the High-Speed Serial Link (HSSL)
- transfer of data in the data table by data transfer using M files
- secondary processor remote programming capability

Automatic Transfer of Remote Input Data Over the HSSL

With automatic transfer, the primary 1747-BSN module is continually updating a copy of the remote input image table in an interface located in the secondary 1747-BSN module. This remote input data includes discrete and analog data.(1) Automatic transfer of input image is carried out over the High-Speed Serial Link (HSSL) and is independent of the application program.

---

(1) This does not include Block Transfer analog data.

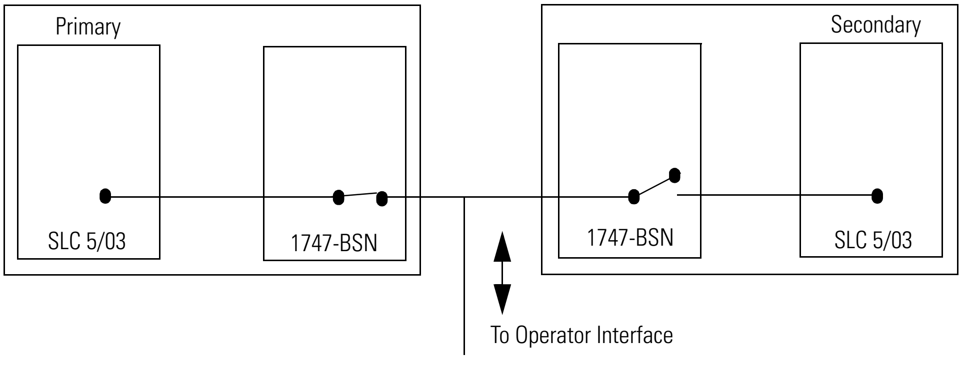
The secondary 1747-BSN module responds with current data to the remote I/O scan requests of the secondary processor, making the secondary processor “think” that it is communicating with remote I/O chassis. This response prevents remote I/O faults in the secondary processor, since the secondary BSN module is not physically connected to the remote I/O link.

In addition to providing the secondary processor a copy of the remote discrete input image table, the 1747-BSN modules in both systems work together to provide the secondary processor a copy of block transfer data read from the remote I/O chassis.

The 1747-BSN automatically exchanges system status information when communicating with another 1747-BSN through the High-Speed Serial Link (HSSL). Both modules are continuously communicating with each other, even when both processors are not in Run mode. This is achieved by sending and receiving the system status words. This information can be read from either module at any time and provides for both SLC 5/0x (primary and secondary) the “other side status”. HSSL is also used to transfer user data table data, up to 2K total per BSN module.

RS-232/485 Communication Channel

The RS-232/485 channel is a dumb communication channel which is used to switch the connection to one operator interface or other serial communications device (through RS-232/485). The communication channel relays are closed when the module is primary; otherwise the relays are open as shown in the figure below.



RIO/DH+ Communication Channel

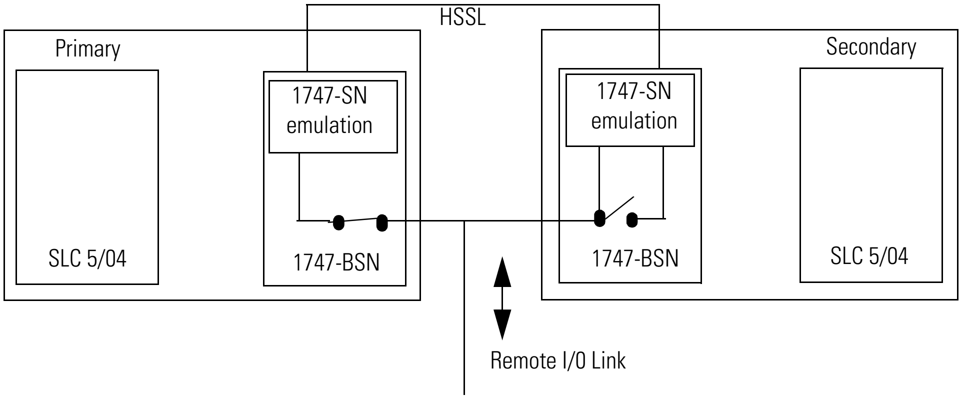
The RIO/DH+ communication channel has a relay that is closed when the 1747-BSN module is in the primary mode; otherwise the relay is open. The communication channel supports the following configurations (The illustration on the next page shows the channel block diagram):

1. RIO: Supports the 57.6, 115.2 and 230.4 KBaud configurations. When the module is in primary mode, it emulates all the 1747-SN Series B functionality. When the module is in secondary mode, the channel receives the input data from the primary 1747-BSN module and sends it to the SLC 5/0x, emulating the 1747-SN Series B backplane communication. As a result, the secondary SLC 5/0x “thinks” that it is really connected to the remote racks.

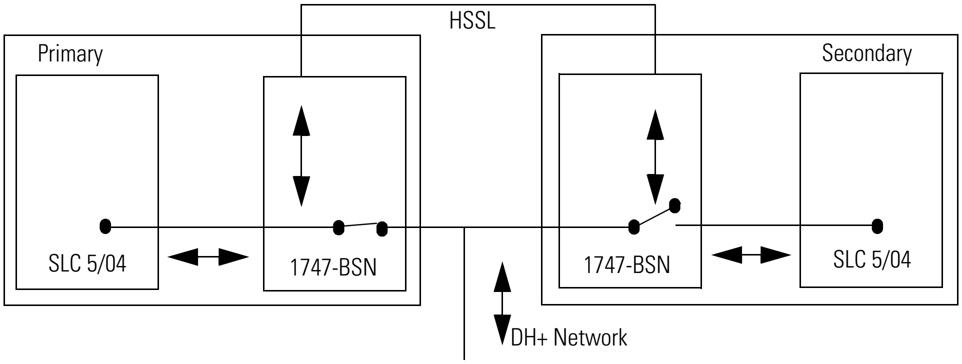
2. DH+: Works under 57.6, 115.2 and 230.4 KBaud configurations. The primary 1747-BSN enters into the DH+ network using the first address that follows the secondary SLC 5/04 network address. The commands that it receives are sent through the HSSL to the secondary 1747-BSN and then to the secondary processor (the secondary 1747-BSN emulates the DH+ network for the secondary SLC 5/04). The replies come back to the primary 1747-BSN (again through the HSSL) and are sent to the DH+ network. This link has a functionality similar to that found in the 1785-BCM series B and C. The link is designed to make remote programming possible for the secondary SLC 5/04. The primary SLC 5/04 is connected to the DH+ network through the primary 1747-BSN communication channel relays and sends/receives messages without any intervention of the 1747-BSN module.

3. Disabled: In this mode the RIO/DH+ communication channel is disabled and is used only as a relay that is closed when the 1747-BSN is in the primary mode and otherwise is open.

RIO



DH+



Secondary Processor Remote Programming

IMPORTANT The programming device must be connected through the DH+ network and must not bypass the relay in the 1747-BSN module (if the programs in both processor are identical; e.g., if both DH+ mode addresses are the same). If you were to connect the device directly to a processor, it must be connected to the primary processor. Therefore, if your system were to switch to the secondary system, you must then move your device to your new primary processor.

In the 1747 backup system, only the primary processor is connected to the link. The secondary processor is not physically connected, but it communicates with the secondary 1747-BSN module making this processor think that it is on the DH+ network. This separate link prevents Data Highway faults in the secondary processor.

The 1747-BSN backup module provides remote programming capability for your secondary processor. This means that even with the programming device directly connected to the primary processor, the secondary SLC 500 processor memory can be programmed and/or monitored.

The primary 1747-BSN module provides an access point for a programming device to access the secondary processor. Using the 1747-BSN module, you can communicate from a programming device across the backup modules to the secondary processor. You must assign both the primary and the secondary processor the same node address in the DH+ link. The 1747-BSN module reserves the subsequent node address as an access-point address to the secondary processor; that is, this address is accessed by a programming device in order to program the secondary processor.

This means that if you set both processors node addresses to *n*, the programming device can communicate with the secondary processor with the node address *n*+1. Therefore, do not use node address *n*+1 for any other device on the DH+ network.

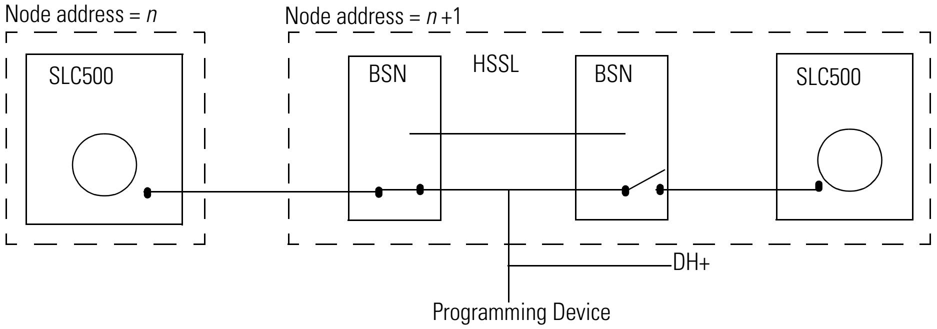
For example, a primary processor with node number 16 on DH+ network has the same address switch setting as the secondary processor node.

However, in this situation, the terminal addressed to station 16 attaches to the primary processor and the terminal addressed to station 17 attaches to the secondary processor.

IMPORTANT We recommend assigning the same node address (*n*) for both processors (primary and secondary) in a redundant system. You must also reserve the next node address (*n*+1) because this address is the access-point address to the secondary processor.

The figure below shows how the programming device sees the secondary processor which is not physically connected to the DH+ link.

IMPORTANT The DH+ connection to the secondary processor is meant for remote programming only. Also, only one terminal at a time may connect to the secondary processor. No DH+ messages can be sent to or from the secondary processor.



Data Table Transfer

Each 1747-BSN module is capable of transferring up to 2 KWord of the SLC 5/0x data table. This capacity is increased as the number of 1747-BSN modules per chassis is increased in a redundant system. For example, a system with eight 1747-BSN's in each chassis is capable of transferring up to 16 KWord of synchronizing data table information.

The 2 KWord that can be transferred by a single module is divided into 16 blocks of 128 words (that is equal to the maximum length of a COP instruction). The address space M0:s.3500 through M0:s.5547 is used by the primary SLC 5/0x to send the data. The address space M1:s.3500 through M1:s.5547 is used by the secondary SLC 5/0x to read the data from the secondary 1747-BSN. In addition, the address spaces M0:s.3400 through M0:s.3499; and M1:s.3400 through M1:s.3499 are used for status and control exchange between the SLC 5/0x and the 1747-BSN.

How the Backup System Operates

Some amount of support ladder program is necessary to make both modules operate properly in a backup system.

Both processors have a transmitter program and a receiver program. At system startup, the processor reads the Module Status Word (MSW) and the System Status Word (SSW) from the 1747-BSN to determine its state. If the processor is in the primary state, it executes the transmitter program. If the processor is in the secondary state, it executes the receiver program.

Primary Program

Each data block that can be transferred from the primary to secondary SLC 5/0x is identified by its address in the M0/M1 files. The application program must transfer each block using the following procedure.

A MOV instruction is used to get the Data Transfer Status Word (DTSW) and to see if the block can be transferred from the SLC 5/0x to the 1747-BSN (if the data block status bit is clear). When this bit is clear, the new data block must be copied to this corresponding M0 address. Then the data block control bit in the Data Transfer Control Word (DTCW) must be set to advise the 1747-BSN that this new data block is ready to transfer to the secondary system. This bit must be held set until the corresponding bit in the DTSW is set by the 1747-BSN. The bit must then be cleared.

Secondary Program

For each block that can be received by the secondary SLC 5/0x, its application program must do the following procedure.

A MOV instruction is used to get the Data Transfer Status Word (DTSW) and to see if new data for the block is ready for reading in the secondary 1747-BSN module. When this bit is set, the new data block must be copied from the 1747-BSN M1 file. Then its corresponding bit in the Data Transfer Handshake Word (DTHW) must be set to advise the 1747-BSN that the data block was already received. This bit must be held set until the corresponding bit in DTSW is cleared by the 1747-BSN module. The bit must then be cleared.

Backup System Theory of Operation

A redundant system using the 1747-BSN can be initially defined as an "Asynchronized Data Transfer" system. While the Input Image Table is automatically acquired from the RIO link by the secondary system, the Data Table is transferred by an application program written by the user. Try to minimize the amount of data to be transferred from the primary to the secondary processor because the program logic generates the same outputs based on the same inputs.

All timer and counter values must be transferred at least once because the primary and backup processors may have started at different times and are not synchronized. Therefore, timer and counter data may be different in the two processors. For example, after a faulted processor is repaired and reinstalled as a backup, the timer and counter accumulated values, as well as the control words, in the two processors are different.

The number of data table words to be transferred from a primary to a secondary processor is dependent on the user program and/or user architecture.

Startup Sequencing

When the system is powered up for the first time, the following sequence must be followed:

1. Disconnect both the primary and secondary SLC 5/0x from DH+ or DH485.
2. Download programs containing the desired DH+ or DH485 node address to each processor. Leave both SLC 5/0x in program mode, and power down both systems.
3. Apply power to the system that is intended to be primary.
4. Apply power to the system that is intended to be secondary.
5. Change the primary SLC processor to Run mode.
6. Change the secondary SLC processor to Run mode.

Power-up Sequencing

To determine the primary and secondary systems, use the following powerup sequence:

1. Apply power to the system that is intended to be primary.
2. After the primary SLC 5/0x has been powered-up and is operating correctly, apply power to the system that is intended to be secondary.

If both systems are powered-up at the same time, they use a “talk-and-listen algorithm” to determine which system is going to be primary and which is going to be secondary. The RIO and DH+ relays remain open until only one of the systems assumes primary status. This ensures that only one of the SLC 5/0x processors takes control over the RIO and DH+ links.

Processor Mode Change Sequencing

The 1747-BSN does not control the operating mode of the SLC 5/0x in either the primary or the backup systems. If a processor mode change is required, it should be done primarily in the secondary system. After that, if required, the processor mode change should also be done in the primary processor.

If the primary operating mode is changed from Run mode to Program mode, the secondary system takes control.

ATTENTION

Use the processor mode change key to change the primary operating mode. Unexpected operation or system shutdown could occur if you use a programming terminal to change the primary operating mode from Remote Run to Remote Program.

Fail-over Sequencing

When the primary system fails, the communications relays are opened immediately. The primary 1747-BSN ceases its data transfer function and sends switching commands to the secondary 1747-BSN. After all the RIO communications terminate, the secondary communication channel relays are closed. Then the former secondary SLC is ready to take control of the system.

Since the secondary’s input image table has been loaded on-line with real data from the Smart Switch, the application program does not have to delay the use of its input image table until the completion of at least one complete RIO scan. The secondary’s (now primary) input image table always has valid data.

Power Down Sequencing

No special considerations (beyond the standard precautions for a programmable controller) are required when powering down the system. Power down should begin with the secondary system and proceed to the primary. If the primary system is powered down first, a switchover occurs.

Restarting a Failed System

A failed system that has been repaired can be restarted as described in the following sequence. It is not necessary to power-down or change the mode of primary SLC 5/0x.

1. Set the secondary SLC 5/0x (processor of the repaired system) to program mode.
2. Apply power to the secondary system.
3. Download the applications program to the secondary SLC 5/0x.
4. Change the backup processor mode of operation from Program mode to Run mode.

Configuration and Programming

This chapter contains information necessary to:

- understand remote I/O image files
- understand RIO configuration using G files
- control and view RIO devices using the M0 and M1 files
- understand slot addressing
- quickly configure the RIO Scanner

Understanding Remote Input and Output Image Files

The SLC system allows you to assign up to 32 words of input and output image data to a scanner. This allows your scanner to access a maximum of four full logical racks (512 input and output points) of data from remote devices.

The illustration on page 5-2 shows how logical racks, logical groups, and words are allocated within the I/O image files. Note that this illustration describes the input image file. The scanner's output image file is the same, except that its addressing scheme starts with O:e.O and ends with O:e.31.

			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Logical Rack 0	Rack 0 Group 0	Word 0																	I:3.0
	Rack 0 Group 1	Word 1																	I:3.1
	Rack 0 Group 2	Word 2																	I:3.2
	Rack 0 Group 3	Word 3																	I:3.3
	Rack 0 Group 4	Word 4																	I:3.4
	Rack 0 Group 5	Word 5																	I:3.5
	Rack 0 Group 6	Word 6																	I:3.6
	Rack 0 Group 7	Word 7																	I:3.7
Logical Rack 1	Rack 1 Group 0	Word 8																	I:3.8
	Rack 1 Group 1	Word 9																	I:3.9
	Rack 1 Group 2	Word 10																	I:3.10
	Rack 1 Group 3	Word 11																	I:3.11
	Rack 1 Group 4	Word 12																	I:3.12
	Rack 1 Group 5	Word 13																	I:3.13
	Rack 1 Group 6	Word 14																	I:3.14
	Rack 1 Group 7	Word 15																	I:3.15
Logical Rack 2	Rack 2 Group 0	Word 16																	I:3.16
	Rack 2 Group 1	Word 17																	I:3.17
	Rack 2 Group 2	Word 18																	I:3.18
	Rack 2 Group 3	Word 19																	I:3.19
	Rack 2 Group 4	Word 20																	I:3.20
	Rack 2 Group 5	Word 21																	I:3.21
	Rack 2 Group 6	Word 22																	I:3.22
	Rack 2 Group 7	Word 23																	I:3.23
Logical Rack 3	Rack 3 Group 0	Word 24																	I:3.24
	Rack 3 Group 1	Word 25																	I:3.25
	Rack 3 Group 2	Word 26																	I:3.26
	Rack 3 Group 3	Word 27																	I:3.27
	Rack 3 Group 4	Word 28																	I:3.28
	Rack 3 Group 5	Word 29																	I:3.29
	Rack 3 Group 6	Word 30																	I:3.30
	Rack 3 Group 7	Word 31																	I:3.31
			17_8	16_8	15_8	14_8	13_8	12_8	11_8	10_8	7_8	6_8	5_8	4_8	3_8	2_8	1_8	0_8	

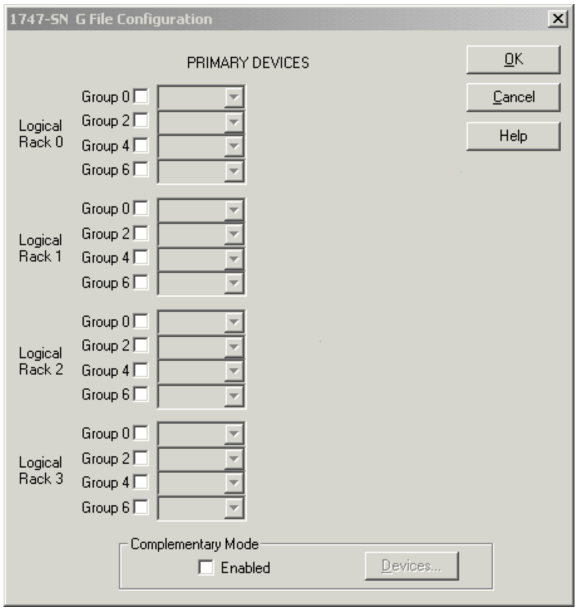
The 1747-BSN module's I/O image structure is described below:

- The I/O image file consists of four logical racks (numbered 0, 1, 2, and 3) of input image and four logical racks of output image.
- Each logical rack consists of eight logical groups (numbered 0, 1, 2, 3, 4, 5, 6, and 7).
- Each logical group consists of two words (an input word and an output word).
- Each word consists of two bytes (a high and a low byte). Low byte is bits 0 through 7 and high byte is bits 8 through 15.
- Each byte consists of eight bits, with each bit having the ability to control one discrete I/O point.

RIO Configuration Using G Files

When you program your SLC system, use the G file to configure the scanner's I/O image file. The Backup Scanner's G file configuration is based on the devices that you have on the RIO link. G file configuration consists of setting logical device starting addresses and the logical device image size of each physical device/adapter with which the scanner communicates.

You enter G file configuration information using programming software. The default RSLogix 500 G file configuration screen is shown below. See Appendix B for further details on G file configuration.



TIP

For RSLogix 500 version 5.50 and later, configure the 1747-BSN M0/M1 size for 5548 words so that this non-generic G file configuration screen appears after you click on the configure G file button. For RSLogix 500 versions prior to 5.50, configure the 1747-BSN M0/M1 size for 5547 words in order for this non-generic G file configuration screen to appear after you click on the configure G file button.

Neither your application program nor your programming device can access or alter the G file while on-line with the processor. To change the G file, you must go off-line into the program file, make any necessary changes, and download the program containing the altered configuration. The G file consists of five words which are described on the following page.

Word 0 - contains scanner information for the SLC processor. Your programming device *automatically* sets up Word 0. Do not attempt to alter word 0.

IMPORTANT The term “primary” is used in conjunction with the term “complementary,” when referring to a complementary I/O configuration. Primary refers to I/O image space found in Logical Racks 0 through 3 when in complementary I/O mode. Normal refers to the same image space (racks 0-3) when not in complementary I/O mode.

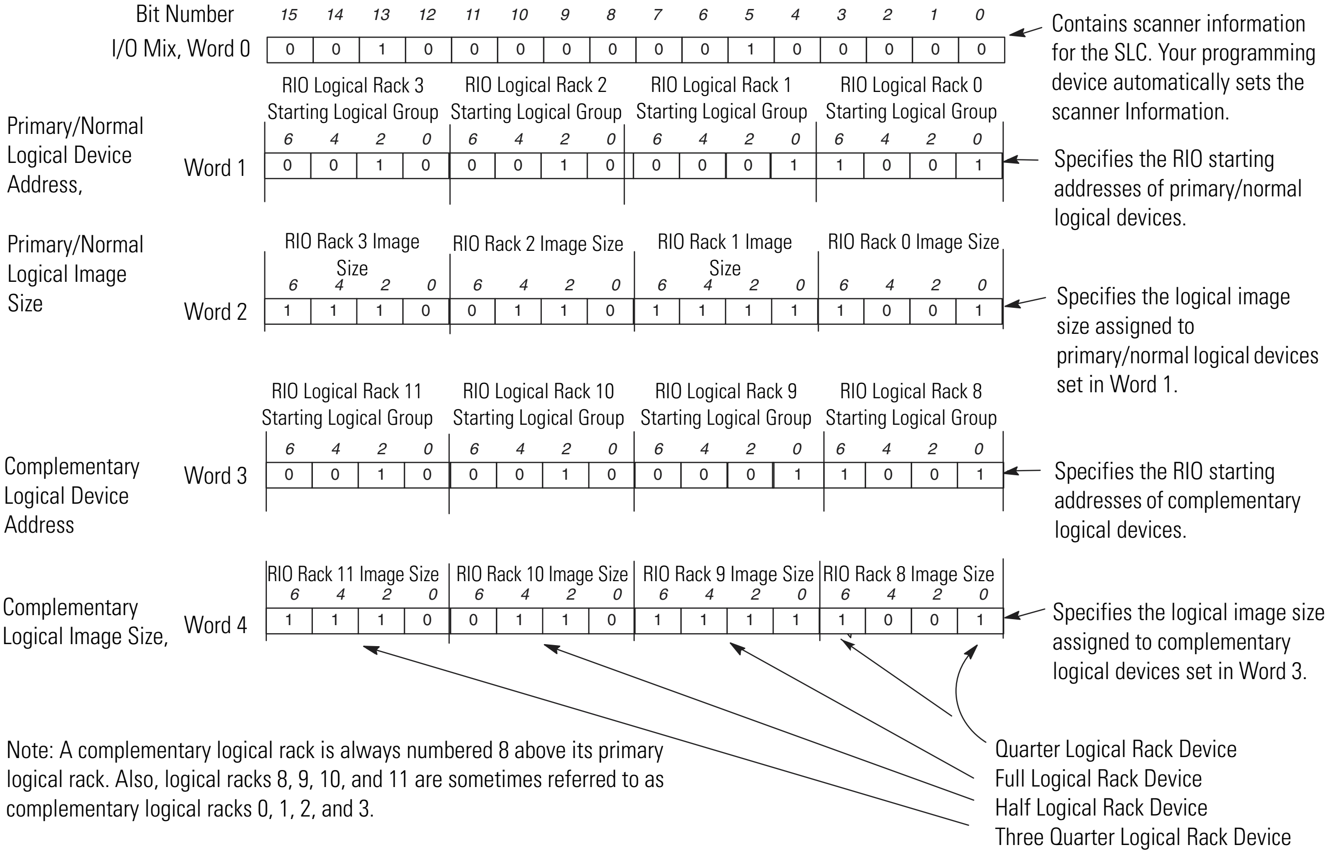
Word 1, Primary/Normal Logical Device Address - specifies the logical starting address of each primary/normal RIO link device. The logical address consists of the logical rack number (0, 1, 2, or 3) and starting logical group (0, 2, 4, or 6). Each bit in this word represents a logical address. To specify an address, place a 1 at the bit corresponding to the starting logical address of each logical device.

Word 2, Primary/Normal Device Logical Image Size - specifies the logical image size (amount of scanner I/O image) of the devices set in word 1. As with word 1, these bits correspond to RIO logical rack and logical group numbers. To specify image size, place a 1 at each group a device occupies.

Word 3, Complementary Logical Device Address - specifies the logical starting address of each complementary RIO link device. The logical address consists of the logical rack number (8, 9, 10, or 11 because a complementary device is always 8 above its primary) and starting logical group (0, 2, 4, or 6). Each bit in this word represents a logical address. To specify an address, place a 1 at the bit corresponding to the starting logical address of each logical device.

Word 4, Complementary Device Logical Image Size - specifies the logical image size (amount of scanner I/O image) of the complementary devices set in word 3. As with word 3, these bits correspond to RIO logical rack and logical group numbers. To specify image size, place a 1 at each group a device occupies.

IMPORTANT Setting device addresses in word 3 of the G file configures the system to operate in the complementary I/O mode. Not setting device addresses in word 3 causes the system to operate only in the *primary/normal* mode.



		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	I/O Mix, Word 0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
Primary/Normal Logical Device Address,	Word 1	0	0	1	0	0	0	1	0	0	0	0	1	1	0	0	1
Primary/Normal Logical Image Size	Word 2	1	1	1	0	0	1	1	0	1	1	1	1	1	0	0	1
Complementary Logical Device Address	Word 3	0	0	1	0	0	0	1	0	0	0	0	1	1	0	0	1
Complementary Logical Image Size,	Word 4	1	1	1	0	0	1	1	0	1	1	1	1	1	0	0	1

Contains scanner information for the SLC. Your programming device automatically sets the scanner Information.

Specifies the RIO starting addresses of primary/normal logical devices.

Specifies the logical image size assigned to primary/normal logical devices set in Word 1.

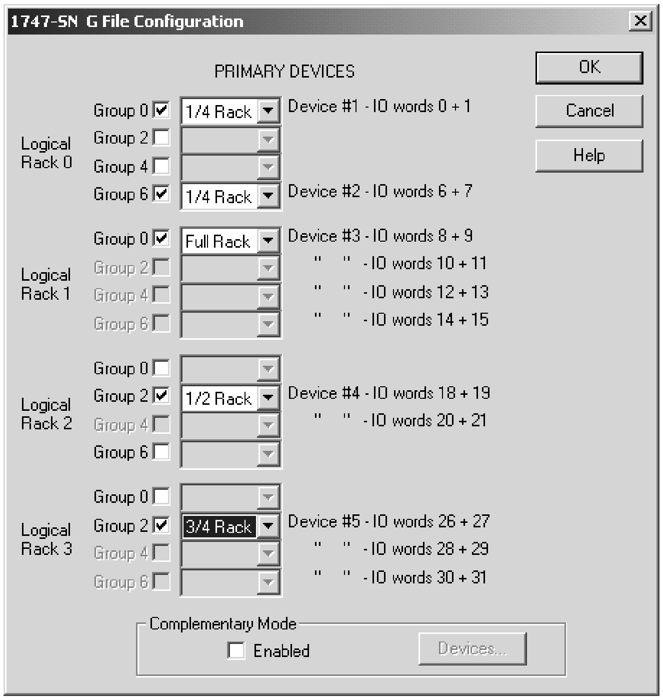
Specifies the RIO starting addresses of complementary logical devices.

Specifies the logical image size assigned to complementary logical devices set in Word 3.

Quarter Logical Rack Device
Full Logical Rack Device
Half Logical Rack Device
Three Quarter Logical Rack Device

Note: A complementary logical rack is always numbered 8 above its primary logical rack. Also, logical racks 8, 9, 10, and 11 are sometimes referred to as complementary logical racks 0, 1, 2, and 3.

When this G file is configured using RSLogix 500, it is displayed as:



PRIMARY DEVICES

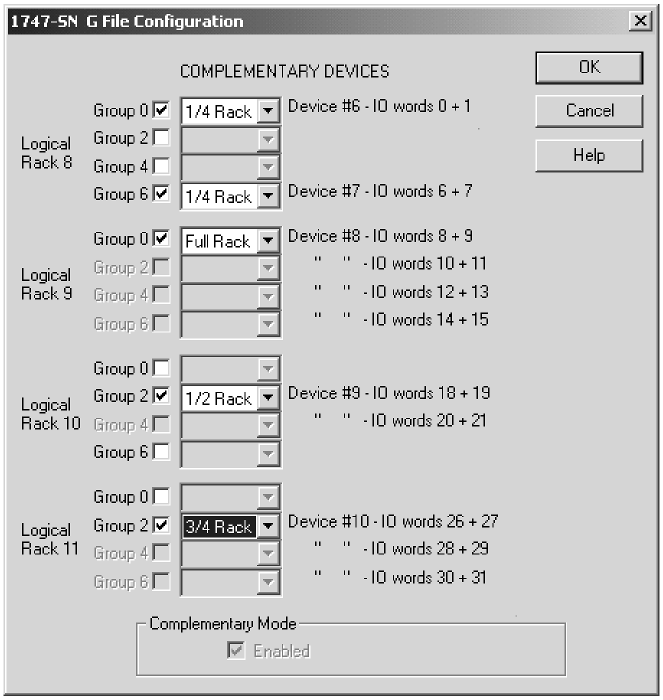
Logical Rack 0
Group 0 ☑ 1/4 Rack Device #1 - IO words 0 + 1
Group 2 ☐
Group 4 ☐
Group 6 ☑ 1/4 Rack Device #2 - IO words 6 + 7

Logical Rack 1
Group 0 ☑ Full Rack Device #3 - IO words 8 + 9
Group 2 ☐ " " - IO words 10 + 11
Group 4 ☐ " " - IO words 12 + 13
Group 6 ☐ " " - IO words 14 + 15

Logical Rack 2
Group 0 ☐
Group 2 ☑ 1/2 Rack Device #4 - IO words 18 + 19
Group 4 ☐ " " - IO words 20 + 21
Group 6 ☐

Logical Rack 3
Group 0 ☐
Group 2 ☑ 3/4 Rack Device #5 - IO words 26 + 27
Group 4 ☐ " " - IO words 28 + 29
Group 6 ☐ " " - IO words 30 + 31

Complementary Mode
☐ Enabled Devices...



COMPLEMENTARY DEVICES

Logical Rack 8
Group 0 ☑ 1/4 Rack Device #6 - IO words 0 + 1
Group 2 ☐
Group 4 ☐
Group 6 ☑ 1/4 Rack Device #7 - IO words 6 + 7

Logical Rack 9
Group 0 ☑ Full Rack Device #8 - IO words 8 + 9
Group 2 ☐ " " - IO words 10 + 11
Group 4 ☐ " " - IO words 12 + 13
Group 6 ☐ " " - IO words 14 + 15

Logical Rack 10
Group 0 ☐
Group 2 ☑ 1/2 Rack Device #9 - IO words 18 + 19
Group 4 ☐ " " - IO words 20 + 21
Group 6 ☐

Logical Rack 11
Group 0 ☐
Group 2 ☑ 3/4 Rack Device #10 - IO words 26 + 27
Group 4 ☐ " " - IO words 28 + 29
Group 6 ☐ " " - IO words 30 + 31

Complementary Mode
☑ Enabled

Rules for Configuring the Scanner

General

- The smallest portion of the scanner's I/O image that can be allocated to a single RIO device is two logical groups (1/4 logical rack).
- If a device is configured in word 1, there must be image allocated to it in word 2. This rule also applies to words 3 and 4 with the following exception: if word 3 = 1 and word 4 = 0, the complementary mode is selected even though no complementary devices are configured.
- A logical device's starting group must begin at even group numbers (0, 2, 4, or 6). Each bit in words 2 and 4 represent an even logical group number.

Concerning Complementary I/O

- It is valid for you to have a complementary device configured even if no associated primary device exists. Also, complementary devices do not have to be the same logical image size as the primary device.
- G file words 1 and 2 can both be zero (no primary devices). However, in this case there must be at least one complementary device configured in G file words 3 and 4.
- If there is at least one primary device configured in G file words 1 and 2, then words 3 and 4 can both be zero, or the G file size can be set to 3 (complementary mode not selected).
- The starting group of the primary and complementary chassis should be the same if they share the same image space. If the starting group is not the same, the image of the complementary device must not "cross over" into the space of a primary device. For example, if a primary device exists at Logical Rack 1, Logical Group 4, the maximum size of a complementary device at Logical Rack 9, Logical Group 0 is a half logical rack. The image does not cross over into Logical Group 4.
- A complementary device cannot be configured at locations where primary devices are configured unless they both start at the same location.

- If you configure your system so that complementary I/O is not selected (words 3 and 4 are zero), you must not set up any of the actual devices to be in the primary mode. If you do, the system flags the device as faulted and prevents the device from running.

- Control functions (i.e., device inhibit, device reset, and device output reset) are only selectable for the primary device, but also apply to the complementary device. Control functions for complementary devices cannot be exclusively enabled.

Example: G File Showing Primary and Complementary Device Configurations

In the example that follows, the scanner is configured to communicate with primary and complementary devices. Below are the device addresses and image sizes:

- Logical Racks 0/8, Logical Group 2 contain a primary 3/4 logical rack device and a complementary 3/4 logical rack device.

- Logical Racks 1/9, Logical Group 0 contain no primary device and a complementary 1/2 logical rack device.

- Logical Racks 1/9, Logical Group 6 contain a primary 1/4 logical rack device and a complementary 1/4 logical rack device.

- Logical Racks 2/10, Logical Group 0 contain a primary 3/4 logical rack device and a complementary 1/4 logical rack device.

- Logical Racks 3/11, Logical Group 2 contain a primary 1/4 logical rack device and a complementary 1/2 logical rack device.

- Logical Racks 3/11, Logical Group 6 contain a primary 1/4 logical rack device and no complementary device.

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
I/O Mix, Word 0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0

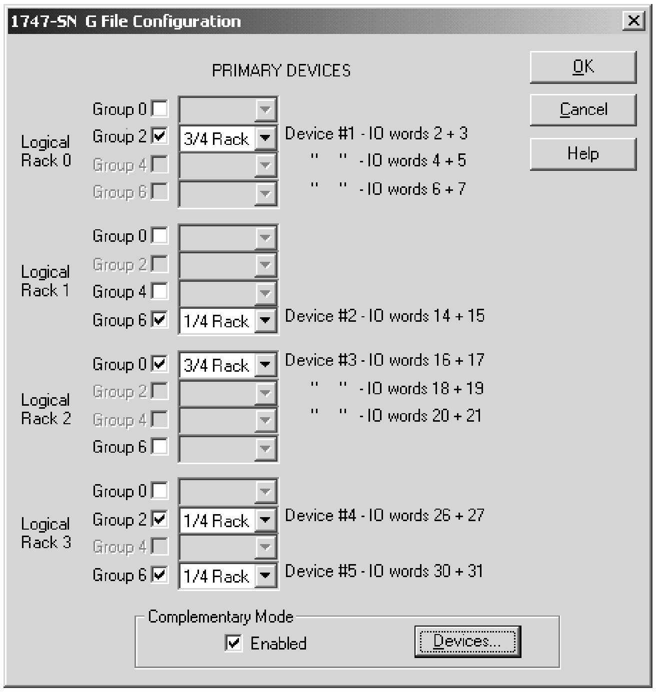
	RIO Logical Rack 3 Starting Logical Group				RIO Logical Rack 2 Starting Logical Group				RIO Logical Rack 1 Starting Logical Group				RIO Logical Rack 0 Starting Logical Group			
	6	4	2	0	6	4	2	0	6	4	2	0	6	4	2	0
Primary/Normal Logical Device Address, Word 1	1	0	1	0	0	0	0	1	1	0	0	0	0	0	1	0

	RIO Rack 3 Image Size				RIO Rack 2 Image Size				RIO Rack 1 Image Size				RIO Rack 0 Image Size			
	6	4	2	0	6	4	2	0	6	4	2	0	6	4	2	0
Primary/Normal Logical Image Size, Word 2	1	0	1	0	0	1	1	1	1	0	0	0	1	1	1	0

	RIO Logical Rack 11 Starting Logical Group				RIO Logical Rack 10 Starting Logical Group				RIO Logical Rack 9 Starting Logical Group				RIO Logical Rack 8 Starting Logical Group			
	6	4	2	0	6	4	2	0	6	4	2	0	6	4	2	0
Complementary Logical Device Address, Word 3	0	0	1	0	0	0	0	1	1	0	0	1	0	0	1	0

	RIO Rack 11 Image Size				RIO Rack 10 Image Size				RIO Rack 9 Image Size				RIO Rack 8 Image Size			
	6	4	2	0	6	4	2	0	6	4	2	0	6	4	2	0
Complementary Logical Image Size, Word 4	0	1	1	0	0	0	0	1	1	0	1	1	1	1	1	0

When this G file is configured using RSLogix 500, it is displayed as:



PRIMARY DEVICES

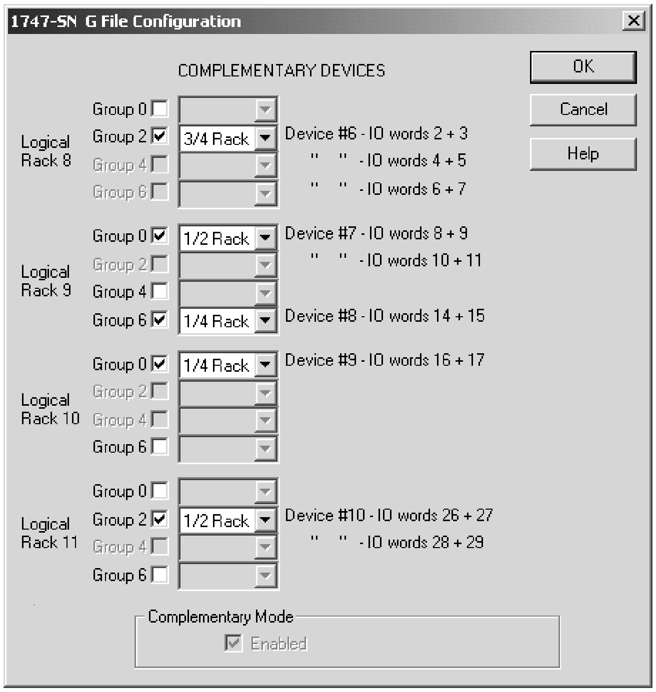
Logical Rack 0
Group 0 ☐
Group 2 ☑ 3/4 Rack — Device #1 - IO words 2 + 3
Group 4 ☐ — " " - IO words 4 + 5
Group 6 ☐ — " " - IO words 6 + 7

Logical Rack 1
Group 0 ☐
Group 2 ☐
Group 4 ☐
Group 6 ☑ 1/4 Rack — Device #2 - IO words 14 + 15

Logical Rack 2
Group 0 ☑ 3/4 Rack — Device #3 - IO words 16 + 17
Group 2 ☐ — " " - IO words 18 + 19
Group 4 ☐ — " " - IO words 20 + 21
Group 6 ☐

Logical Rack 3
Group 0 ☐
Group 2 ☑ 1/4 Rack — Device #4 - IO words 26 + 27
Group 4 ☐
Group 6 ☑ 1/4 Rack — Device #5 - IO words 30 + 31

Complementary Mode
☑ Enabled
Devices...



COMPLEMENTARY DEVICES

Logical Rack 8
Group 0 ☐
Group 2 ☑ 3/4 Rack — Device #6 - IO words 2 + 3
Group 4 ☐ — " " - IO words 4 + 5
Group 6 ☐ — " " - IO words 6 + 7

Logical Rack 9
Group 0 ☑ 1/2 Rack — Device #7 - IO words 8 + 9
Group 2 ☐ — " " - IO words 10 + 11
Group 4 ☐
Group 6 ☑ 1/4 Rack — Device #8 - IO words 14 + 15

Logical Rack 10
Group 0 ☑ 1/4 Rack — Device #9 - IO words 16 + 17
Group 2 ☐
Group 4 ☐
Group 6 ☐

Logical Rack 11
Group 0 ☐
Group 2 ☑ 1/2 Rack — Device #10 - IO words 26 + 27
Group 4 ☐ — " " - IO words 28 + 29
Group 6 ☐

Complementary Mode
☑ Enabled

Examples of Illegal Configurations

Having a primary device configured at Logical Rack 1, Logical Group 2 (bit 5) would be illegal since this image space is already being used by a complementary device. Having a complementary device configured at Logical Rack 10, Logical Group 2 (bit 9) would also be illegal since this image space is already being used by a primary device.

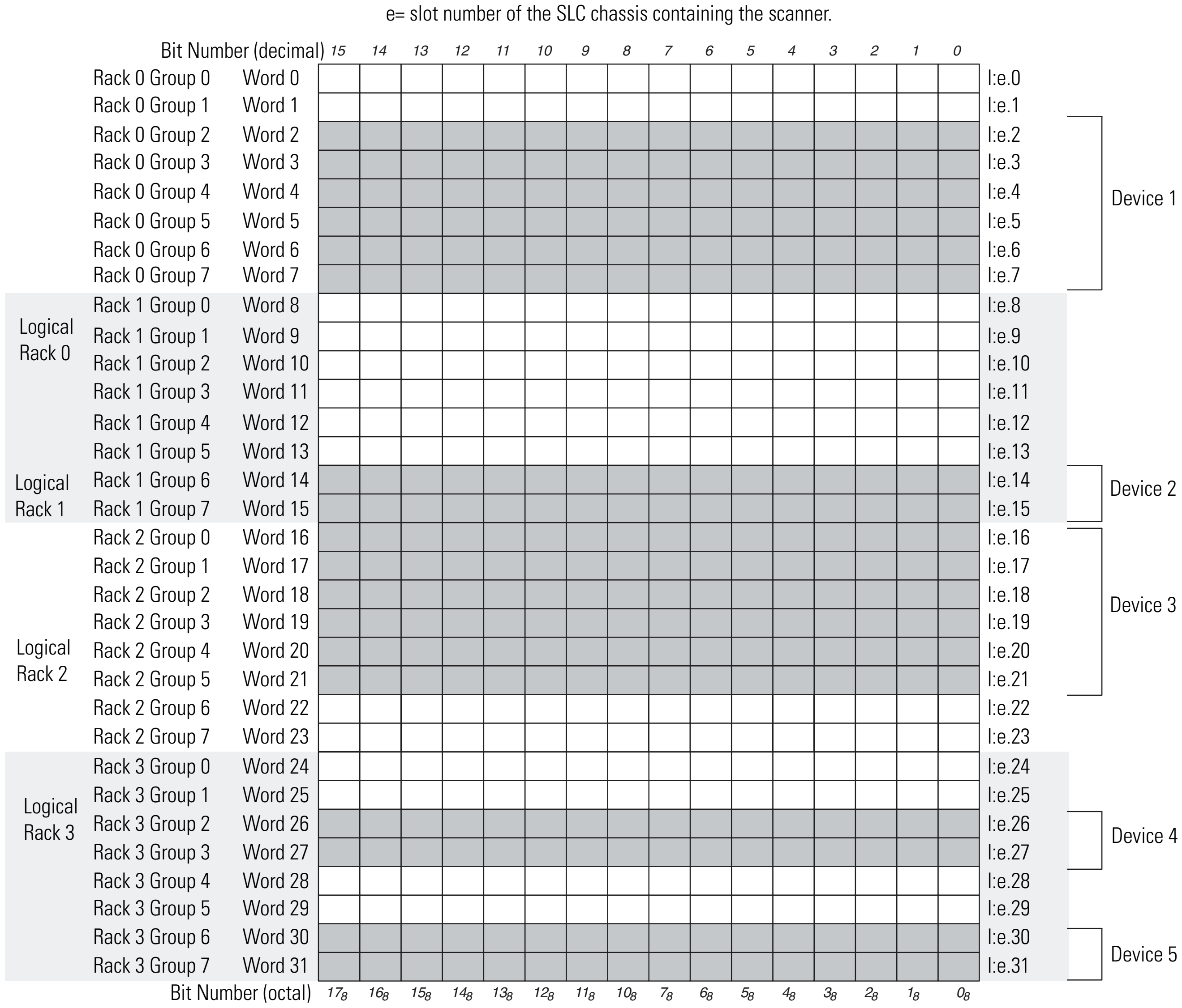
Note that the complementary device at Logical Rack 8, Logical Group 2 could be an ASB using 10 words (1-1/4 logical racks) of data, and thereby cross into RIO Logical Rack 9.

The G file configuration above would provide the primary and complementary input images to the scanner, which are illustrated on the following pages. The output images would be the same.

Example Scanner Input Image of the Primary Devices

Below are the primary device addresses and sizes. The following figure contains complementary device addresses and sizes.

- Device 1 - starting at Logical Rack 0, Logical Group 2 is a primary 3/4 logical rack device.
- Logical Rack 1, Logical Group 0 contains no primary device.
- Device 2 - starting at Logical Rack 1, Logical Group 6 is a primary 1/4 logical rack device.
- Device 3 - starting at Logical Rack 2, Logical Group 0 is a primary 3/4 logical rack device.
- Device 4 - starting at Logical Rack 3, Logical Group 2 is a primary 1/4 logical rack device.
- Device 5 - starting at Logical Rack 3, Logical Group 6 is a primary 1/4 logical rack device.



Example Scanner Input Image of the Complementary Devices

Below are the complementary device addresses and sizes. The following figure contains primary device addresses and sizes.

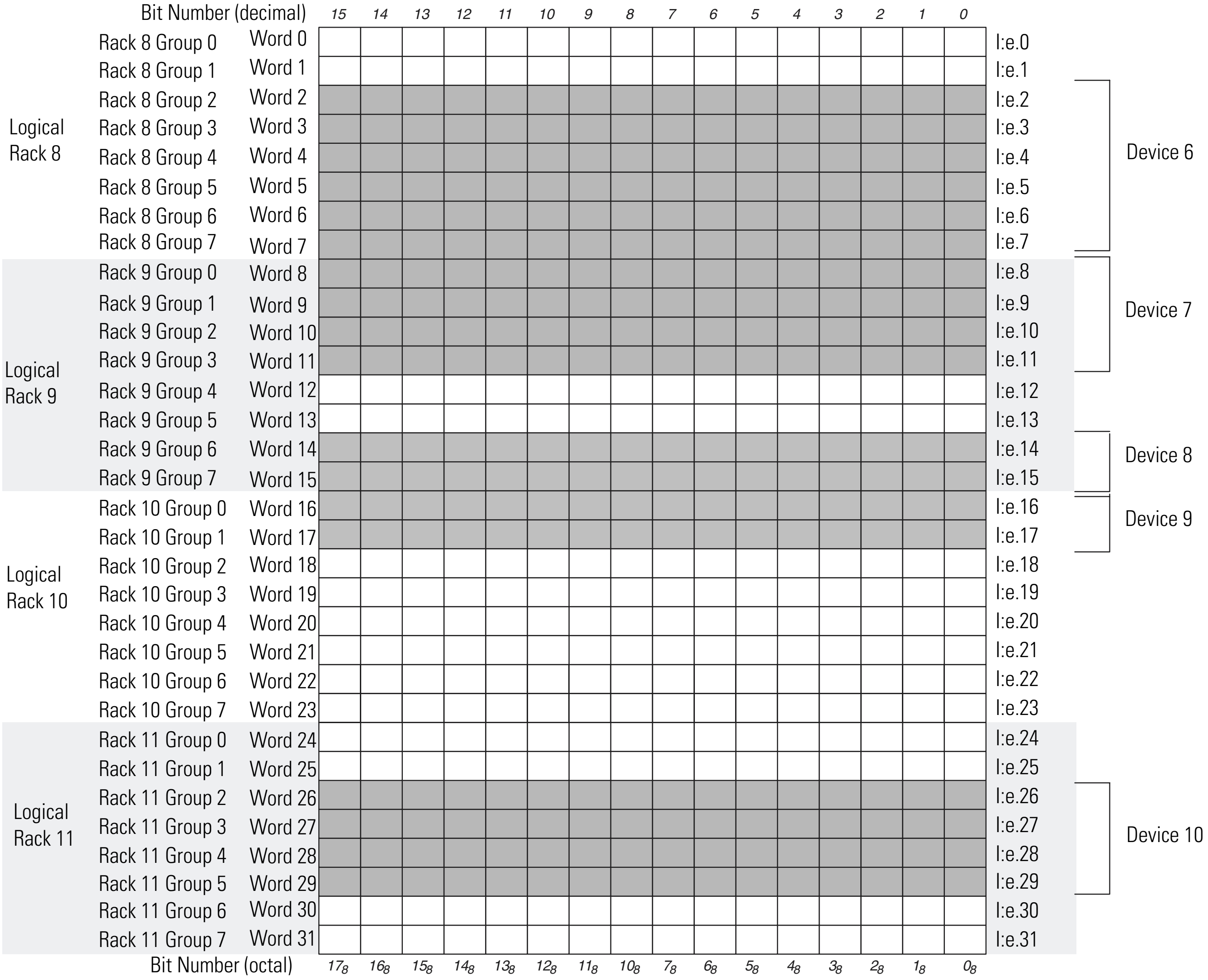
- Device 6 - starting at Logical Rack 8, Logical Group 2 is a complementary 3/4 logical rack device.
- Device 7 - starting at Logical Rack 9, Logical Group 0 is a complementary 1/2 logical rack device.

- Device 8 - starting at Logical Rack 9, Logical Group 6 is a complementary 1/4 logical rack device.

- Device 9 - starting at Logical Rack 10, Logical Group 0 is a complementary 1/4 logical rack device.

- Device 10 - starting at Logical Rack 11, Logical Group 2 is a complementary 1/2 logical rack device.

- Logical Rack 11, Logical Group 6 has no complementary device.



		Bit Number (decimal)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Logical Rack 8	Rack 8 Group 0	Word 0																	I:e.0	
	Rack 8 Group 1	Word 1																	I:e.1	
	Rack 8 Group 2	Word 2	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.2	Device 6
	Rack 8 Group 3	Word 3	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.3	
	Rack 8 Group 4	Word 4	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.4	
	Rack 8 Group 5	Word 5	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.5	
	Rack 8 Group 6	Word 6	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.6	
	Rack 8 Group 7	Word 7	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.7	
Logical Rack 9	Rack 9 Group 0	Word 8	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.8	Device 7
	Rack 9 Group 1	Word 9	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.9	
	Rack 9 Group 2	Word 10	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.10	
	Rack 9 Group 3	Word 11	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.11	
	Rack 9 Group 4	Word 12																	I:e.12	
	Rack 9 Group 5	Word 13																	I:e.13	
	Rack 9 Group 6	Word 14	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.14	Device 8
	Rack 9 Group 7	Word 15	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.15	
Logical Rack 10	Rack 10 Group 0	Word 16	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.16	Device 9
	Rack 10 Group 1	Word 17	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.17	
	Rack 10 Group 2	Word 18																	I:e.18	
	Rack 10 Group 3	Word 19																	I:e.19	
	Rack 10 Group 4	Word 20																	I:e.20	
	Rack 10 Group 5	Word 21																	I:e.21	
	Rack 10 Group 6	Word 22																	I:e.22	
	Rack 10 Group 7	Word 23																	I:e.23	
Logical Rack 11	Rack 11 Group 0	Word 24																	I:e.24	
	Rack 11 Group 1	Word 25																	I:e.25	
	Rack 11 Group 2	Word 26	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.26	Device 10
	Rack 11 Group 3	Word 27	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.27	
	Rack 11 Group 4	Word 28	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.28	
	Rack 11 Group 5	Word 29	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	I:e.29	
	Rack 11 Group 6	Word 30																	I:e.30	
	Rack 11 Group 7	Word 31																	I:e.31	
		Bit Number (octal)	17_8	16_8	15_8	14_8	13_8	12_8	11_8	10_8	7_8	6_8	5_8	4_8	3_8	2_8	1_8	0_8		

G-File Considerations For Configuring Remote I/O

You should understand the following information before you configure your scanner's G file.

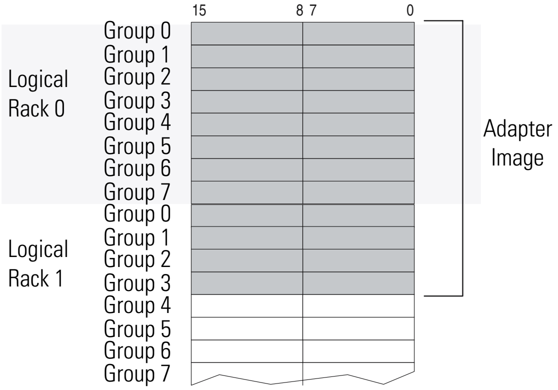
- You can only change the RIO configuration by modifying the G file while offline in your program file. Your application program cannot access the G file, nor can you modify it while online with your programming device. However, your SLC control program can dynamically inhibit and uninhibit RIO devices via the M0 file.
- RIO devices larger than 1 logical rack appear as multiple devices on the RIO link. Refer to the Crossing Logical Rack Boundaries section below.
- The address and size of the devices you list in the G file must match the settings of each RIO device.

Crossing Logical Rack Boundaries

You express remote I/O image boundaries in an even number of groups. For example, the 1747-ASB image can be any size from two logical groups up to 32 logical groups (four logical racks), in 2-logical group increments.If the scanner image assigned to an adapter is greater than 8 logical groups (one logical rack), the image crosses logical rack boundaries. If the scanner image assigned to an adapter is less than 8 logical groups, it too can cross a logical rack boundary depending upon the starting logical group number. The significance of crossing logical rack boundaries is discussed in the next section.

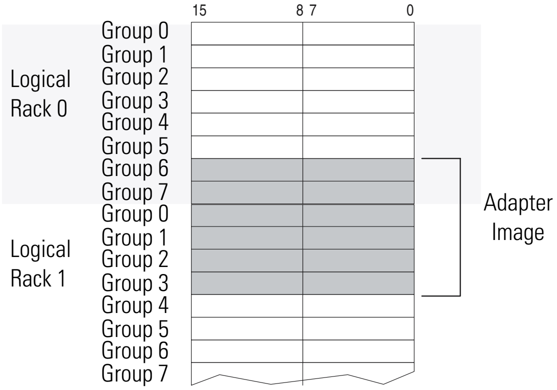
Examples 1 and 2 that follow show adapters with logical image sizes that cross logical racks 0 and 1. The image size of the adapter in example 1 consumes all of logical rack 0 (eight logical groups) and half of logical rack 1 (four logical groups). The image size of the adapter in example 2 consumes two groups in logical rack 0 and four groups in logical rack 1.

Example 1 – Scanner Input or Output Image



Adapter image is 12 logical groups in size and crosses a logical rack boundary due to its size.

Example 2 – Scanner Input or Output Image



Adapter image is 6 logical groups in size and crosses a logical rack boundary due to its starting logical group number.

Creating More than One Logical Rack Device

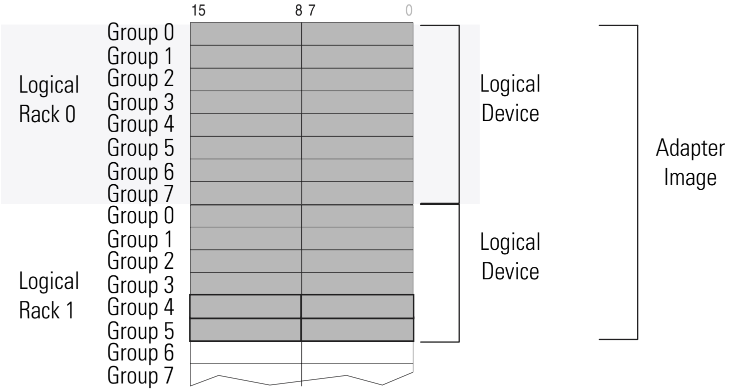
RIO discrete transfers occur on a logical device basis, not on an adapter basis. A logical device is any portion of a logical rack that is assigned to a single adapter.

When the scanner image assigned to an adapter is more than one logical device, the scanner sees the single physical device as multiple logical devices on the RIO link. The scanner communicates with each logical device independently, even if the logical devices are all assigned to one adapter. If a physical device image is more than one logical device, the following is true:

- The scanner does not update all of the adapter image at the same time. The number of logical devices determines the number of RIO discrete transfers that are needed to update the entire adapter image.

- The adapter may receive different communication commands for each logical device. In this case, the adapter decides which command it responds to.

In this example, the adapter is configured to start at Logical Rack 0, Logical Group 0, and uses 14 words of I/O image. Note that two RIO discrete transfers are required for the scanner to update the adapter image containing two logical devices.



Understanding M Files

M Files Overview

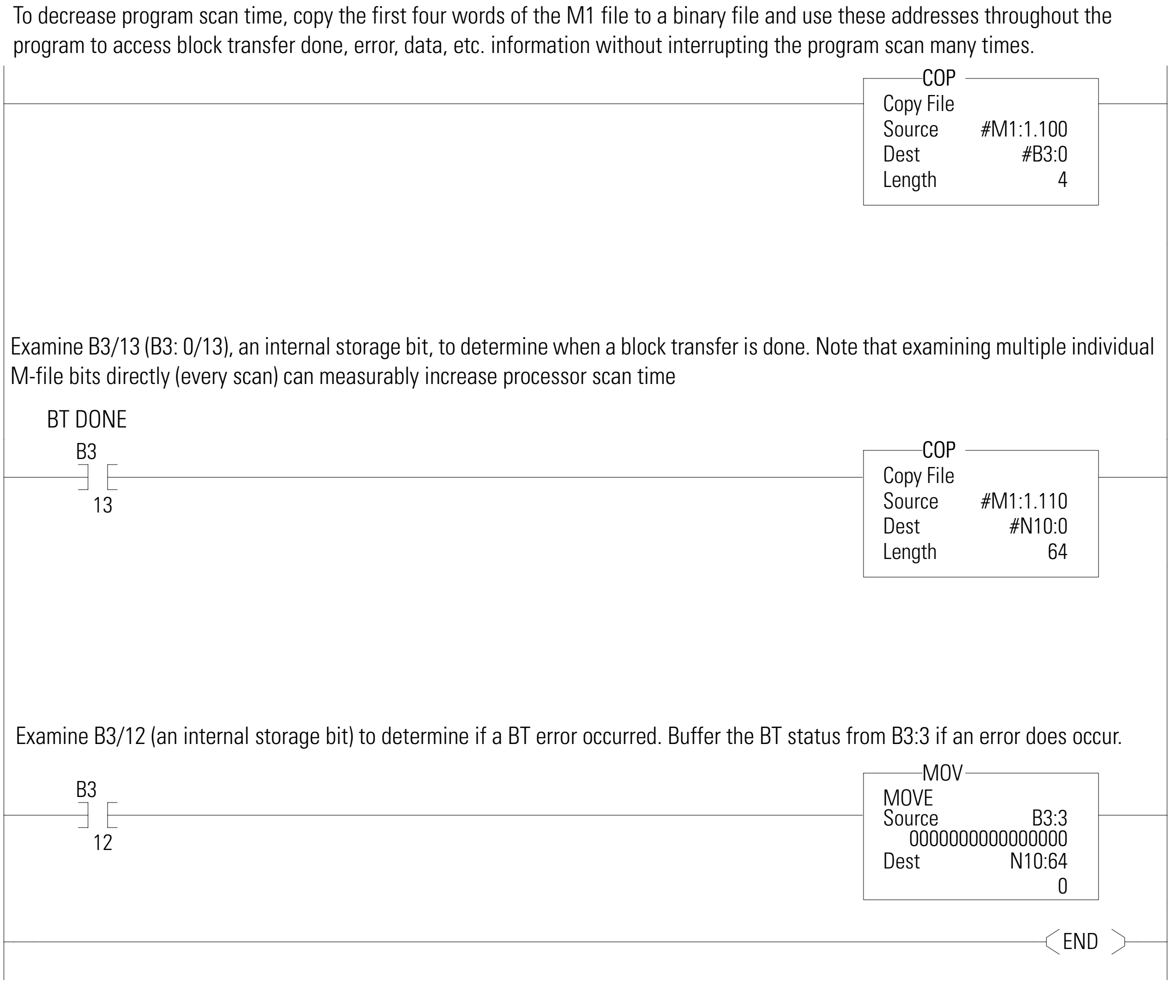
The scanner provides RIO device control and status information through the M0 and M1 files. The M0 file is a control file. The M1 file is a status file.

There is no image for M file data in SLC processors as there is for I/O data. The M files are buffers in the 1747-BSN module, accessible only via ladder logic instructions that address them. Each occurrence of a ladder instruction addressed with an M-file address is an interrupt to the ladder program scan. This is similar to the way Immediate I/O Instructions operate.

Instructions with the M0 file addressed write data to the M0 file in the 1747-BSN. Instructions with the M1 file addressed read data from the M1 file in the 1747-BSN.

M-file bits/words in the ladder program, therefore, impact the ladder scan time. If scan time is critical, use COP instructions to copy blocks of M1 file data to processor data file addresses throughout the program. It is more efficient to do one large M-file transfer than to do many small transfers.

It is also more efficient to address instructions in the ladder program using internal data files (binary, integer, etc.). Then use COP instructions to copy this data file to the M0 file in one large block at the end of the program. Refer to the ladder example that follows. For more information on M files, refer to Appendix B. You can find M file information relating to Block Transfer operations in Chapter 7, RIO Block Transfer.



To decrease program scan time, copy the first four words of the M1 file to a binary file and use these addresses throughout the program to access block transfer done, error, data, etc. information without interrupting the program scan many times.

Examine B3/13 (B3: 0/13), an internal storage bit, to determine when a block transfer is done. Note that examining multiple individual M-file bits directly (every scan) can measurably increase processor scan time

Examine B3/12 (an internal storage bit) to determine if a BT error occurred. Buffer the BT status from B3:3 if an error does occur.

IMPORTANT If you are using an SLC 5/02 processor, M file data cannot be directly monitored. To monitor M files, you must move the M file words into an SLC file that can be monitored, e.g., an integer “N” file. SLC 5/03 or later processors allow you to monitor M files directly. However, do not address M file bits more than necessary throughout your application program. The processor accesses M files like immediate I/O. Therefore, excessive addressing of M files can greatly increase SLC processor scan time. For more information on M files, refer to Appendix B.

M0 Control File Description

You can control the operation of individual devices on the RIO link with M0 word 8 through M0 word 27 (M0:e.8 through M0:e.27). Through your application program, you can use the M0 file to:

- Device Inhibit – command the 1747-BSN module to stop scanning an RIO device by using words 8 to 11.
- Device Reset – command an RIO device's outputs to reset while the SLC processor is in Run or Test mode by using words 16 to 19.
- Remote Output Reset – command an RIO device's outputs to reset upon the SLC processor leaving Run mode (regardless of the RIO device's *Hold Last State* setting), or while in Test mode by using words 24 to 27.

If you *do not* modify the Device Reset and Remote Output Reset words, the device outputs reflect the scanner output image whenever the SLC processor is in Run mode. If the SLC processor is in Program, Test, or Fault mode, it instructs the device to reset its outputs.

M file data is nonretentive. Upon entering Run or Test modes, the SLC processor sets the M0 file to a default state. The processor does not use the M0 file until a full program scan occurs (after entering Run mode). This allows you to change the M file settings before they take effect.

IMPORTANT The 1747-BSN module does not use M0 words 0 to 7.

	M0 (Control) File- RIO Device Control Words	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Device Inhibit Control	Logical Rack 0 Device Inhibit Word 8	x	x	x	x	x	x	x	x	x	x	x	x	0	1	1	0
	Logical Rack 1Device Inhibit Word 9	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0
	Logical Rack 2 Device Inhibit Word 10	x	x	x	x	x	x	x	x	x	x	x	x	1	0	0	1
	Logical Rack 3 Device Inhibit Word 11	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	1
Device Reset Control	Logical Rack 0 Device Inhibit Word 16	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0
	Logical Rack 1 Device Inhibit Word 17	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0
	Logical Rack 2 Device Inhibit Word 18	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0
	Logical Rack 3 Device Inhibit Word 19	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0
Output Reset Control	Logical Rack 0 Remote Output Reset Word 24	x	x	x	x	x	x	x	x	x	x	x	x	1	0	0	1
	Logical Rack 1 Remote Output Reset Word 25	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	1
	Logical Rack 2 Remote Output Reset Word 26	x	x	x	x	x	x	x	x	x	x	x	x	0	0	1	0
	Logical Rack 3 Remote Output Reset Word 27	x	x	x	x	x	x	x	x	x	x	x	x	0	0	1	0

e = slot number of the SLC rack containing the scanner
x = bit not used/defined

IMPORTANT Control functions (i.e., device inhibit, device reset, and device output reset) are only selectable for the primary device, but also apply to the complementary device. Control functions for complementary devices cannot be exclusively enabled.

M0 File - RIO Device Inhibit Control

M0 Words 8 through 11 - you use these words to command the scanner to stop scanning logical racks 0, 1, 2, and 3. Bits 0 to 3 in each word correspond to I/O group locations within logical racks 0, 1, 2, and 3.

To stop scanning (inhibit) a device listed in the configuration (G) file, set the bit corresponding to the starting group address of the device to 1. Setting bits that do *not* correspond to the device logical *starting group address* does not inhibit the device. To resume scanning a device, reset the bit (which corresponds to the starting group address of the device) to 0.

Inhibiting a device does not affect the current settings of the *Device Fault Status* (words 12 to 15 of the M1 file). Inhibited devices *eventually time out* and either return to their last state or reset (depending on the device's last state setting).

Default: When the processor enters the Run mode, the scanner automatically inhibits any device not configured in the G file (bit set to 1). Attempting to inhibit an unconfigured device has no effect.

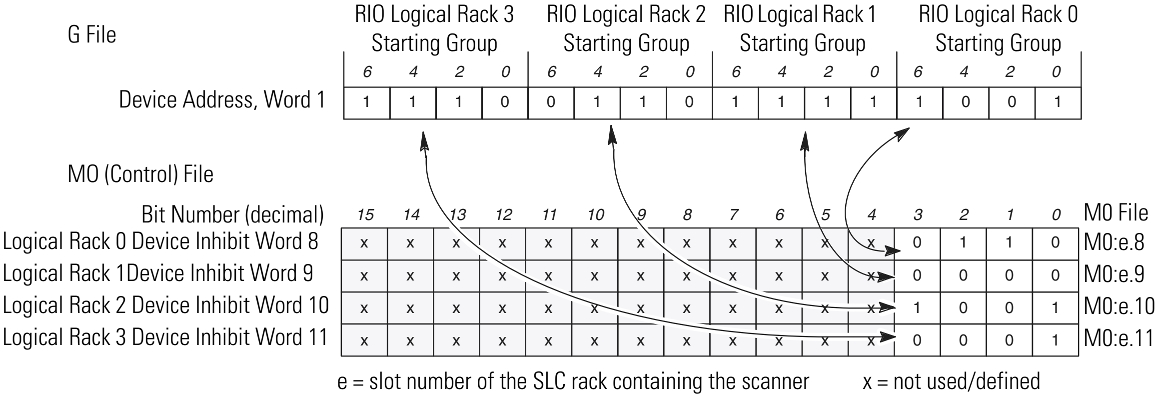
M0 (Control) File Words 8 through 11

	Not Defined												Starting Group				
													6	4	2	0	
Bit Number (decimal)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M0 File
Logical Rack 0 Device Inhibit Word 8	x	x	x	x	x	x	x	x	x	x	x	x	0	1	1	0	M0:e.8
Logical Rack 1Device Inhibit Word 9	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	M0:e.9
Logical Rack 2 Device Inhibit Word 10	x	x	x	x	x	x	x	x	x	x	x	x	1	0	0	1	M0:e.10
Logical Rack 3 Device Inhibit Word 11	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	1	M0:e.11

e = slot number of the SLC rack containing the scanner
x = not used/defined

Example of Device Inhibit Control

The 1747-BSN Scanner inhibits (sets to 1) the bits in M0:e.8 through M0:e.11 (by default) wherever there are no configured devices present. The illustration below compares the configured devices (G file word 2) to the groups that the scanner automatically inhibits.



G File	RIO Logical Rack 3 Starting Group				RIO Logical Rack 2 Starting Group				RIO Logical Rack 1 Starting Group				RIO Logical Rack 0 Starting Group			
	6	4	2	0	6	4	2	0	6	4	2	0	6	4	2	0
Device Address, Word 1	1	1	1	0	0	1	1	0	1	1	1	1	1	0	0	1

M0 (Control) File

Bit Number (decimal)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M0 File
Logical Rack 0 Device Inhibit Word 8	x	x	x	x	x	x	x	x	x	x	x	x	0	1	1	0	M0:e.8
Logical Rack 1Device Inhibit Word 9	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	M0:e.9
Logical Rack 2 Device Inhibit Word 10	x	x	x	x	x	x	x	x	x	x	x	x	1	0	0	1	M0:e.10
Logical Rack 3 Device Inhibit Word 11	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	1	M0:e.11

e = slot number of the SLC rack containing the scanner x = not used/defined

M0 File - RIO Device Reset Control

M0 Words 16 through 19 – you use these words to command a reset (0) of RIO device outputs when the SLC processor is in Run or Test mode. This allows you to selectively reset logical device outputs based on a previous condition(s) that you defined. Bits 0 to 3 correspond to the logical I/O group locations within logical racks 0, 1, 2, and 3.

To command an RIO device to a reset (0) condition (from Run or Test mode), set the bit corresponding to the starting logical address of the device to 1. Setting bits that do not correspond to a device starting address does not force a reset. To remove the reset condition, reset the bit (corresponding to the device logical starting address) to 0. Refer to the mode table below.

Default: The SLC processor resets all bits in this field to 0 when it enters Run or Test mode.

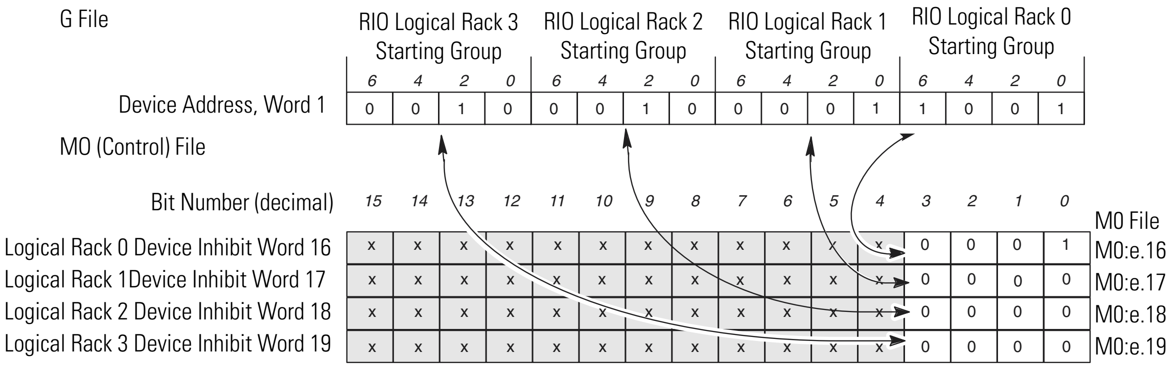
M0 (Control) File Words 16 through 19

	Not Defined												Starting Group				
													6	4	2	0	
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M0 File
Logical Rack 0 Device Inhibit Word 16	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	M0:e.16
Logical Rack 1Device Inhibit Word 17	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	M0:e.17
Logical Rack 2 Device Inhibit Word 18	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	M0:e.18
Logical Rack 3 Device Inhibit Word 19	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	M0:e.19

e = slot number of the SLC rack containing the scanner
x = bit not used/ defined

Example of Device Reset Control

The application has commanded the device starting at Logical Rack 0, Group 0 (M0:e.16/0) to a reset condition (bit set to 1). The default setting for all device reset bits is 0.



	6	4	2	0	6	4	2	0	6	4	2	0	6	4	2	0
Device Address, Word 1	0	0	1	0	0	0	1	0	0	0	0	1	1	0	0	1

Bit Number (decimal)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M0 File
Logical Rack 0 Device Inhibit Word 16	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	1	M0:e.16
Logical Rack 1Device Inhibit Word 17	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	M0:e.17
Logical Rack 2 Device Inhibit Word 18	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	M0:e.18
Logical Rack 3 Device Inhibit Word 19	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	M0:e.19

e = slot number of the SLC rack containing the scanner x = bit not used/defined

M0 File - Remote Output Reset Control

M0 Words 24 through 27 – use these words to command a logical device to reset all of its outputs when the SLC processor leaves the Run mode and enters the Test, Program, or Fault mode (regardless of the device's *Hold Last State* setting).

Resetting the bit (corresponding to the starting address of a device) to 0 allows the Hold Last State switch on the logical device to determine output operation when the SLC processor leaves the Run mode. Setting the bit to 1 commands all outputs off (regardless of the device's *Hold Last State* setting).

Only the device's logical starting address bit matters. Setting other bits has no effect. Bits 0 to 3 correspond to I/O group locations within logical racks 0, 1, 2, and 3.

Default: When the processor enters Run or Test mode, the scanner sets the starting address bit of each device configured in the G file to 1.

ATTENTION

The use of the device's Hold Last State switch can result in its outputs remaining energized when not under control of the SLC processor. Only experienced SLC programmers should use this function.

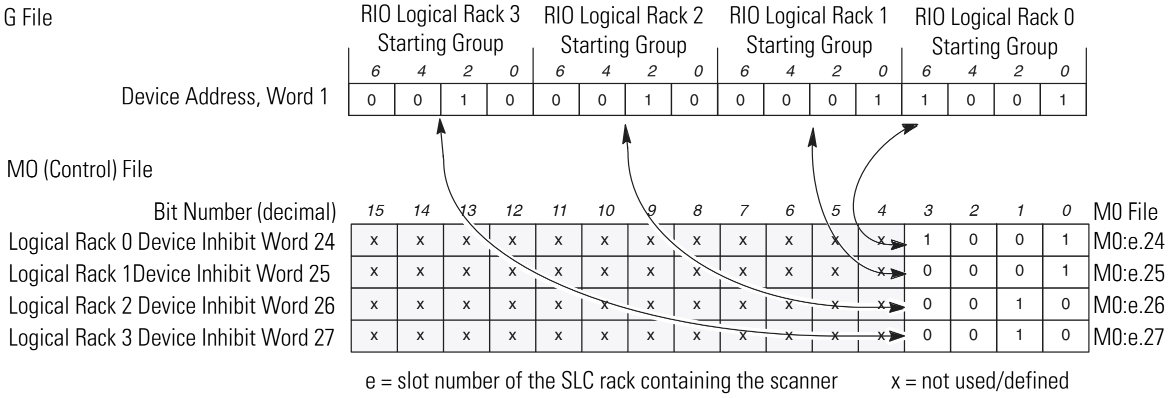
MO (Control) File Words 24 through 27

	Not Defined												Starting Group				
													6	4	2	0	
Bit Number (decimal)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M0 File
Logical Rack 0 Device Inhibit Word 24	x	x	x	x	x	x	x	x	x	x	x	x	1	0	0	1	M0:e.24
Logical Rack 1Device Inhibit Word 25	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	1	M0:e.25
Logical Rack 2 Device Inhibit Word 26	x	x	x	x	x	x	x	x	x	x	x	x	0	0	1	0	M0:e.26
Logical Rack 3 Device Inhibit Word 27	x	x	x	x	x	x	x	x	x	x	x	x	0	0	1	0	M0:e.27

e = slot number of the SLC rack containing the scanner x = not used/defined

Example of Remote Output Reset Control

By default the scanner sets the bits in M0:e.24 through M0:e.27 to 1 wherever there are configured devices present. This commands all devices' outputs to reset regardless of their Hold Last State switch. The application program can remove commanded reset of devices by resetting bits to 0.



G File

	RIO Logical Rack 3 Starting Group				RIO Logical Rack 2 Starting Group				RIO Logical Rack 1 Starting Group				RIO Logical Rack 0 Starting Group			
	6	4	2	0	6	4	2	0	6	4	2	0	6	4	2	0
Device Address, Word 1	0	0	1	0	0	0	1	0	0	0	0	1	1	0	0	1

MO (Control) File

Bit Number (decimal)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M0 File
Logical Rack 0 Device Inhibit Word 24	x	x	x	x	x	x	x	x	x	x	x	x	1	0	0	1	M0:e.24
Logical Rack 1Device Inhibit Word 25	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	1	M0:e.25
Logical Rack 2 Device Inhibit Word 26	x	x	x	x	x	x	x	x	x	x	x	x	0	0	1	0	M0:e.26
Logical Rack 3 Device Inhibit Word 27	x	x	x	x	x	x	x	x	x	x	x	x	0	0	1	0	M0:e.27

e = slot number of the SLC rack containing the scanner x = not used/defined

Device Reset and Remote Output Reset Considerations From This Mode

The 1747-BSN Scanner Device Reset words (M0:e.16 to M0:e.19) and the Remote Output Reset words (M0:e.24 to M0:e.27) operate in conjunction with each RIO device to determine the state of that RIO device's outputs. The output control information that the scanner sends to the RIO device depends on how you configure these bits. The RIO device acts on the output control information in accordance with its functionality and configuration. To fully understand how a specific device responds to the Device Reset and Remote Output Reset words, you must determine the operation of the RIO device. To determine RIO device output operation, refer to that device's user manual.

ATTENTION

When using the Device Reset and Remote Output Reset words, you must completely understand and fully test all device output operations before beginning normal system operation.

To properly use the Device Reset and Remote Output Reset words, you must consider the output control information sent to the devices during two SLC processor operating conditions:

- The SLC processor is in any given mode (Run, Program, Test, or Fault).
- The SLC processor is leaving any mode and entering another.

If you do *not* modify the Device Reset and Remote Output Reset words, the device outputs reflect the scanner output image whenever the SLC processor is in Run mode. If the SLC processor is in Program, Test, or Fault mode, it instructs the device to reset its outputs.

If you modify the default settings, the Device Reset and Remote Output Reset words change. The table on page 5-22 contains examples of what changes occur. *The information in the table is based on the assumption that the scanner's slot is always enabled and the RIO link device is communicating with the scanner.*

To determine how the Device Reset and Remote Output Reset words operate, locate the box where the row and column are headed by the modes in question. The shaded boxes represent the Device Reset and Remote Output Reset word operation while in that mode.

Example 1

When powering up into Run mode, the scanner, by default, resets the appropriate bit in the Device Reset word to 0. The appropriate bit in the Remote Output Reset word is set to 1. As a result, the RIO link device outputs reflect the scanner's output image.

Example 2

Once the SLC processor is in Run mode, the bits in the Remote Output Reset word have no effect on the RIO link device's outputs. Setting the appropriate bits in the Device Reset Word to 1 instructs the RIO link device to reset its outputs.

Example 3

When going from Run to Program mode, if both of the appropriate bits in the Device Reset and Remote Output Reset words are reset to 0 before leaving Run mode, the RIO link device is instructed to decide whether to hold its last output state or to reset its outputs.

From this mode	To this mode		
	Run	**Test**	**Program**
Power-up	DR = 0 ROR = 1 Default values are set automatically. Outputs reflect those of the canner output image.		DR = X ROR = X
Run	ROR = X DR = 0 Outputs are unchanged. DR = 1 Outputs are turned off.	DR = 0 In this instance, the last ROR = 0 state switch setting is valid.	DR = 0 In this instance, ROR = 0 state switch setting is valid.
		DR = X ROR = 1 These two combinations reset device outputs. DR = 1 ROR = X	DR = X ROR = 1 These two combinations reset device outputs. DR = 1 ROR = X
Test	DR = 0 ROR = 1 Default values are set automatically. Outputs reflect those of the scanner output image.	DR = 0 In this instance, the last ROR = 0 state switch setting is valid. Once these outputs are reset, they remain reset regardless of the DR and ROR settings. DR = X ROR = 1 These two combinations reset device outputs. SR = 1 ROR = X	Outputs remain unchanged.
Program	DR = 0 ROR = 1 Default values are set automatically. Outputs reflect those of the scanner output image.	DR = 0 ROR = 1 These default values are set automatically. Outputs are reset, unless ROR is changed to 0 on the first scan.	DR = X ROR = X

M1 Status File Description

M1 file words 0 through 47 contain the status of all devices on the scanner's RIO link. *M1 is a read only file; do not write to this file.*

Words 0 through 47 of the M1 file provide the following information:

- Word 0 (M1:e.0) - general communication status (overall device fault and communications attempted)
- Word 2 (M1:e.2) - RIO baud rate status
- Word 3 (M1:e.3) - complementary device starting address status
- Word 4 (M1:e.4) - complementary logical image size status
- Word 5 (M1:e.5) - complementary active device status
- Word 8 (M1:e.8) - primary/normal device starting address status
- Word 9 (M1:e.9) - primary/normal logical image size status
- Word 10 (M1:e.10) - active device status
- Words 12-15 (M1:e.12 -15) - device fault status
- Words 16-31 (M1:e.16-31) - primary/normal device retry counters
- Words 32-47 (M1:e.32-47) - complementary device retry counters

General Communication Status - Enable Device Fault Bit

Word 0, bit 0 is the Enabled Device Fault status bit. When any enabled device is faulted, this bit is set to 1. A fault may be caused by a communication problem with a remote device.

M1 (Status) File Word 0

Bit Number (decimal)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M1 File
General Communication Status Word, Word 0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	1	1	M1:e.0

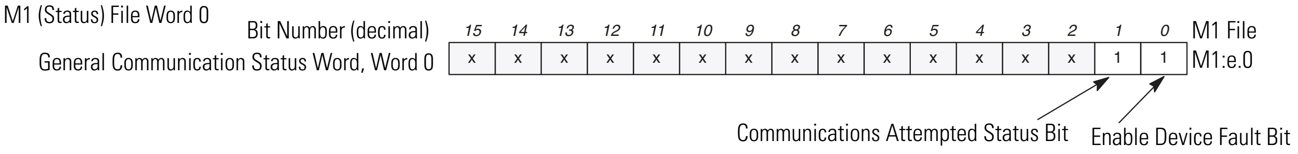
Enabled Device Fault Bit

General Communication Status - Communication Attempted Bit

Word 0, bit 1 is the Communications Attempted status bit. When RIO communication has been attempted with *all configured devices*, this bit is set to 1. There are no further transitions of this bit until a processor change of state occurs (i.e., Program mode to Run mode or Test mode, or Test mode to Run mode).

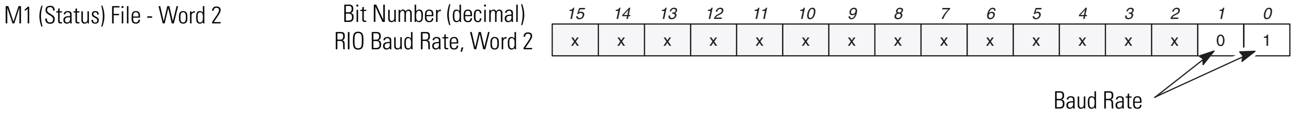
Until this bit is set, all devices in M1 file word 10 (active device status) appear to be inactive. This bit can be used to condition the Enabled

Device Fault bit. If the Communications Attempted bit is 1, the Enabled Device Fault bit is valid.



RIO Baud Rate Status

Word 2, bits 0 to 1 display the RIO communication baud rate you have set the scanner to via its DIP switch. Writing to word 2 does not change the scanner baud rate.

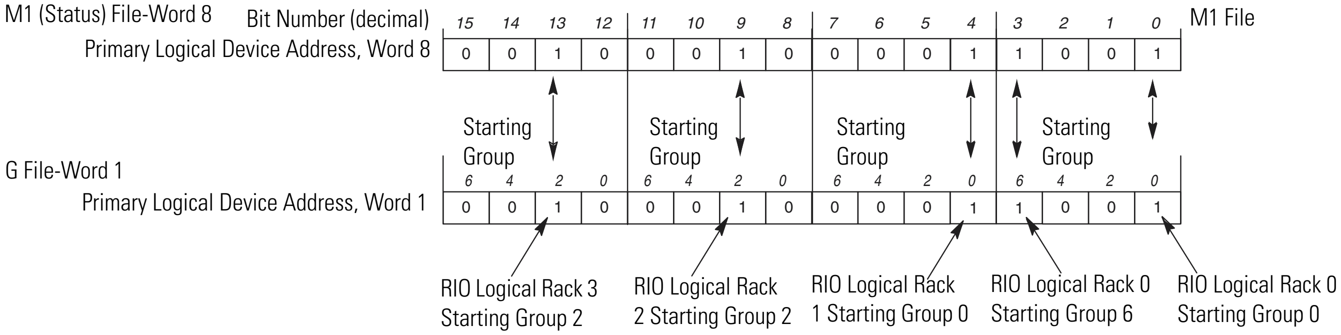


As illustrated by the table below, bit 0 = SW1 and bit 1 = SW2.

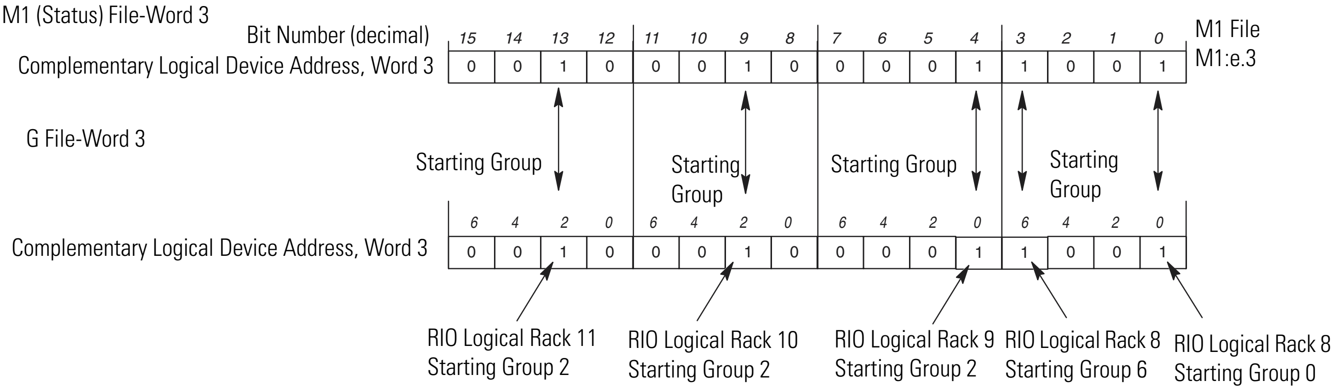
Bit 1-0	Baud Rate	SW 1-2
11	57.6K Baud	11
01	115.2K Baud	10
10	230.4K Baud	01
00	Disabled	00

Logical Device Starting Address Status

Word 8 provides status/feedback of the logical device starting addresses you configured in word 1 of the G file (primary/normal logical devices). Writing to M1 file word 8 *does not* alter the contents of the G file.

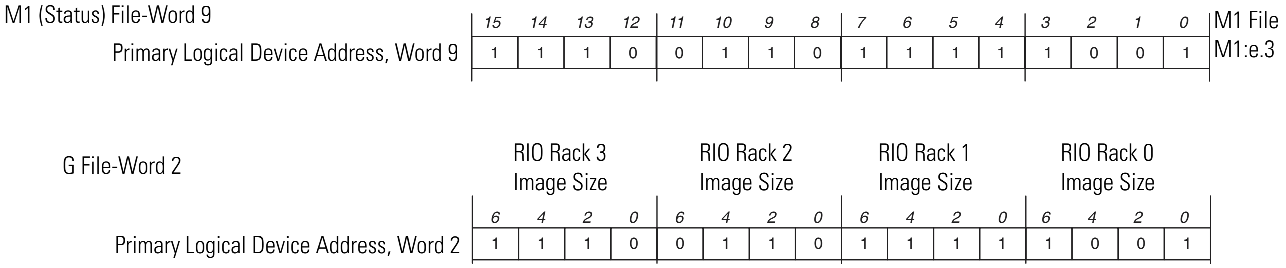


Word 3 provides status/feedback of the logical device starting addresses you configured in word 3 of the G file (complementary devices). Writing to M1 file word 3 does *not* alter the contents of the G file.

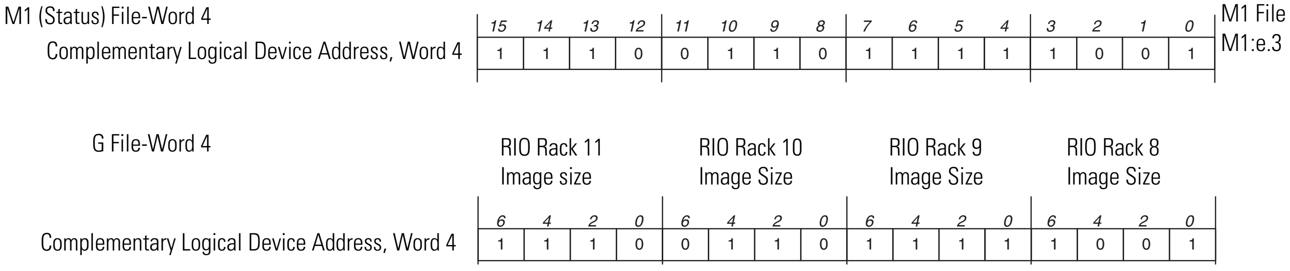


Logical Device Image Size Status

Word 9 provides status/feedback of the logical device image size you configure in word 2 of the G file (primary/normal devices). A bit set to 1 shows the logical image size of each logical device. Writing to word M1 file word 9 does *not* alter the contents of the G file.



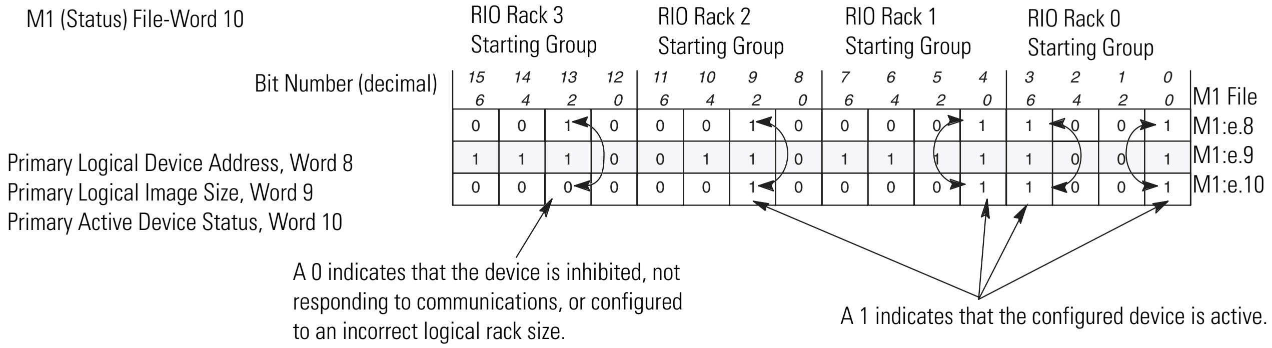
Word 4 provides status/feedback of the logical device image size you configure in word 4 of the G file (complementary devices). A bit set to 1 shows the logical image size of each logical device. Writing to word M1 file word 4 does *not* alter the contents of the G file.



Active Device Status

Word 10 provides active device status for primary/normal devices. When a RIO device is communicating with the scanner the bit corresponding to the device's logical starting group is set to 1.

Devices that are inhibited in the M0 file (M0:e.8 - M0:e.11) are represented by a 0. Unless devices are inhibited, not responding to communications, or configured to an incorrect logical rack size, this word is identical to the device configuration (M1:e.8).

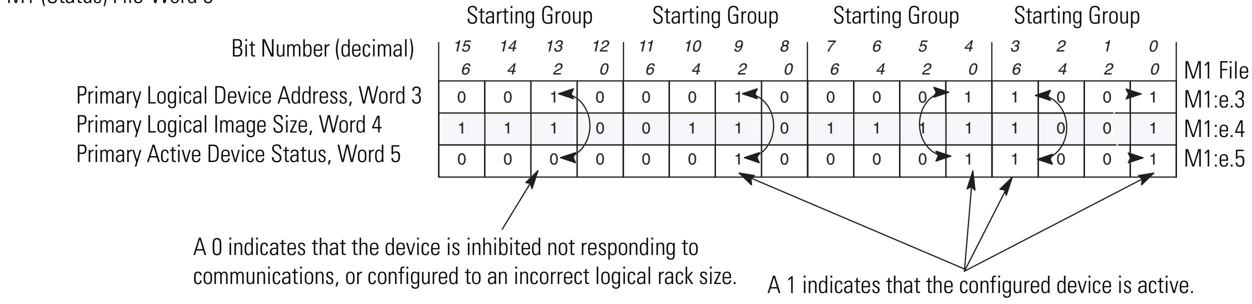


Bit Number (decimal)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
M1 File	6	4	2	0	6	4	2	0	6	4	2	0	6	4	2	0
M1:e.8	0	0	1	0	0	0	1	0	0	0	0	1	1	0	0	1
M1:e.9	1	1	1	0	0	1	1	0	1	1	1	1	1	0	0	1
M1:e.10	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	1

Word 5 provides active device status for complementary devices. When a RIO device is communicating with the scanner the bit corresponding to the device's logical starting group is set to 1.

Devices that are inhibited in the M0 file are represented by a 0. Unless devices are inhibited, not responding to communications, or configured to an incorrect logical rack size, this word is identical to the device configuration (M1:e.3).

IMPORTANT When a primary device is inhibited, its complementary device is also inhibited. A complementary device cannot be exclusively inhibited.

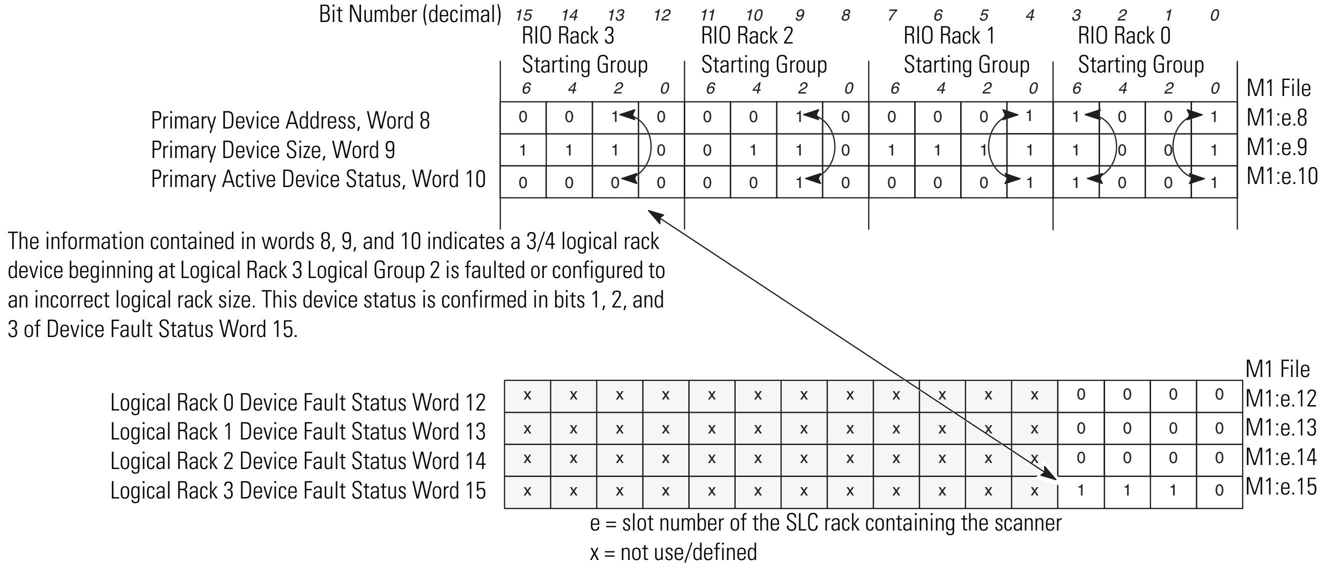


Bit Number (decimal)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
M1 File	6	4	2	0	6	4	2	0	6	4	2	0	6	4	2	0
M1:e.3	0	0	1	0	0	0	1	0	0	0	0	1	1	0	0	1
M1:e.4	1	1	1	0	0	1	1	0	1	1	1	1	1	0	0	1
M1:e.5	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	1

Logical Device Fault Status

Words 12 through 15, bits 0 to 7 indicate the device fault status for logical racks 0, 1, 2, 3, 8, 9, 10, and 11. Bits 0 through 3 are for primary/normal devices and bits 4 through 7 are for complementary devices. Each bit corresponds to a quarter logical rack location. If a device is not responding to communications, has gone offline, or is configured to an incorrect logical rack size, all bits corresponding to the device are set to 1. This is highlighted in the example below.

M1 (Status) File Primary/Normal Device Fault Status



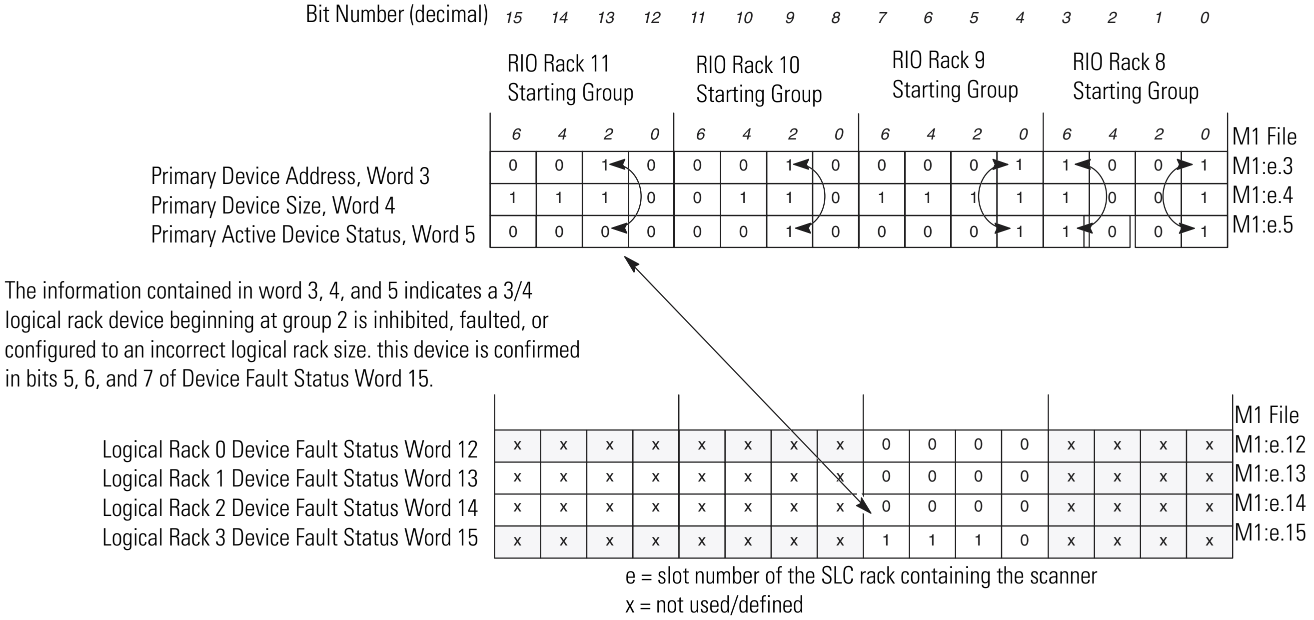
Bit Number (decimal)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	RIO Rack 3 Starting Group				RIO Rack 2 Starting Group				RIO Rack 1 Starting Group				RIO Rack 0 Starting Group				
	6	4	2	0	6	4	2	0	6	4	2	0	6	4	2	0	M1 File
Primary Device Address, Word 8	0	0	1	0	0	0	1	0	0	0	0	1	1	0	0	1	M1:e.8
Primary Device Size, Word 9	1	1	1	0	0	1	1	0	1	1	1	1	1	0	0	1	M1:e.9
Primary Active Device Status, Word 10	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	1	M1:e.10

The information contained in words 8, 9, and 10 indicates a 3/4 logical rack device beginning at Logical Rack 3 Logical Group 2 is faulted or configured to an incorrect logical rack size. This device status is confirmed in bits 1, 2, and 3 of Device Fault Status Word 15.

																	M1 File
Logical Rack 0 Device Fault Status Word 12	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	M1:e.12
Logical Rack 1 Device Fault Status Word 13	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	M1:e.13
Logical Rack 2 Device Fault Status Word 14	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	M1:e.14
Logical Rack 3 Device Fault Status Word 15	x	x	x	x	x	x	x	x	x	x	x	x	1	1	1	0	M1:e.15

e = slot number of the SLC rack containing the scanner
x = not use/defined

M1 (Status) File Complementary Device Fault Status



Bit Number (decimal)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	RIO Rack 11 Starting Group				RIO Rack 10 Starting Group				RIO Rack 9 Starting Group				RIO Rack 8 Starting Group				
	6	4	2	0	6	4	2	0	6	4	2	0	6	4	2	0	M1 File
Primary Device Address, Word 3	0	0	1	0	0	0	1	0	0	0	0	1	1	0	0	1	M1:e.3
Primary Device Size, Word 4	1	1	1	0	0	1	1	0	1	1	1	1	1	0	0	1	M1:e.4
Primary Active Device Status, Word 5	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	1	M1:e.5

The information contained in word 3, 4, and 5 indicates a 3/4 logical rack device beginning at group 2 is inhibited, faulted, or configured to an incorrect logical rack size. this device is confirmed in bits 5, 6, and 7 of Device Fault Status Word 15.

																	M1 File
Logical Rack 0 Device Fault Status Word 12	x	x	x	x	x	x	x	x	0	0	0	0	x	x	x	x	M1:e.12
Logical Rack 1 Device Fault Status Word 13	x	x	x	x	x	x	x	x	0	0	0	0	x	x	x	x	M1:e.13
Logical Rack 2 Device Fault Status Word 14	x	x	x	x	x	x	x	x	0	0	0	0	x	x	x	x	M1:e.14
Logical Rack 3 Device Fault Status Word 15	x	x	x	x	x	x	x	x	1	1	1	0	x	x	x	x	M1:e.15

e = slot number of the SLC rack containing the scanner
x = not used/defined

RIO Status Example

The following example illustrates an M1 status file example. It shows a typical M1 file and the G file used to configure the scanner. There are no inhibited devices specified in the M0 file (not shown). Notice that:

- M1:e.8 is an image of word 1 (primary/normal logical device address) of the G file.
- M1:e.3 is an image of word 3 (complementary logical device address) of the G file.
- M1:e.9 is an image/copy of word 2 (primary/normal logical device size) of the G file.
- M1:e.4 is an image/copy of word 4 (complementary logical device size) of the G file.
- The three quarter logical rack device located in logical rack 3 (M1:e.9/13) is not active. The fault is indicated by the Enabled Device Fault status bit, bit 0, word 0 (M1:e.0/0).
- The three quarter logical rack device located in logical rack 11 (M1:e.4/13) is not active. The fault is indicated by the Enabled Device Fault status bit, bit 0, word 0 (M1:e.0/0).

Because the device at M1:e.8/13 is faulted, bit 13 of word 10 (M1:e.10/13) is 0. M1:e.15/1 through M1:e.15/3, which correspond to M1:e.9/13 through M1:e.9/15 are also set to 1, indicating a problem with the device in logical rack 3.

Because the device at M1:e.3/13 is faulted, bit 13 of word 5 (M1:e.5/13) is 0. M1:e.15/5 through M1:e.15/7, which correspond to M1:e.4/13 through M1:e.4/15 are also set to 1, indicating a problem with the device in logical rack 11.

M1 (Status) File Primary/Normal

Bit Number (decimal)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M1 File
Status Word, Word 0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	1	1	M1:e.0
Baud Rate, Word 2	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0	1	M1:e.2
	RIO Logical Rack 3				RIO Logical Rack 2				RIO Logical Rack 1				RIO Logical Rack 0				
Primary Device Address, Word 8	0	0	1	0	0	0	1	0	0	0	0	1	1	0	0	1	M1:e.3
Primary Device Size, Word 9	1	1	1	0	0	1	1	0	1	1	1	1	1	0	0	1	M1:e.4
Primary Active Device Status, Word 10	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	1	M1:e.5
Logical Rack 0 Device Fault Status Word 12	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	M1:e.12
Logical Rack 1 Device Fault Status Word 13	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	M1:e.13
Logical Rack 2 Device Fault Status Word 14	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	M1:e.14
Logical Rack 3 Device Fault Status Word 15	x	x	x	x	x	x	x	x	x	x	x	x	1	1	1	0	M1:e.15

e = slot number of the SLC rack containing the scanner x = not used/defined

G File

	RIO Logical Rack 3 Starting Group				RIO Logical Rack 2 Starting Group				RIO Logical Rack 1 Starting Group				RIO Logical Rack 0 Starting Group			
	6	4	2	0	6	4	2	0	6	4	2	0	6	4	2	0
Primary Logical Device Address, Word 1	0	0	1	0	0	0	1	0	0	0	0	1	1	0	0	1
Primary Logical Image Size, Word 2	1	1	1	0	0	1	1	0	1	1	1	1	1	0	0	1

M1 (Status) File Complementary

Bit Number (decimal)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M1 File
Status Word, Word 0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	1	1	M1:e.0
Baud Rate, Word 2	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0	1	M1:e.2
	RIO Logical Rack 11				RIO Logical Rack 10				RIO Logical Rack 9				RIO Logical Rack 8				
Primary Device Address, Word 3	0	0	1	0	0	0	1	0	0	0	0	1	1	0	0	1	M1:e.3
Primary Device Size, Word 4	1	1	1	0	0	1	1	0	1	1	1	1	1	0	0	1	M1:e.4
Primary Active Device Status, Word 5	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	1	M1:e.5
Logical Rack 0 Device Fault Status Word 12	x	x	x	x	x	x	x	x	0	0	0	0	x	x	x	x	M1:e.12
Logical Rack 1 Device Fault Status Word 13	x	x	x	x	x	x	x	x	0	0	0	0	x	x	x	x	M1:e.13
Logical Rack 2 Device Fault Status Word 14	x	x	x	x	x	x	x	x	0	0	0	0	x	x	x	x	M1:e.14
Logical Rack 3 Device Fault Status Word 15	x	x	x	x	x	x	x	x	1	1	1	0	x	x	x	x	M1:e.15

e = slot number of the SLC rack containing the scanner x = not used/defined

G File

	RIO Logical Rack 11 Starting Group				RIO Logical Rack 10 Starting Group				RIO Logical Rack 9 Starting Group				RIO Logical Rack 8 Starting Group			
	6	4	2	0	6	4	2	0	6	4	2	0	6	4	2	0
Complementary Logical Device Address, Word 3	0	0	1	0	0	0	1	0	0	0	0	1	1	0	0	1
Complementary Logical Image Size, Word 4	1	1	1	0	0	1	1	0	1	1	1	1	1	0	0	1

IMPORTANT Individual quarter logical racks within a device cannot be faulted. Therefore, only the starting logical group of the device needs to be monitored.

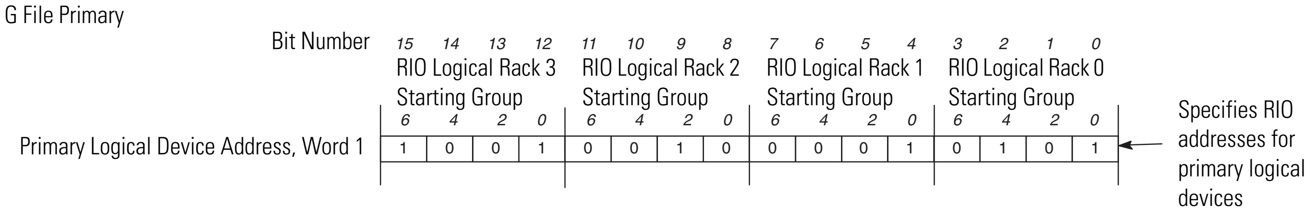
RIO Communication Retry Counter (M1:e.16 - 47)

M1 File Status Words 16 through 47 indicate how many RIO communication retries the scanner makes to each adapter on the RIO link if communication problems occur. Each word (16 to 47) contains a retry counter for each configured quarter logical rack (words 16 to 31 are for primary logical racks, 0 to 3, and 32 to 47 are for complementary racks, 8 to 11). Retry counters are useful for troubleshooting communication problems (such as electrical noise or poor communication line connections) between the scanner and any adapters. The scanner clears the retry counters when going from Program to Run mode, Test to Run mode, and Program to Test mode. Note that the display (in words M1:e.16 to 31) of retry counters corresponds to the bits set in the *Primary Logical Device Address* - Word 1 of the G file. Likewise, the display (in words M1:e.32 to 47) correspond to the bits set in the *Complementary Logical Device Address* – Word 3 of the G file.

IMPORTANT Your SLC control program cannot initialize/clear retry counters.

Retry Counter Example for Primary Devices

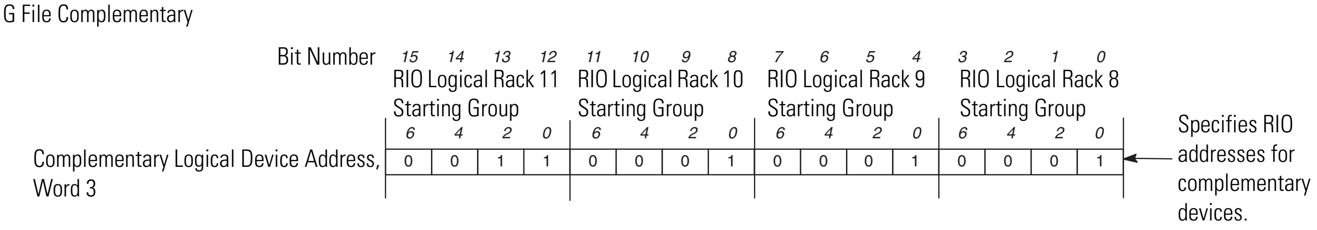
The scanner's I/O image tables are configured as shown with M1 status files displaying the corresponding retry counters:



G File Primary

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RIO Logical Rack 3 Starting Group				RIO Logical Rack 2 Starting Group				RIO Logical Rack 1 Starting Group				RIO Logical Rack 0 Starting Group			
	6	4	2	0	6	4	2	0	6	4	2	0	6	4	2	0
Primary Logical Device Address, Word 1	1	0	0	1	0	0	1	0	0	0	0	1	0	1	0	1

Specifies RIO addresses for primary logical devices



G File Complementary

Bit Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RIO Logical Rack 11 Starting Group				RIO Logical Rack 10 Starting Group				RIO Logical Rack 9 Starting Group				RIO Logical Rack 8 Starting Group			
	6	4	2	0	6	4	2	0	6	4	2	0	6	4	2	0
Complementary Logical Device Address, Word 3	0	0	1	1	0	0	0	1	0	0	0	1	0	0	0	1

Specifies RIO addresses for complementary devices.

M1:e.16 - communication retry counter for RIO logical rack 0, group 0
M1:e.17 - not used in this example
M1:e.18 - communication retry counter for RIO logical rack 0, group 4
M1:e.19 - not used in this example
M1:e.20 - communication retry counter for RIO logical rack 1, group 0
M1:e.21 - not used in this example
M1:e.22 - not used in this example
M1:e.23 - not used in this example
M1:e.24 - not used in this example
M1:e.25 - communication retry counter for RIO logical rack 2, group 2
M1:e.26 - not used in this example
M1:e.27 - not used in this example
M1:e.28 - communication retry counter for RIO logical rack 3, group 0
M1:e.29 - not used in this example
M1:e.30 - not used in this example
M1:e.31 - communication retry counter for RIO logical rack 3, group 6
M1:e.32 - communication retry counter for RIO logical rack 8, group 0
M1:e.33 - not used in this example
M1:e.34 - not used in this example
M1:e.35 - not used in this example
M1:e.36 - communication retry counter for RIO logical rack 9, group 0
M1:e.37 - not used in this example
M1:e.38 - not used in this example
M1:e.39 - not used in this example
M1:e.40 - communication retry counter for RIO logical rack 10, group 0
M1:e.41 - not used in this example
M1:e.42 - not used in this example
M1:e.43 - not used in this example
M1:e.44 - communication retry counter for RIO logical rack 11, group 0
M1:e.45 - communication retry counter for RIO logical rack 11, group 2
M1:e.46 - not used in this example
M1:e.47 - not used in this example

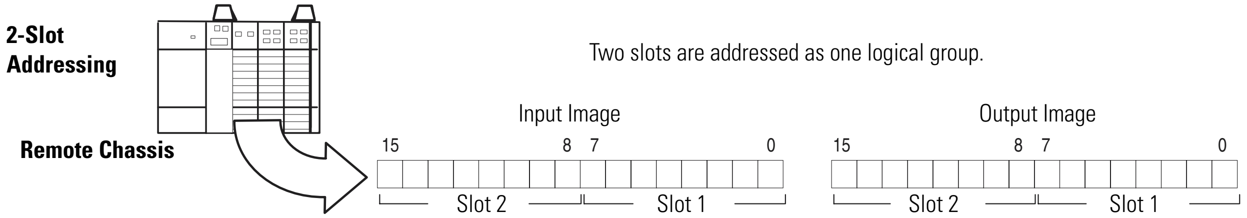
Understanding Slot Addressing

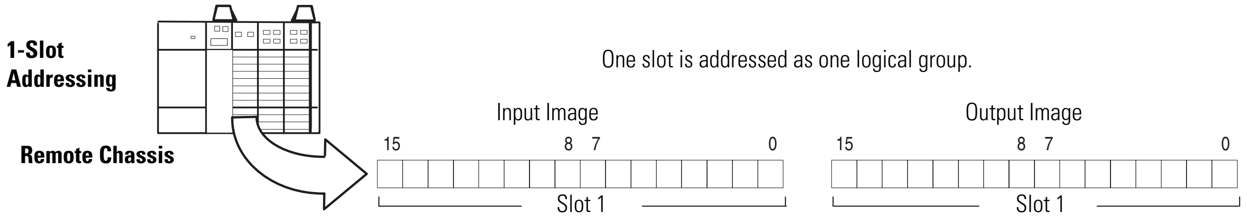
This section provides information about:

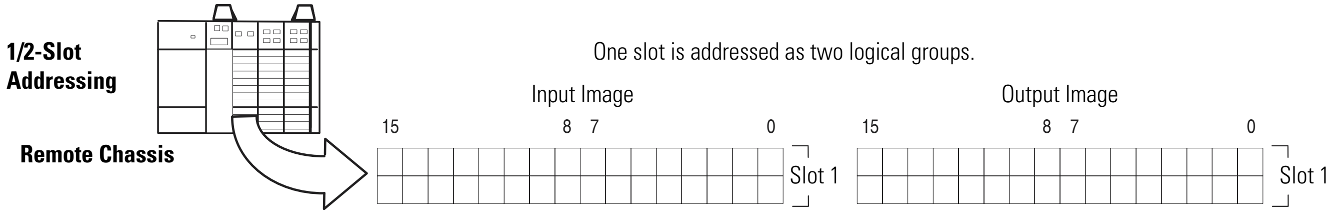
- 2-slot addressing
- 1-slot addressing
- 1/2-slot addressing

Understanding slot addressing is critical to most efficiently allocate your scanner's I/O image files.

Slot addressing refers to how each remote chassis slot is assigned a specific amount of the I/O image. The amount depends on which type of slot addressing you choose at your adapter; 2-slot, 1-slot, and 1/2-slot addressing is available, as shown below:


Two slots are addressed as one logical group.


One slot is addressed as one logical group.


One slot is addressed as two logical groups.

For more information on slot addressing, refer to your ASB module user manual.

Note that slot addressing (e.g., 1/2-, 1-, and 2-slot) may not apply to all types of RIO devices. Refer to each RIO device's user manual to determine the type of slot addressing required.

SLC/Scanner Configuration Using HHT

Your SLC 5/02 processor can be programmed with an HHT(1) (Hand-Held Terminal). Although the configuration steps are similar, they are not identical. Therefore, the following basic steps are provided. For specific instructions, refer to the user manual included with your programming device. For more information on M and G files, refer to Appendix B.

1. Locate an open slot in your SLC chassis. Remember that you must use an SLC 5/02 or later processor.

2. Assign the scanner to a physical slot in the SLC processor's chassis by selecting 1747-BSN from the list. If the scanner selection is not available, select OTHER from the I/O Configuration Screen and enter the Module ID number: 13609.

3. Enter the number of Scanned Input and Output Words using the Specialty I/O and Advanced Setup menus.

   The default value is 32 I/O words. You can specify less than 32 and reduce the processor scan time by transferring only the part of the input and output image that your application requires.

   IMPORTANT Do not set either of these values to 0. If you do, the scanner will not work correctly.

4. Using the Specialty I/O Configuration menu, set the M1 and M0 file sizes to 32 words (48 words if using complementary I/O). 32 words is the minimum required for operation. If you do not set the M1 and M0 file sizes to at least 32 words, the programming device does not allow you to access the M files in the SLC control program. The maximum M file sizes are 5548 words for complete capability. There is no penalty for setting the M file lengths to this maximum.

5. Set the G file size to 3 (5 if using complementary I/O) using the Specialty I/O Configuration menu.

6. Enter your setup information using the Modify G File menu.

   IMPORTANT SLC 5/02 processors scan chassis I/O slots left to right starting at slot 1, regardless of the module type. SLC 5/03 and later processors scan slots with discrete I/O modules first, left to right starting at slot 1, and then slots with specialty modules, left to right starting at slot 1.

---

(1) The SLC 5/03, SLC 5/04 and SLC 5/05 processors cannot be programmed with the HHT.

Module Control and Status Word

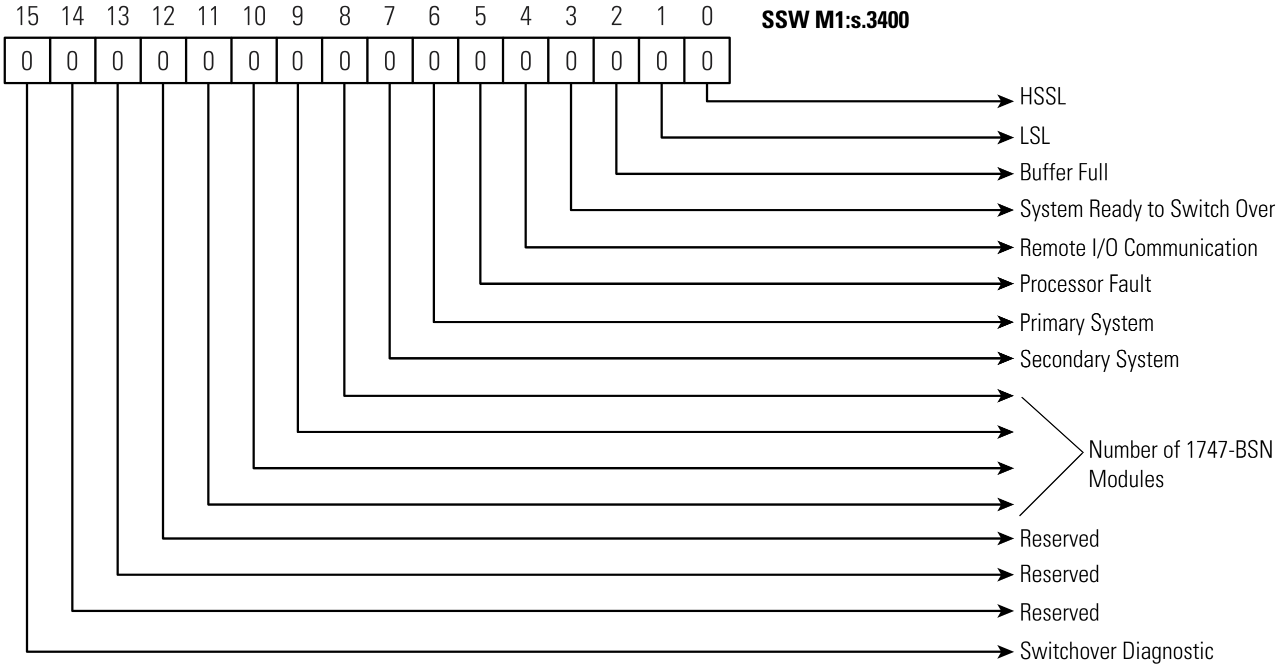
The addresses M0:s.400 through M0:s.3499 and M1:s.3400 through M1:s.3499 are used for status and control exchange between the SLC 5/0x and the 1747-BSN. This item contains the initial definition for these words. The table below shows the address allocation for these words.

Control Word Description	**M0 Address**	
	Start	**End**
Reserved	3400	3409
Data Transfer Control Word - DTCW	3410	3410
Data Transfer Handshake Word - DTHW	3411	3411
Reserved	3412	3499

Status Word Description	**M1 Address**	
	Start	**End**
System Status Word - SSW	3400	3400
Module Status Word - MSW	3401	3401
Switch Assemblies Status Word - SASW	3402	3402
Reserved	3403	3409
Data Transfer Status Word - DTSW	3410	3410
Reserved	3411	3419
Module Status Counters	3420	3426
Reserved	3427	3429
Data Block Counters	3430	3445
Reserved	3446	3499

System Status Word

The System Status Word (SSW) presents a synthesis of the backup system status. The primary objective of this word is to provide a fast way to see if the system is working well. The status words described in the following sections provide additional details about the 1747-BSN module and system. The SSW status bits have the following meaning:



- **HSSL High-Speed Serial Link status:** This bit is set when the HSSL is faulted or when it has communication errors in one or more 1747-BSN module in the backup system. The normal state of this bit is OFF.
- **LSL Local Status Link status:** This bit is set when the Local Status Link is not working well in either the local or the remote system. If the system has only one 1747-BSN module in each chassis, the LSL is not be used and is always reset. The normal state of this bit is OFF.
- **Buffer Full:** This bit is set when the secondary 1747-BSN is receiving a second block of data. This occurs before the reading of the first block by the secondary SLC 5/0x. The normal state of this bit is OFF.
- **System Ready to Switchover:** This bit is set when the backup system is ready to switchover. It is reset when there is one or more errors occurring in the backup system and the system cannot switchover. The normal state of this bit is ON.
- **Remote I/O Link:** This bit is set when one or more of the RIO links in the system is faulted or has communication errors. The normal state is OFF.

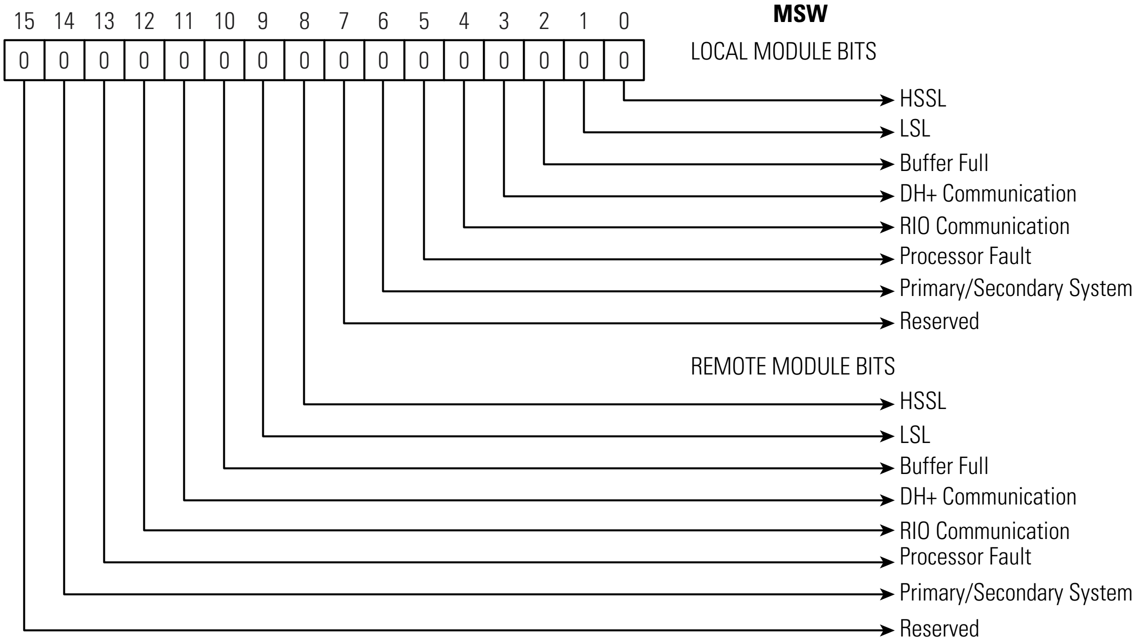
- **Processor Fault:** This bit is set when either the local or remote processor is in failure, in Program mode, or in Test mode. The normal state of this bit is OFF.
- **Primary System:** This bit is set when the local system is in primary state.
- **Secondary System:** This bit is set when the local system is in the secondary state. If the primary bit is set, the secondary system bit is not set. The reverse is also true. When the secondary bit is set, the primary bit is not set. At powerup and during switchover, these primary and secondary bits could be reset at the same time.
- **Number of 1747-BSN Modules:** This bit indicates the number of 1747-BSN modules in the system.

System Status Word Bits				Number of 1747-BSN's
11	10	9	8	
0	0	0	0	1
0	0	0	1	2
0	0	1	0	3
0	0	1	1	4
0	1	0	0	5
0	1	0	1	6
0	1	1	0	7
0	1	1	1	8

- **Switchover Diagnostic:** This bit is set when a 1747-BSN module tests the functionality of the Remote I/O detector circuit when the processor changes form Run to Program mode. If the test fails, this bit is set until the next power down. In this case the module must be replaced.

Module Status Word

The Module Status Word (MSW) shows the status of the 1747-BSN module itself and its counterpart in the remote system. Bits 0 through 7 show the local status while bits 8 through 15 show the remote status. The module status bits have the following meaning:



- **HSSL High-Speed Serial Link status:** This bit is set when the HSSL is faulted or has communication errors in the local 1747-BSN module (bit 0) or in the remote one (bit 8). The normal state of this bit is OFF.
- **LSL Local Status Link status:** This bit is set when the Local Status Link is not working well in the local system (bit 1) or remote system (bit 9). If the system has only one 1747-BSN module in each chassis, then the LSL is not used and this bit is reset. The normal state of this bit is OFF.
- **Buffer Full:** This bit is set when the secondary 1747-BSN is receiving a second block of data before the reading of the first block by the secondary SLC 5/0x. It could be occurring in the local system (bit 2) or remote system (bit 10).
- **DH+ Communication:** This bit is set when a problem occurs with the DH+ network in the local (bit 3) or remote (bit 11) 1747-BSN module. The type of problem that may occur is different for primary and secondary systems (as shown below). The normal state of this bit is OFF):
  1. Primary mode: The network is down or the channel is seeing communication errors.
  2. Secondary mode: Communication errors to the secondary SLC 5/04.

- **RIO Communication:** This bit is set when a problem occurs with the RIO link in the local (bit 4) or remote (bit 12) 1747-BSN module. The type of problem that may occur is different for primary and secondary systems (as shown below). The normal state of this bit is OFF):
    1. Primary mode: The channel is faulted or has communication errors.
    2. Secondary: No communication could be seen by this module in the RIO link. This indicates that the primary RIO link is not working or that a problem exists in the RIO connection to the secondary 1747-BSN module.
- **Processor Fault:** This bit is set when one of the processors (SLC 5/0x), either the local (bit 5) or remote (bit 13), is in failure or is in Program or Test mode. The normal state of this bit is OFF.
- **Primary/Secondary System:** This bit is set when the local (bit 6) or remote (bit 14) system is in the primary mode, otherwise the system is in the secondary mode. Bits 6 and 14 are never set at the same time. During power-up or switchover time, these bits can be clear at the same time. The normal operation is with one bit set for the primary system and one bit clear for the secondary system.

Transferring Data over the High-Speed Serial Link (HSSL)

Two M0 file words and one M1 file word are used to control the transfer of data from the primary SLC processor to the secondary SLC processor. This link should be used to transfer internal processor data that must be kept current in the secondary processor in the event that a switchover takes place, making the secondary processor the primary processor. Actual input data from remote I/O chassis need not be transferred over this link since the secondary 1747-BSN allows the secondary processor to receive this data from the remote I/O network in a “listen mode”. Outputs are only controlled by the primary processor.

The three words used to accomplish data transfer over the HSSL are:

1. M0:s.3410: Data Transfer Control Word (DTCW)

   This word is used by the primary processor to initiate the transfer of data to the primary 1747-BSN. This data is ultimately transferred to the secondary 1747-BSN over the HSSL, then to the secondary processor.

2. M1:s.3410: Data Transfer Status Word (DTSW)

   This word is used in both the primary system and the secondary system to affect the transfer of data on the HSSL.

   - Primary System - The word must be monitored by the primary processor because this word is used by the primary 1747-BSN to acknowledge that the 1747-BSN has received the data block from the processor and is ready for the next transfer for that particular data block number.
   - Secondary System - The word is used by the 1747-BSN to let the processor know that a new data block is ready for transfer from the secondary 1747-BSN to the secondary processor. The ladder program in the secondary processor must monitor the bits in this word to know when to copy new HSSL data from the 1747-BSN to the processor data table.

3. M0:s.3411: Data Transfer Handshake Word (DTHW)

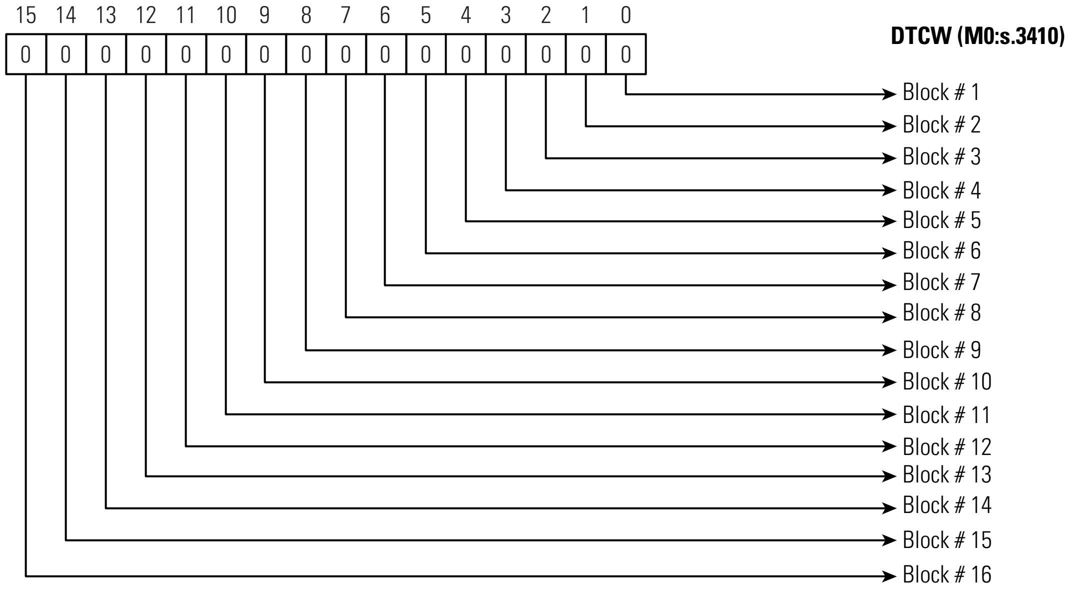
   This word is used by the secondary processor to inform the secondary 1747-BSN that it has received the latest data block and is ready for the next transfer for that particular data block number.

Data Transfer Control Word

The Data Transfer Control Word (DTCW) is used to control the data table transfer from primary to the secondary system. Each bit corresponds to a block in the 1747-BSN module.

If the primary SLC 5/0x writes a new block into the primary 1747-BSN, then one bit corresponding to this block must be set in the DTCW word to advise the backup module that a new valid block is ready to be transferred to the secondary SLC 5/0x. The figure below shows the meaning of the bits in this word.

Data Table Control Word



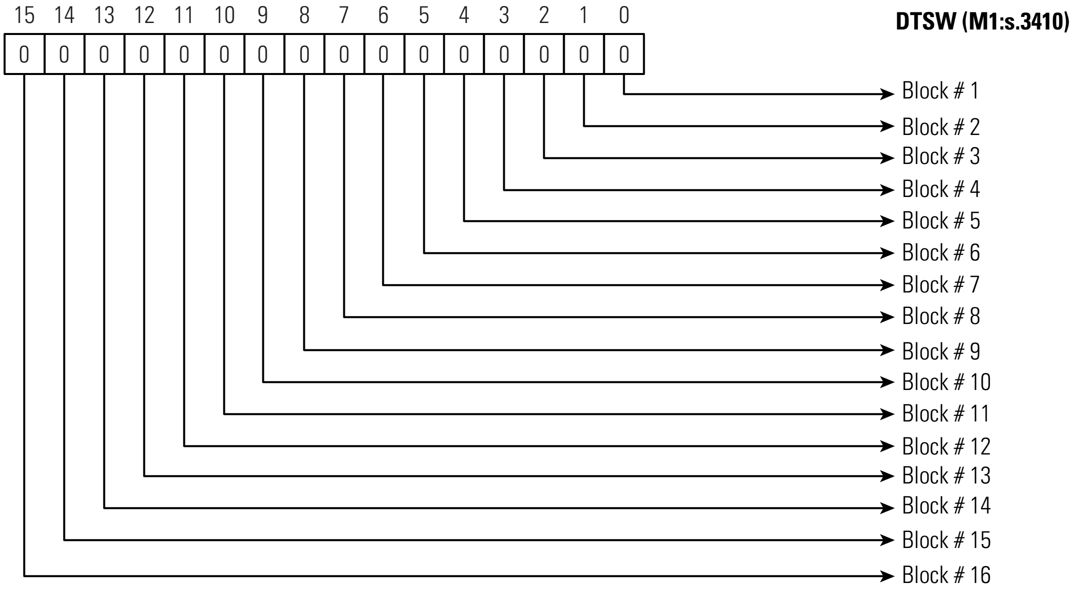
Data Transfer Status Word

The Data Transfer Status Word (DTSW) is used in the data table transfer from primary to the secondary system. Each bit corresponds to a block in the 1747-BSN module.

When the primary SLC 5/0x writes a new block into the primary 1747-BSN and sets the bit corresponding to this block in DTCW, the 1747-BSN module transfers the block to the secondary backup module. The 1747-BSN also sets the corresponding bit in DTSW to advise the SLC 5/0x that the block has been transferred and the block area is free to transfer a new data block. This bit is held set until the corresponding bit in DTCW is cleared by the primary processors.

When the secondary 1747-BSN receives a new data block from the primary system, it writes the block into the M1 area to be read by the secondary SLC 5/0x and sets its corresponding bit in DTSW to advise that this new block is ready to read. After reading the block, the SLC 5/0x must set the corresponding bit in DTHW. Then the 1747-BSN module clears the bit in DTSW. The figure below shows the meaning of the bits in this word.

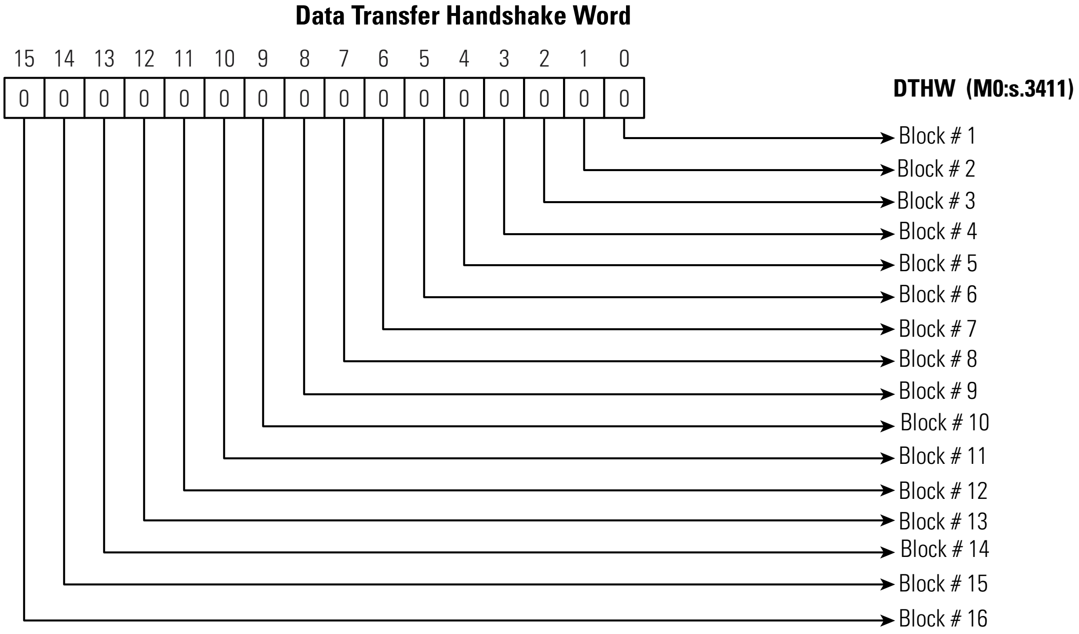
Data Transfer Status Word



Data Transfer Handshake Word

The Data Transfer Handshake Word (DTHW) is used in the data table transfer from primary to the secondary system. Each bit corresponds to a block in the 1747-BSN module.

When the secondary 1747-BSN receives a new block from the primary system, it writes the block to the M1 area to be read by the secondary SLC 5/0x. The secondary 1747-BSN also sets its corresponding bit in DTSW to advise that this new block is ready to read. After reading the block, the SLC 5/0x must set the corresponding handshake bit in DTHW. Then the 1747-BSN module clears the bit in DTSW.



The following ladder program example shows how to monitor the status bits in the System Status Word (SSW) and in the Module Status Word (MSW) that indicate which processor is in the primary mode and which is in the secondary mode. This information is needed in order to determine whether the local processor needs to execute the primary system or the secondary system HSSL data transfer ladder logic.

File 2 monitors the appropriate SSW and MSW bits and executes subroutine 4 if the local system is in primary mode.

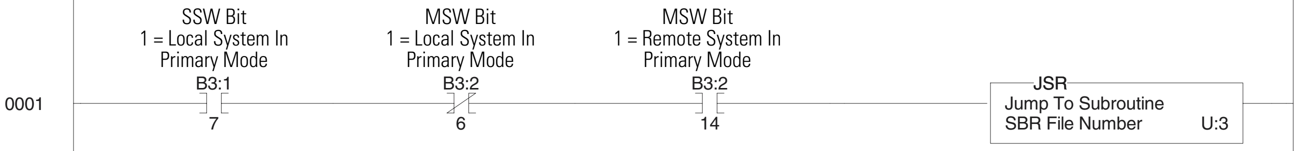
TIP

Set the M1 and M0 file lengths for the 1747-BSN to the maximum (5548). There is no memory or speed penalty for creating the maximum buffer size for these files in the 1747-BSN modules.

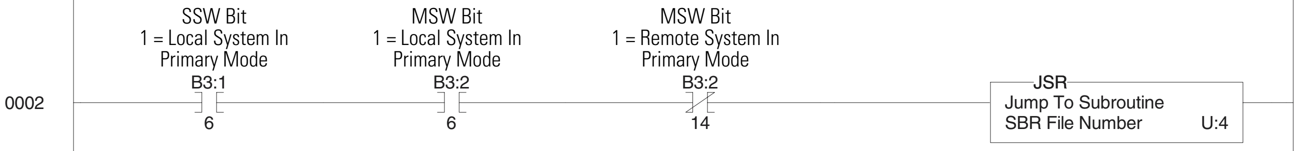
Program File 2

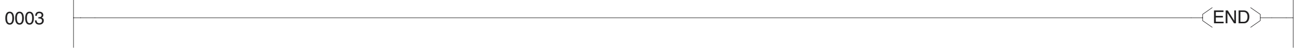
Copy the System Status Word (SSW) and the Module Status Word (MSW) from the 1747-BSN module to internal storage words in the SLC processor every program scan. Bits 6 and 7 of the SSW indicate whether this processor is primary of secondary at any given time. Bits 6 and 14 of the MSW indicate whether the local or remote system is in primary or secondary mode, respectively. Virtual SSW ans MSW words/bits are used to minimize M-file transfers, thereby minimizing their effects on the program scan.

This rung executes subroutine number 3 when the local system is in the secondary mode. Subroutine 3 contains logic which must be executed to properly transfer HSSL data from the secondary 1747-BSN to the secondary SLC processor.

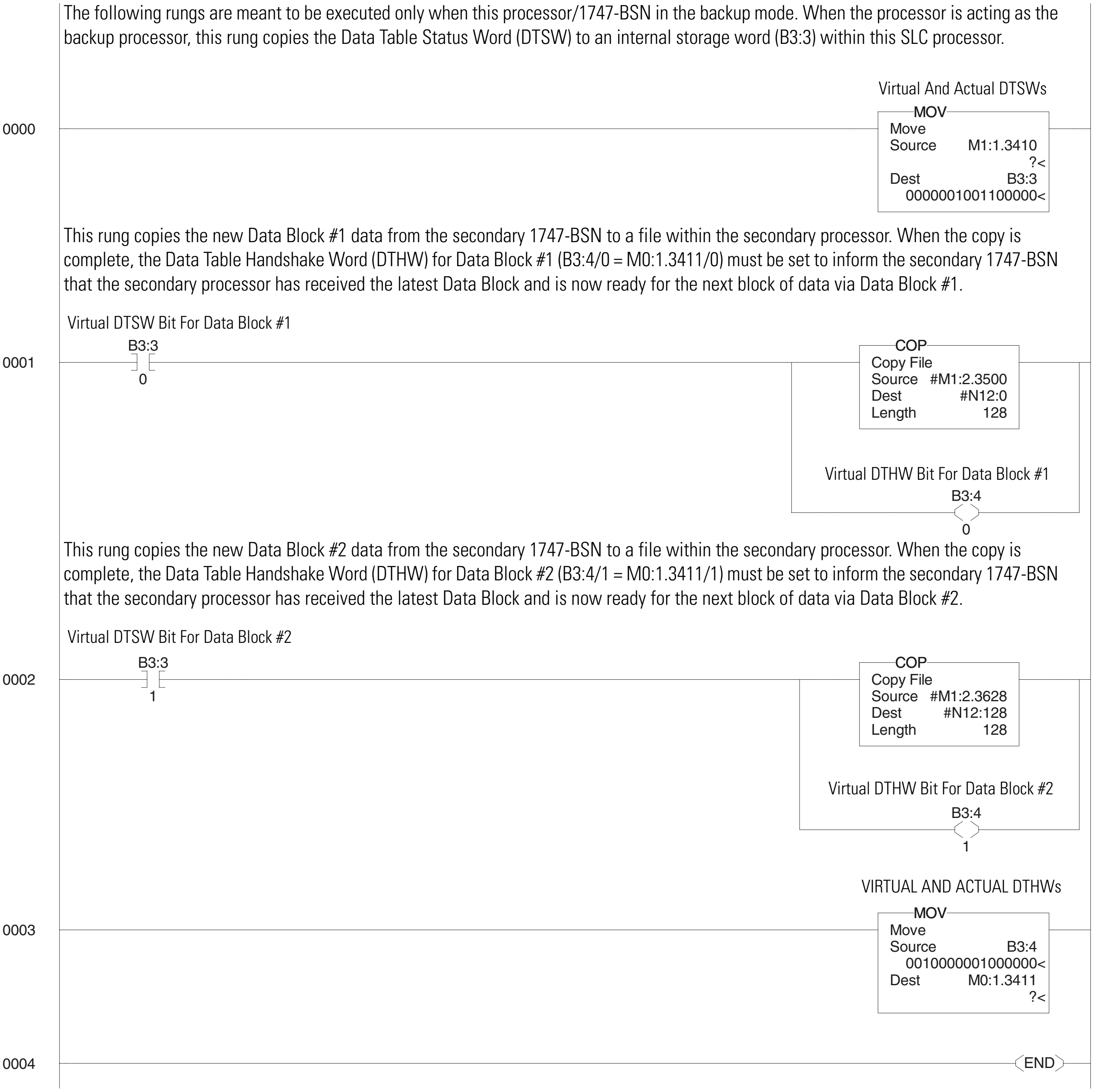


This rung executes subroutine number 4 when the local system is in the primary mode. Subroutine 4 contains logic which must be executed to properly transfer HSSL data from the primary 1747-BSN to the primary SLC processor.





Program File 3

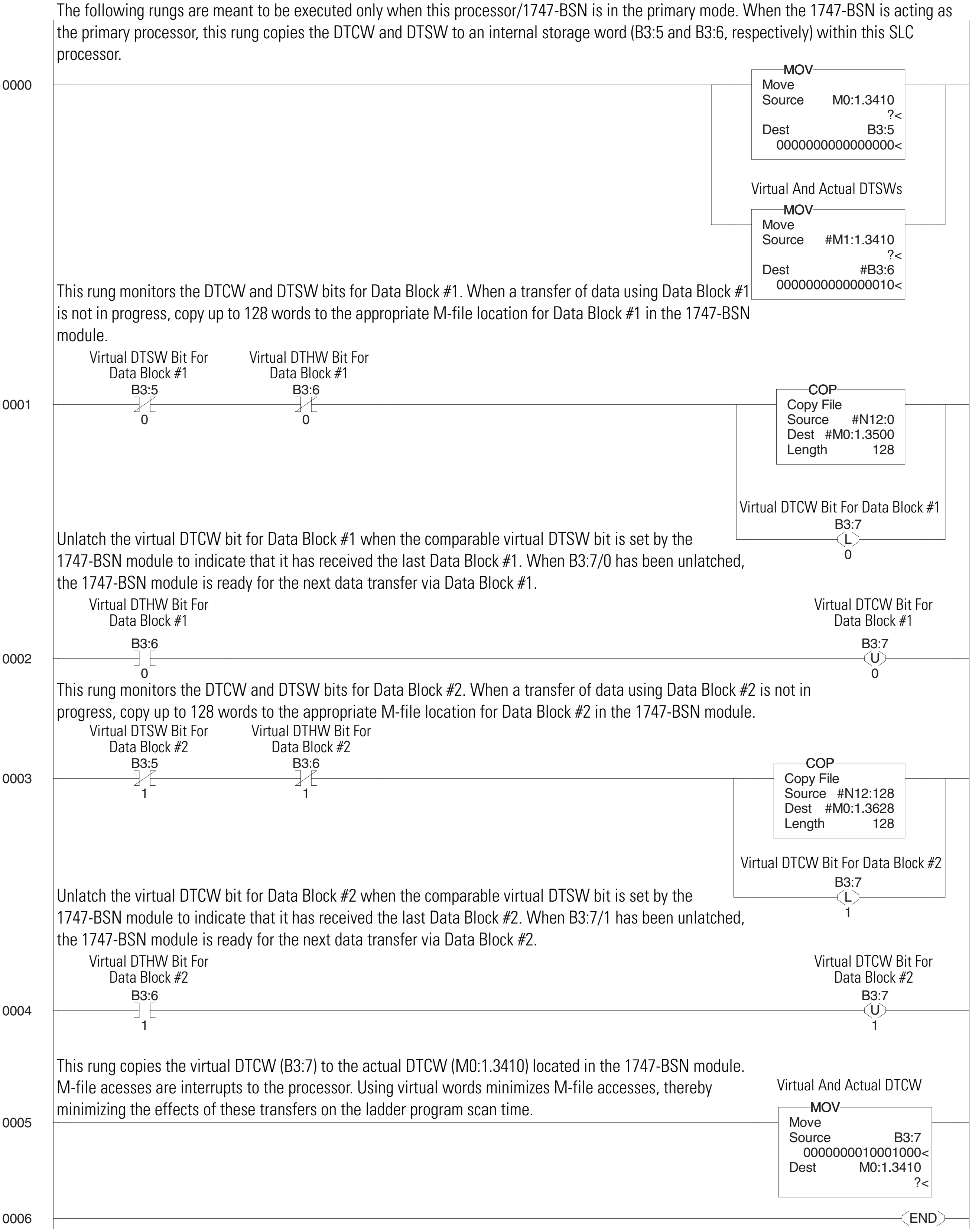


The following rungs are meant to be executed only when this processor/1747-BSN in the backup mode. When the processor is acting as the backup processor, this rung copies the Data Table Status Word (DTSW) to an internal storage word (B3:3) within this SLC processor.

This rung copies the new Data Block #1 data from the secondary 1747-BSN to a file within the secondary processor. When the copy is complete, the Data Table Handshake Word (DTHW) for Data Block #1 (B3:4/0 = M0:1.3411/0) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #1.

This rung copies the new Data Block #2 data from the secondary 1747-BSN to a file within the secondary processor. When the copy is complete, the Data Table Handshake Word (DTHW) for Data Block #2 (B3:4/1 = M0:1.3411/1) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #2.

Program File 4



The following rungs are meant to be executed only when this processor/1747-BSN is in the primary mode. When the 1747-BSN is acting as the primary processor, this rung copies the DTCW and DTSW to an internal storage word (B3:5 and B3:6, respectively) within this SLC processor.

This rung monitors the DTCW and DTSW bits for Data Block #1. When a transfer of data using Data Block #1 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #1 in the 1747-BSN module.

Unlatch the virtual DTCW bit for Data Block #1 when the comparable virtual DTSW bit is set by the 1747-BSN module to indicate that it has received the last Data Block #1. When B3:7/0 has been unlatched, the 1747-BSN module is ready for the next data transfer via Data Block #1.

This rung monitors the DTCW and DTSW bits for Data Block #2. When a transfer of data using Data Block #2 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #2 in the 1747-BSN module.

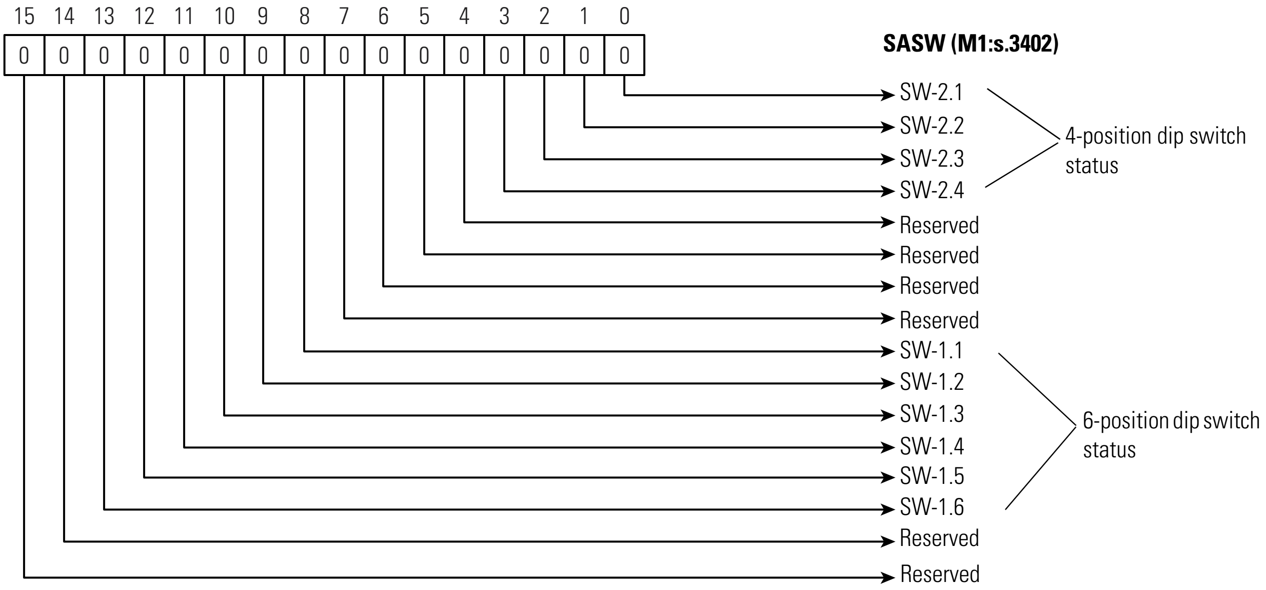
Unlatch the virtual DTCW bit for Data Block #2 when the comparable virtual DTSW bit is set by the 1747-BSN module to indicate that it has received the last Data Block #2. When B3:7/1 has been unlatched, the 1747-BSN module is ready for the next data transfer via Data Block #2.

This rung copies the virtual DTCW (B3:7) to the actual DTCW (M0:1.3410) located in the 1747-BSN module. M-file acesses are interrupts to the processor. Using virtual words minimizes M-file accesses, thereby minimizing the effects of these transfers on the ladder program scan time.

Switch Assemblies Status Word

The Switch Assemblies Status Word (SASW) shows the switch assemblies configuration status for both the six-position (SW1) and the four-position (SW2) dip switch. Bits 0 through 3 show the four-position dip switch status, while bits 8 through 13 show the six-position dip switch status.

Switch Assemblies Status Word



The six-position dip switch is used to configure the communication channel and other miscellaneous information. The switches have the following definition:

- SW-1.1 and SW-1.2: Define the communication channel baud rate as shown below.

SW-1.1	SW-1.2	Baud Rate (KBaud)
1	1	57.6
1	0	115.2
0	1	230.4
0	0	Disabled

- SW-1.3: When ON means that the channel is configured as DH+. When OFF means that the channel is configured as RIO.
- SW-1.4: User identification SW. This switch can be used to differentiate one 1747-BSN from its counterpart and identify when a switchover occurs. This switch is user-configurable and does not affect the operation of the module.
- SW-1.5: Reserved

- SW-1.6: Last Module in LSL. The 1747-BSN module that the user intends to be the last in the Local Status Link must have this switch ON. All the other 1747-BSN modules must have this switch OFF. If a backup system has only one 1747-BSN in each processor chassis, this switch must be ON in this module.

The four-position dip switch is used to configure the 1747-BSN address in the Local Status Link. The table below shows the meaning of bits 0 through 2 of SASW. Switch SW-2-4 is not used.

Switch Assemblies Status Word Bits				1747-BSN Address
SW-2.4	**SW-2.3**	**SW-2.2**	**SW-2.1**	
Not Used	0	0	0	1
	0	0	1	2
	0	1	0	3
	0	1	1	4
	1	0	0	5
	1	0	1	6
	1	1	0	7
	1	1	1	8

Module Status Counters

The1747-BSN module counts the changes in the local module's significant status and indicates the changes in its Module Status Word. The counters show all the errors and status changes since the last power-up and cannot be cleared by the user without powering down the chassis.

Address	Counter
M1:s.3420	HSSL Error Counter
M1:s.3421	LSL Error Counter
M1:s.3422	Buffer Full Counter
M1:s.3423	Reserved
M1:s.3424	RIO Comm. Error Counter
M1:s.3425	Processor Fault Counter
M1:s.3426	Switchover Counter

Data Block Counters

To help verify the functionality of the data transfer application program, the 1747-BSN module has a counter for each data block that can be transferred.

Each time that a block is transferred from the primary SLC 5/0x to the primary 1747-BSN, the data block counter that corresponds to this block is incremented once in the primary 1747-BSN. When the secondary SLC 5/0x reads a block from the secondary 1747-BSN, the corresponding counter is incremented once in the secondary 1747-BSN.

Address	Data Block Counter
M1:s.3430	Block # 1
M1:s.3431	Block # 2
M1:s.3432	Block # 3
M1:s.3433	Block # 4
M1:s.3434	Block # 5
M1:s.3435	Block # 6
M1:s.3436	Block # 7
M1:s.3437	Block # 8
M1:s.3438	Block # 9
M1:s.3439	Block # 10
M1:s.3440	Block # 11
M1:s.3441	Block # 12
M1:s.3442	Block # 13
M1:s.3443	Block # 14
M1:s.3444	Block # 15
M1:s.3445	Block # 16

RIO Block Transfer

This chapter contains the following information:

- RIO block transfer theory of operation
- RIO block transfer general functional overview
- using BTR/BTW instructions

RIO Block Transfer Theory of Operation

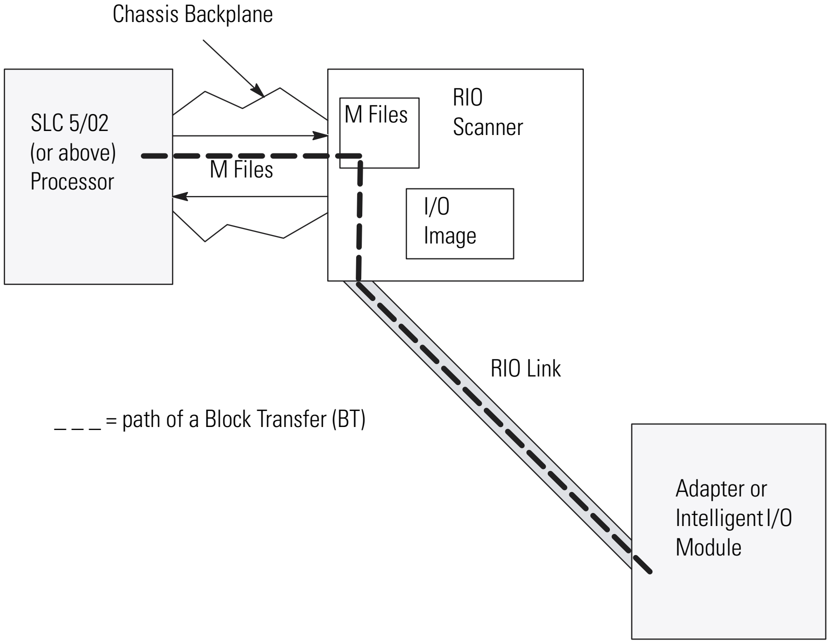
This section provides a conceptual overview of block transfer as it pertains to SLCs, RIO scanners, and remote devices. For specific functionality details, refer to the RIO Block Transfer General Functional Overview section on page 7-5.

What Is RIO Block Transfer?

RIO Block Transfer is a data transfer mechanism that allows you to control the transfer of up to 64 words of data to or from a remote device over the Allen-Bradley RIO link. A *Block Transfer Read* (BTR) is used when a remote device transfers data to the SLC. A *Block Transfer Write* (BTW) is used when an SLC processor writes data to a remote device.

The diagrams on the following pages illustrate the concepts of how block transfers occur using an SLC processor, an RIO scanner, and a remote device. The first diagram illustrates the path a block transfer follows. The second and third diagrams illustrate in greater detail the theory of operation of a BTR and a BTW, respectively.

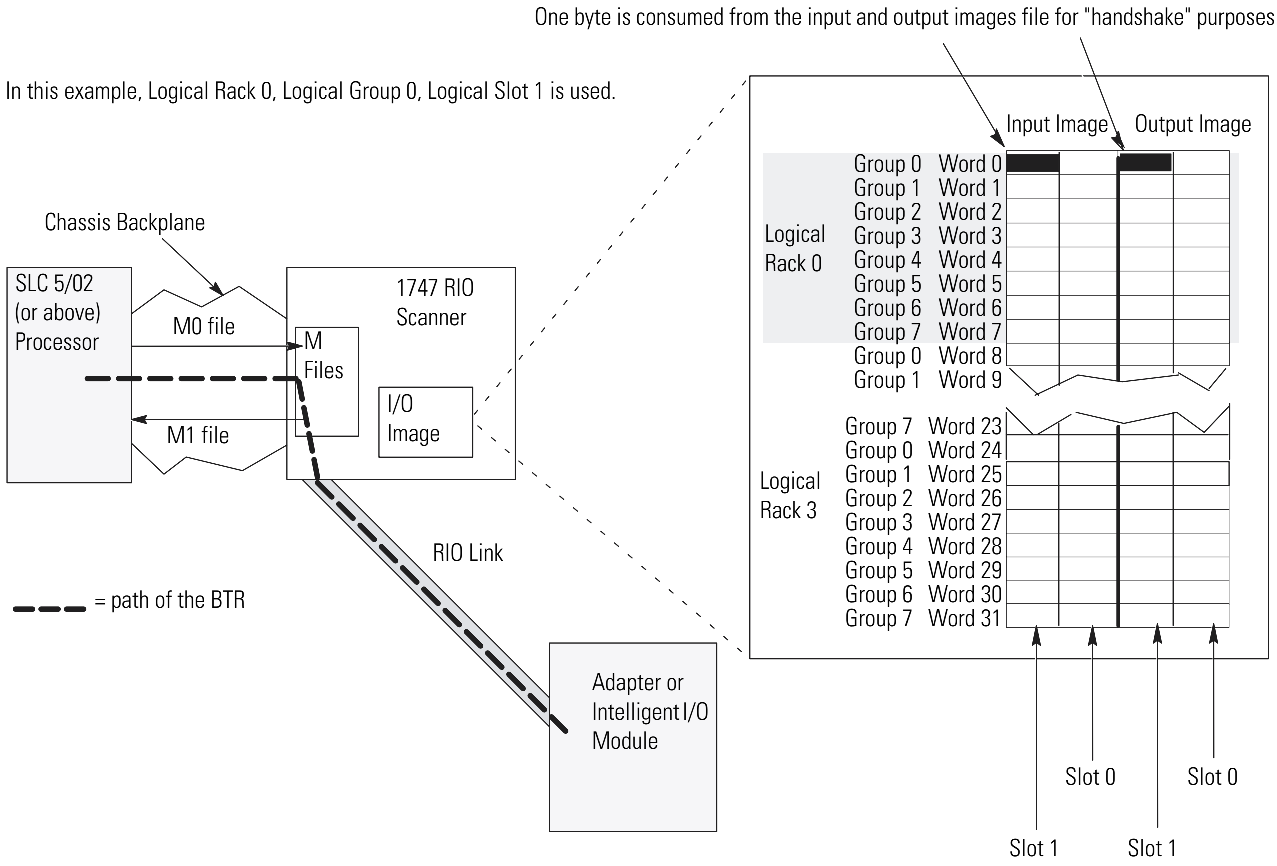
RIO Block Transfer Theory of Operation - Path of a Block Transfer



Block Transfer Write (BTW) data travels from the SLC processor across the chassis backplane via the scanner's M files. The scanner then sends the data across the RIO link to the adapter of intelligent I/O module.

Block Transfer Read (BTR) data travels from the adapter of intelligent I/O module over the RIO link to the scanner. The chassis backplane then transfers BTR data via the scanner's M files to the SLC processor. The SLC control program processes the data once the SLC receives it from the scanner.

RIO Block Transfer Theory of Operation-Block Transfer Read (BTR)



One byte is consumed from the input and output images file for "handshake" purposes.

In this example, Logical Rack 0, Logical Group 0, Logical Slot 1 is used.

The steps below detail a successful Block Transfer Read (BTR):

1. The M0 file contains BTR control information which controls (initiates) the scanner BTR operation. (Refer to the Block Transfer Buffer Layout section for details on control information.)

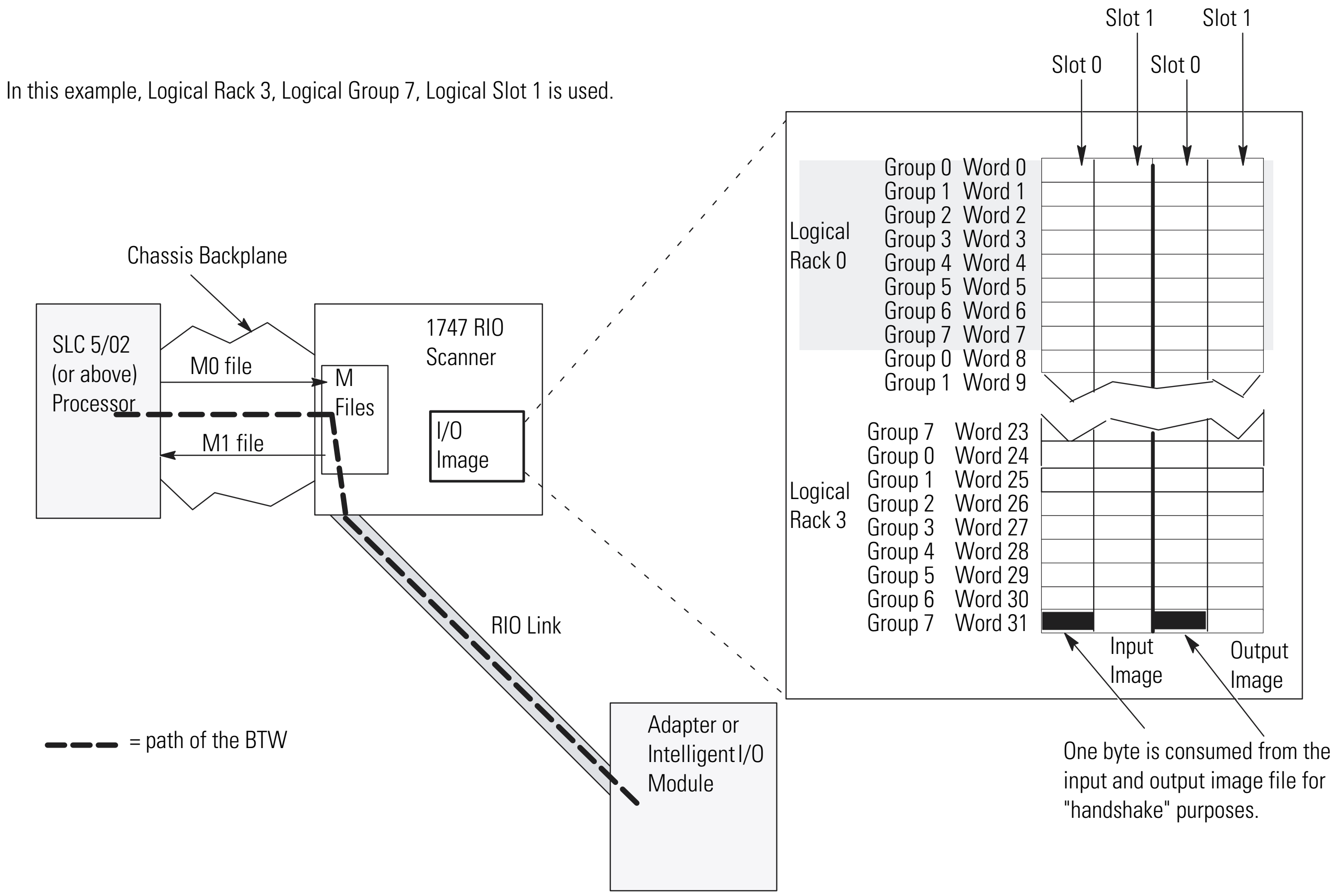
2. The SLC control program initiates a block transfer read by commanding the scanner to perform the read operation. The adapter/intelligent I/O module sends BTR data across the RIO link to the RIO scanner.

3. The scanner writes the BTR data to a unique M1 file location that you specify. Also, one byte of the scanner's I/O image file is used for "handshake" purposes between the scanner and the adapter/intelligent I/O module. Note that the SLC control program must never read or write to this "handshake" image space.

4. Using the M1 file and a COP instruction in the control program the scanner transfers the BTR data to the SLC processor via the chassis backplane. The M1 file also contains BTR status information. (Refer to the Block Transfer Buffer Layout section for details on status information.)

5. The SLC control program processes the BTR information.

RIO Block Transfer Theory of Operation-Block Transfer Write (BTW)



In this example, Logical Rack 3, Logical Group 7, Logical Slot 1 is used.

One byte is consumed from the input and output image file for "handshake" purposes.

The steps below detail a successful Block Transfer Write (BTW):

1. The user's control program executes a MOV or COP instruction to the MO file to initiate a BTW. The SLC processor sends BTW data (via the chassis backplane) to the scanner's MO block transfer control and write data file. (Refer to the Block Transfer Buffer Layout section for details on control information.

2. The scanner reads the BTW data and control data from the MO file. One byte of the scanner's I/O image file is used for handshake purposes. Note that the SLC user program must never read or write to this image space.

3. The M1 file contains BTW status information. Refer to the Block Transfer Buffer Layout section for details on status information.

4. The RIO scanner transfers BTW information across the RIO link to the adapter.

5. The adapter transfers the BTW information to the appropriate adapter or intelligent I/O module.

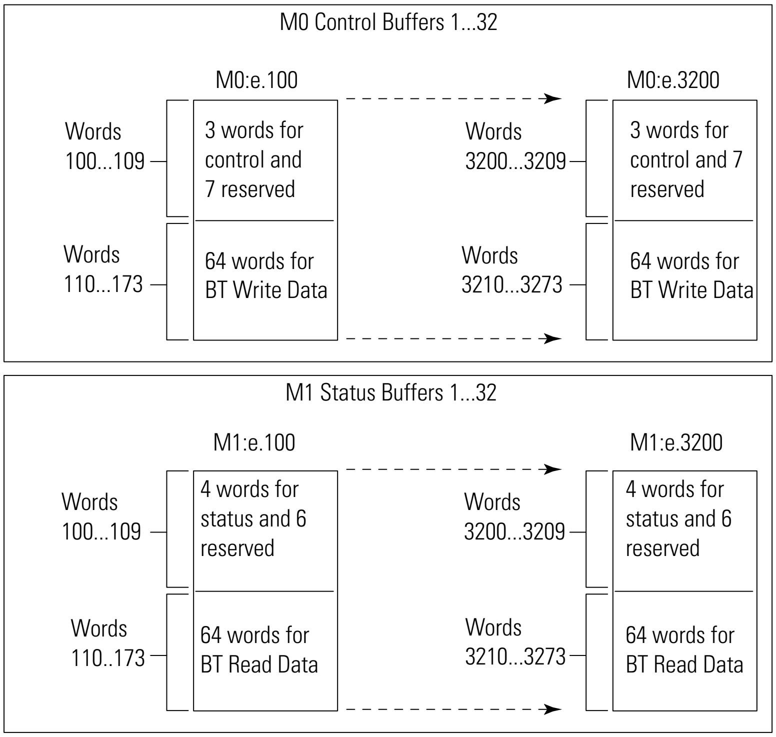
RIO Block Transfer General Functional Overview

The RIO scanner performs block transfers through control/status buffers that you allocate in the scanner's M0 and M1 files. For BTWs, the M0 BT buffer contains BTW control data and BTW data, while a corresponding M1 BT buffer contains only BTW status information. For BTRs, the M0 BT buffer contains only BTR control data, while a corresponding M1 BT buffer contains BTR status information and BTR data. Block transfers occur asynchronous to RIO link discrete transfers. Block transfers occur as RIO scan time allows – discrete I/O transfers have first priority.

A total of 32 block transfer control/status buffers exist in the M0 (output/control) and the M1 (input/status) files.

Block transfer buffers consist of:

- 3 BT control words in an M0 file BT buffer
- 4 BT status words in an M1 file BT buffer
- 64 words of BTW data in an M0 file and 64 words of BTR data in an M1 file. Refer to the diagrams on the following pages.



You use an M0 file BT control buffer to initiate a BT. The corresponding M1 file displays the status of the block transfer.

BT buffers reside on 100 word boundaries in the M0/M1 files starting at word 100. For example, BT buffer 1 resides at M0:e.100 and M1:e.100; BT buffer 2 resides at M0:e200 and M1:e.200; while BT buffer 16 resides at M0:e.1600 and M1:e.1600. Note that the "e" in these examples refers to the physical chassis slot number in which the scanner resides.

All block transfer buffers (M0 and M1) are cleared (set to zero) either when the RIO scanner goes through a power cycle or when the SLC processor commands the scanner to change mode from Program to Test mode, Program to Run mode, or Test to Run mode.

Scanner I/O Image Allocation For Block Transfer

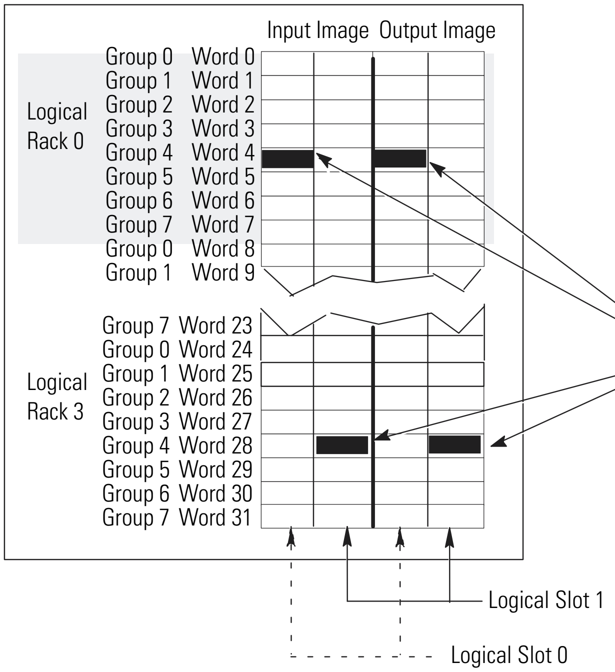
Block transfer operations (BTR and BTW) consume only one byte of the RIO scanner’s I/O image file, independent of what type of I/O slot addressing is used. This one byte image is reserved for communication “handshake” purposes between the remote device (adapter or intelligent I/O module) and the scanner. SLC control programs must never read/write to these image locations because unpredictable operations may result.

Block transfer operations (BTR and BTW) can be addressed to any logical slot within the RIO scanner’s four logical racks. See the examples on the following page.

Examples of BT I/O Image File Allocation

Example 1

1747-SN RIO Scanner's I/O Image Files



The minimum portion of the scanner's image that can be assigned to an adapter is 1/4-logical rack. Each logical device that you assign BT operations (BTR or BTW) consumes one byte from the scanner's input and output image file. The one byte can come from either the low" byte (Logical Slot 0) or the high" byte (Logical Slot 1). Logical Slot 1 only applies for 2 slot addressing.

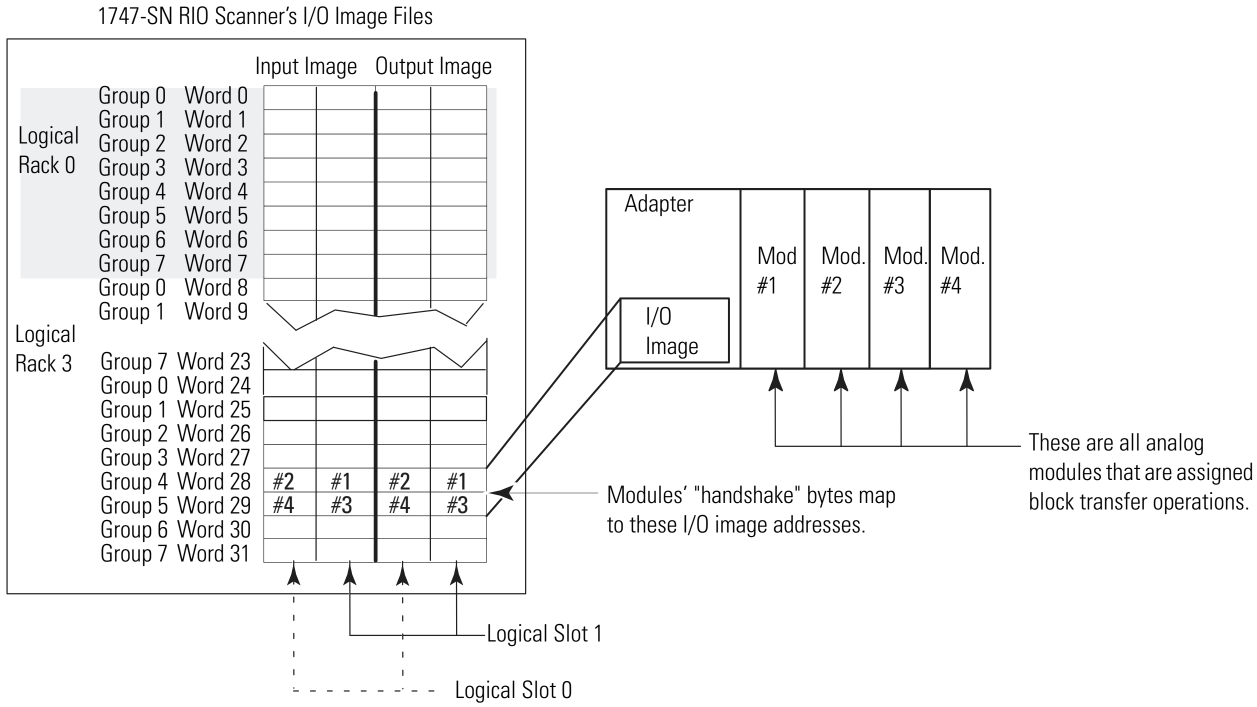
In this example there are two block transfer operations mapped to the scanner's I/O image. One BT operation is mapped to Logical Rack 0, Logical Group 4, Logical Slot 1. The other is mapped to Logical Rack 3, Logical Group 4, Logical Slot 0.

TIP

The logical address of your RIO devices (i.e., adapter and intelligent I/O modules) determine where the block transfer gets mapped.

Example 2

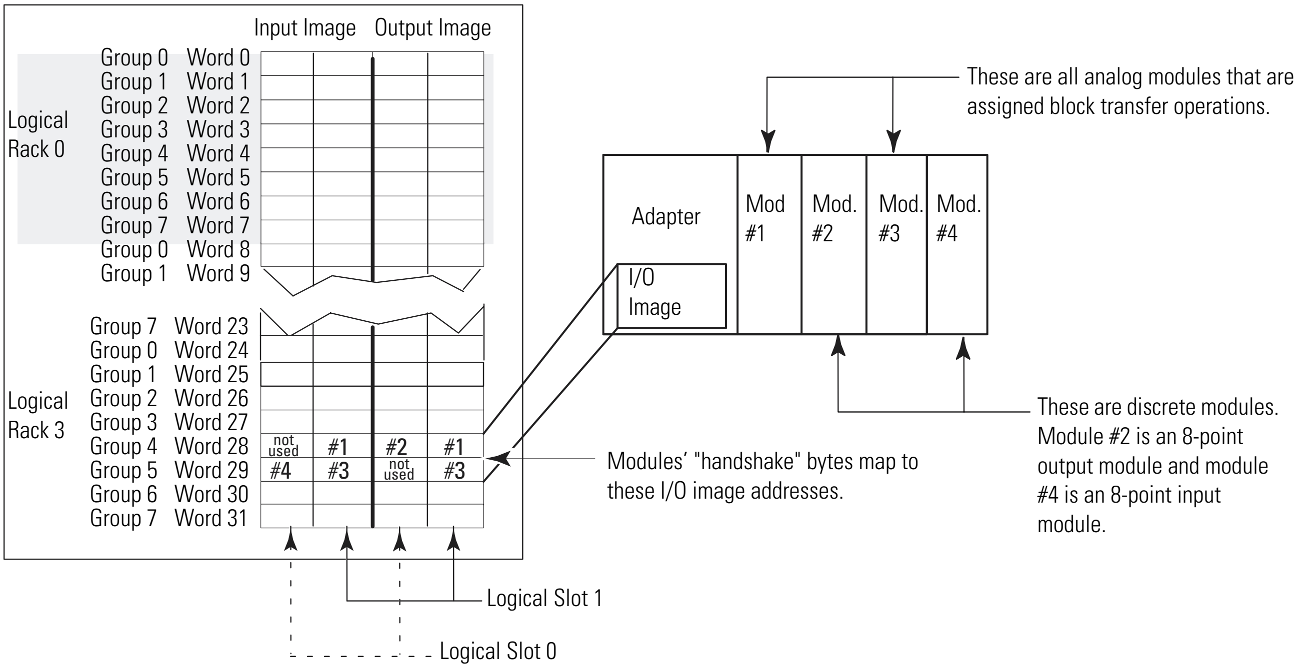
In this example, the remote adapter is configured for 2-slot addressing. It is assigned 1/4-logical rack or the scanner's I/O image files starting at RIO Logical Rack 3, Logical Group 4. The remote adapter controls four analog modules that are configured for block transfer operations. Each module uses both the input and output byte of the logical slot to which it is assigned.



Example 3

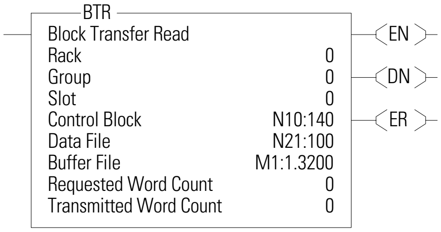
In this example, the remote adapter is using 2-slot addressing. It is assigned 1/4-logical rack of the scanner's I/O image files starting at RIO Logical Rack 3, Logical Group 4. The remote adapter controls two analog devices, which are configured for block transfer operations and two discrete devices (8-point input and 8-point output). Each analog module uses both the input and output byte of the logical slot to which it is assigned, while the discrete modules use only the input or output byte.

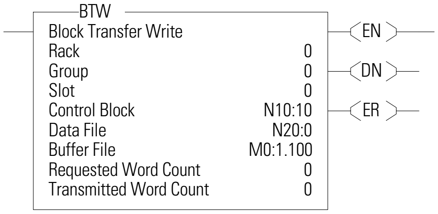
1747-SN RIO Scanner's I/O Image Files



Using Block Transfer Instructions (BTR and BTW)

Block transfer instructions are supported by SLC 5/03 (OS302, Series C), SLC 5/04 (OS401, Series C) and SLC 5/05 (OS501, Series C) and higher processors only. For application examples for block transferring with SLC 5/02 processors, refer to Chapter 5 of the *Remote I/O Scanner User Manual*, publication number 1747-6.6. With block-transfer instructions, you can transfer up to 64 words to or from a remote device over an Allen-Bradley Remote I/O (RIO) link. A Block Transfer Read (BTR) is used to receive data from a remote device. A Block Transfer Write (BTW) is used to send data to a remote device. The RIO Series B scanner (1747-SN) and the back-up scanner (1747-BSN) perform block transfers via M0 and M1 file buffers.





A false-to-true rung transition initiates a BTW or BTR instruction. The BTW instruction tells the processor to write data stored in the BTW Data File to a device at the specified RIO rack/group/slot address. The BTR instruction tells the processor to read data from a device at the specified RIO rack/group/slot address and store it in the BT Data File. A total of 32 block transfer buffers are available; you can execute a maximum of 32 different block transfers. The processor runs each block transfer request in the order it is requested. When the processor changes to Program mode, all pending block transfers are cancelled.

A BTR or BTW instruction writes information into its control structure address (a three-word integer Control Block) when the instruction is entered. The processor uses these values to execute the transfer.

You must enter an M1 file address into BTR Instructions and an M0 file address into BTW Instructions. However, each instruction uses both the M0 and M1 file for that buffer number (1 through 32). For example, to use the first available buffer (1) for a BTR, enter M1:e.100 into the "Buffer File" field. However, M0:e.100 is also used by this BTR. So, the next BT instruction must use another M-file buffer (2 through 32).

RIO Block Transfer General Functional Overview

The RIO scanner performs block transfers through control/status buffers allocated in the scanner's M0 and M1 files. For BTW's, the data stored in the File is copied into the M0 block transfer buffer, the M0 block transfer buffer is then transferred to the RIO device. The corresponding M1 block transfer buffer contains only BTW status information. For BTR's, the M0 block transfer buffer contains only BTR control information. The actual data read from the remote device is received in the scanner's M1 block transfer buffer. This data is then copied into the BTR Data File. A total of 32 block transfer control/status buffers exist in the M0 (output/control) and the M1 (input/status) files.

Parameters for BTR and BTW

The instructions have the following parameters:

- Data File - The address in the SLC processor's data file containing the BTW or BTR data.
- BTR/BTW Buffer File - Block transfer buffer file address; i.e. M0: e.x00, where "e" is the slot number of the scanner and "x" is the buffer number. The range of the buffer number is from 1 to 32. Each BTR and BTW instruction uses both the M1 and M0 files for a specific buffer number. M0 is used for BTR and BTW control and for BTW data. M1 is used for BTR and BTW status and BTR data.
- Control - The control block is an integer data file address that stores all the block transfer control and status information. The control block is three words in length. Note that these integer file addresses should not be used for any other instructions. You should provide the following information for the control structure.
  - Rack - The I/O rack number (0 to 3) of the I/O chassis in which you placed the target I/O module.
  - Group - The I/O group number (0 to 7) which specifies the position of the target I/O module in the I/O chassis. When using 1/2-slot addressing, only even group numbers are valid.
  - Slot - The slot number (0 or 1) within the group. When using 2-slot addressing, the 0 slot is the low (right) slot and the 1 slot is the high (left) slot within the group. When using 1-slot or 1/2-slot addressing, always select slot 0.
  - Requested Word Count - The number of words to transfer. If you set the length to 0, the processor reserves 64 words for block transfer data. The block transfer module transfers the maximum words the adapter can handle. If you set the length from 1 to 64, the processor transfers the number of words specified.

TIP

The three-word control block has the following structure. Before executing a block transfer, the BTR and BTW instructions clear all status bits and initialize word 2 to 0. See Table 7.1, "Control Block Structure," for more information.

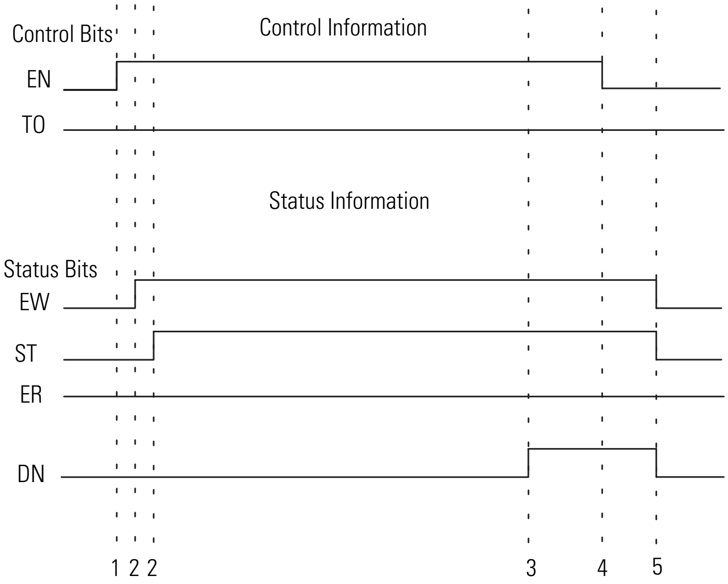
Table 7.1 Control Block Structure

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Word 0	EN	ST	DN	ER		EW		TO	RW		Rack			Group		Slot
Word 1	Requested word count															
Word 2	Transmitted word count/Error code															

Control Status Bits

To use the BTR and BTW instructions correctly, examine the instruction's control and status bits stored in the control structure. These bits are mapped to bits in word 0 of the control block structure.

Figure 7.1 Successful Block Transfer



Successful Block Transfer Read/Write

The illustration on the previous page shows a successful BT operation.

1. The SLC control program copies new data to the data file (BTW only) and solves the BT rung true, which sets the enable (EN) bit.

2. The scanner detects that the EN bit is set, validates the control block information, puts the BT request on the RIO link successfully, and since no other BTs are pending for the same logical rack, sets the enable waiting (EW) and start (ST) bits.

3. The scanner receives a BT reply (with no errors) from the RIO link device, copies the received data to the data file (BTR only) and sets the done (DN) bit.

4. The SLC control program detects the DN bit, processes the BTR data and solves the BT rung false, which clears the enable (EN) bit.

5. The scanner detects that the SLC control program has completed processing (because the EN bit is clear) and clears the EW, ST and DN bits. At this point, the SLC control program could re-initiate the same BT operation by solving the BT rung true again.

TIP Except for the time-out bit, TO (bit 08), do not modify any controller status bits while the block transfer is in progress.

IMPORTANT The BTR/BTW instruction must be scanned (true or false) in order to update the control and status bits.

Table 7.2 Control and Status Bit Descriptions

Control/Status Bit	Description
Enable EN (bit 15)	Block Transfer Enabled - (EN = Enabled). The processor sets/resets this bit depending on the rung state (true/false). The processor sends the enable bit to the RIO scanner when the BTR/BTW instruction is scanned. If the BT is not waiting (EW set) and is not started (ST set), and the EN bit sees a false-to-true transition, the RIO scan triggers a BT.
Start ST (bit 14)	Block Transfer Started - (ST = Started). When the instruction is scanned (true or false), the processor reads this bit from the RIO scanner. The scanner sets this bit when the BT starts. The scanner resets this bit when the ladder logic (processor) clears the EN bit indicating the BT is finished.
Done DN (bit 13)	Block Transfer Successful - (DN = Done). When this bit is set, it indicates the successful completion of a block transfer operation. When the instruction is scanned (true or false), the processor reads the DN bit from the RIO scanner. The scanner clears the DN bit when the ladder logic (processor) clears the EN bit.
Error ER (bit 12)	Block Transfer Error - (ER = Error). When this bit is set, it indicates that the process detected a failed block transfer. When the instruction is scanned (true or false), the processor reads the ER bit from the RIO scanner. The scanner clears the ER bit when the ladder logic (processor) clears the EN bit.
Enable-waiting EW (bit 10)	Block Transfer Enabled and Waiting for Block Transfer to Start - (EW = Enable Waiting). When the EW bit is set and the ST bit is clear, this indicates that a block transfer operation is pending. When the instruction is scanned (true or false), the processor reads the EW bit from the scanner. The scanner clears the EW bit after the ladder logic (processor) clears the EN bit.
Time Out TO (bit 08)	Block Transfer Time-out (TO = Time-out). You can set this bit to cancel block transfer operation by forcing the BT to time out once the Enabled Waiting (EW) bit sets and before the RIO scanner's internal four-second block transfer timer times out or the block transfer completes. Cancelling a block transfer causes an error (ER) bit to set and an error code of -9 to display in the control structure. Note that the Time-out (TO) bit must be cleared before initiating a new block transfer. The RIO scanner ignores a block transfer request if both TO and EN bits are set at the same time.
Read-Write RW (bit 07)	Block Transfer Type. This bit is controlled by the instruction type. A "0" indicates a write operation (BTW); a "1" indicates a read operation (BTR).

In addition to the control and status bits, the control block contains two other parameters the processor uses to execute the block transfer instructions.

Requested Word Count, Word 1 (RLEN)

This is used to configure BTR/BTW length information (0 to 64). Length is the number of BTR/BTW words read from or written to the RIO device. If RLEN = 0 for a BTW instruction, 64 words are sent. If RLEN = 0 for a BTR instruction, the actual length is determined by the RIO device responding to the block transfer read request.

Transmitted Word Count/Error Code, Word 2 (DLEN)

Transmitted Word Count is the status of the actual number of BTW words sent or the number of BTR words received. The processor uses this number to verify the transfer. This number should match the requested word count (unless the transmitted word count is zero). If these numbers do not match, the processor sets the ER bit (bit 12). If there is an error, the processor gives the error code in Word 2 of the control structure in the form of a negative number. See Table 7.3, “BTR/BTW Error Codes,” for a list of error codes. Only one error code is stored at a time (a new error code overwrites the previous error code).

Table 7.3 BTR/BTW Error Codes

Error Code	Description
0	The block transfer completed successfully.
-6	Illegal block transfer length requested.
-7	Block transfer communication error occurred when block transfer request was initiated.
-8	Error in block transfer protocol.
-9	Block Transfer Time-out - Either the SLC user program cancelled the block transfer or the scanner’s block transfer timer timed out. Note that a time-out error occurs if a block transfer is attempted at a location that is not configured for block transfer operation (e.g., requesting a block transfer for a location that is an output module).
-10	No RIO channel configured.
-11	Attempted a block transfer either to a non-configured block transfer device (i.e., an invalid logical rack, group, or slot), or at a complementary device location where there is no corresponding primary image space allocated.
-12	Attempted a block transfer to an inhibited device.

Instruction Operation

1. The scanner processes the BTR/BTW when it detects that the SLC control program rung, which contains the BTR/BTW, goes true.

   If the RIO scanner detects any problem at this point (such as invalid block transfer control field, or unconfigured device), the control structure word 2 fills with the error code and the ER bit (bit 12) is set. If no problems occur, the EW bit (bit 10) and ST bit (bit 14) are set in the control block.

   TIP The ST bit is not set if the scanner is already in the process of block transferring data to a location within the same logical RIO rack. The ST bit is set only after any previous pending block transfers to the same logical rack are completed and the block transfer request is scheduled on the RIO link.

   The SLC control program can monitor the block transfer by examining bits in word 0 of the control block. They indicate when the scanner has started processing (EW and ST) the block transfer and whether the block transfer operation completed successfully (DN) or failed (ER). The SLC control program can take different actions based on these status bits.

2. When a block transfer completes successfully, the DN bit is set. This indicates that the block transfer control block has been updated with the actual transmitted word count. This is important for BTR instructions, because this indicates the number of valid data words received from the remote device. This data is stored in the BTR data file.

3. If the block transfer fails, the length field and the data file are not updated. The ER bit is set and the error code field indicates the problem.

4. The SLC control program must indicate to the scanner when it is done processing the status word in the control structure (because DN or ER was set) so the corresponding control bits can be reused for another block transfer operation. The SLC control program indicates that it is done processing the block transfer when it solves the BT rung false, which clears the EN bit in the control block.

5. When the RIO scanner detects that the EN bit cleared, it then clears the EW, ST and DN or ER bits, as well as the Transmitted Word Count/Error Code. This ensures that the status bits in the control block are not reflecting the results of the previous block transfer operation.

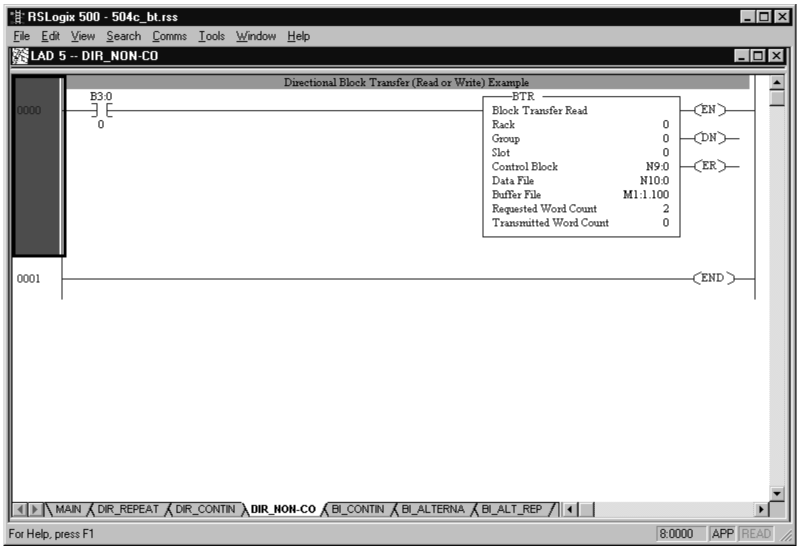
IMPORTANT To prevent configuration conflicts, it is highly recommended that each M-file buffer (My:e.x00) should be used by only one block transfer instruction.

Programming Examples

Table 7.4 Block Transfer Programming Examples

Figure 7.2, "Directional" on page 7-18
Figure 7.3, "Directional Repeating" on page 7-18
Figure 7.4, "Directional Continuous" on page 7-19
Figure 7.5, "Bi-directional Continuous" on page 7-19
Figure 7.6, "Bi-directional Alternating" on page 7-20
Figure 7.7, "Bi-directional Alternating Repeating" on page 7-20

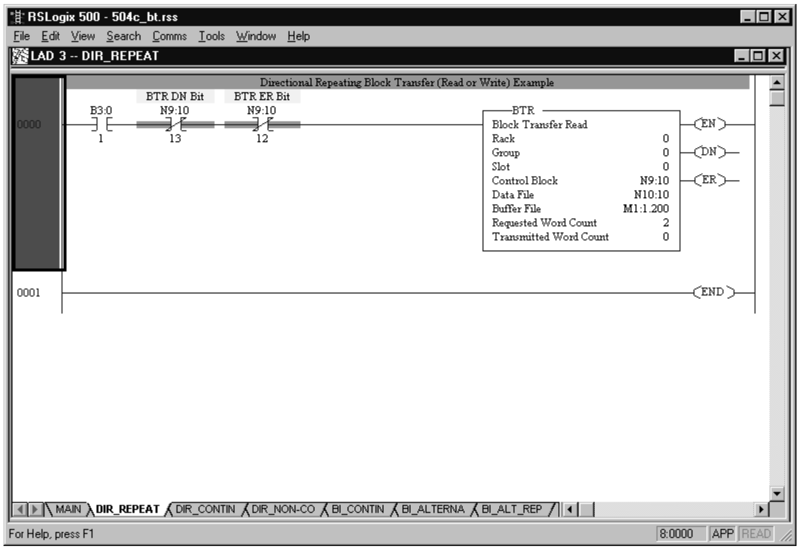
Figure 7.2 Directional



Directional Block Transfer (Read or Write) Example

BTR
Block Transfer Read
Rack 0
Group 0
Slot 0
Control Block N9:0
Data File N10:0
Buffer File M1:1.100
Requested Word Count 2
Transmitted Word Count 0

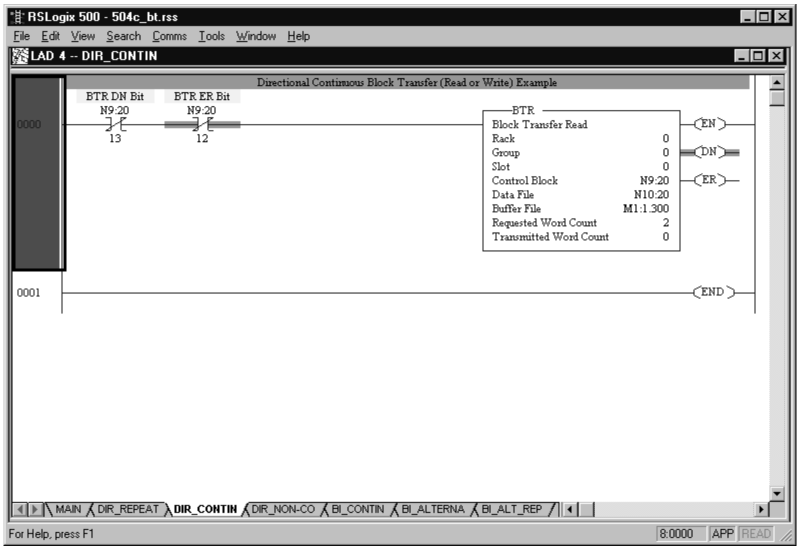
Figure 7.3 Directional Repeating



Directional Repeating Block Transfer (Read or Write) Example

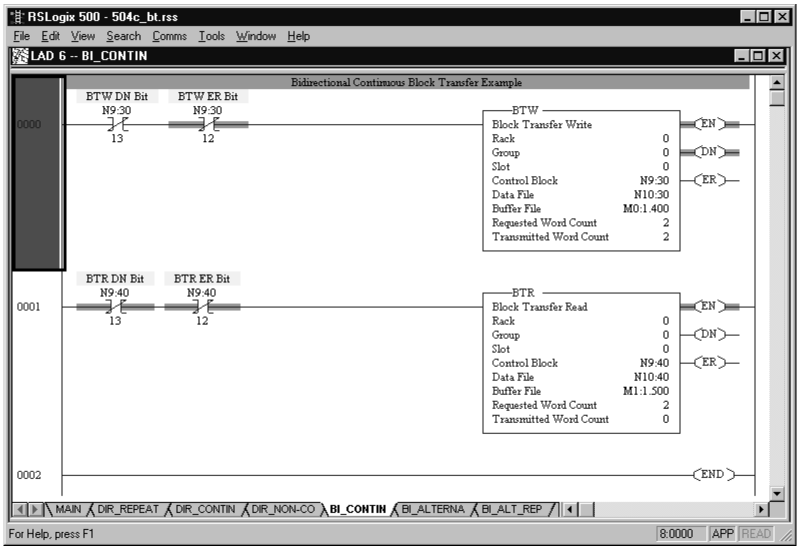
BTR
Block Transfer Read
Rack 0
Group 0
Slot 0
Control Block N9:10
Data File N10:10
Buffer File M1:1.200
Requested Word Count 2
Transmitted Word Count 0

Figure 7.4 Directional Continuous



BTR
Block Transfer Read
Rack 0
Group 0
Slot 0
Control Block N9:20
Data File N10:20
Buffer File M1:1.300
Requested Word Count 2
Transmitted Word Count 0

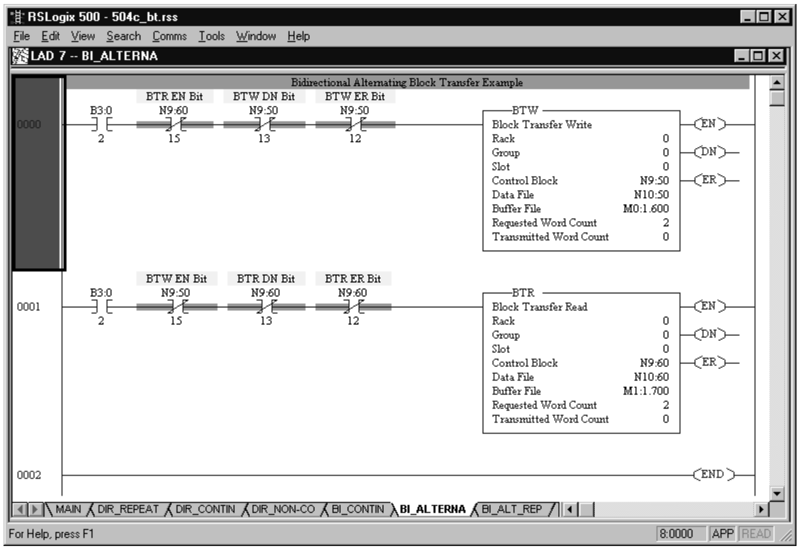
Figure 7.5 Bi-directional Continuous



BTW
Block Transfer Write
Rack 0
Group 0
Slot 0
Control Block N9:30
Data File N10:30
Buffer File M0:1.400
Requested Word Count 2
Transmitted Word Count 2

BTR
Block Transfer Read
Rack 0
Group 0
Slot 0
Control Block N9:40
Data File N10:40
Buffer File M1:1.500
Requested Word Count 2
Transmitted Word Count 0

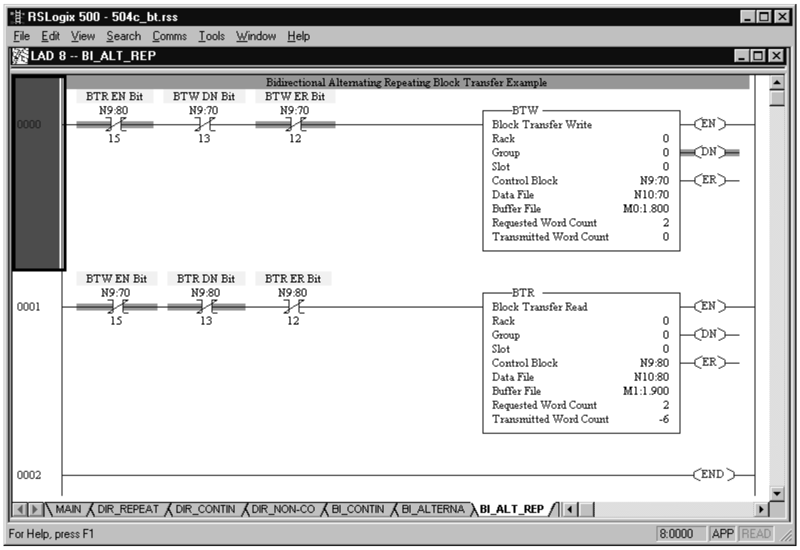
Figure 7.6 Bi-directional Alternating



BTW
Block Transfer Write
Rack 0
Group 0
Slot 0
Control Block N9:50
Data File N10:50
Buffer File M0:1.600
Requested Word Count 2
Transmitted Word Count 0

BTR
Block Transfer Read
Rack 0
Group 0
Slot 0
Control Block N9:60
Data File N10:60
Buffer File M1:1.700
Requested Word Count 2
Transmitted Word Count 0

Figure 7.7 Bi-directional Alternating Repeating



BTW
Block Transfer Write
Rack 0
Group 0
Slot 0
Control Block N9:70
Data File N10:70
Buffer File M0:1.800
Requested Word Count 2
Transmitted Word Count 0

BTR
Block Transfer Read
Rack 0
Group 0
Slot 0
Control Block N9:80
Data File N10:80
Buffer File M1:1.900
Requested Word Count 2
Transmitted Word Count -6

Comparison to the PLC-5 BTR and BTW

Block Transfer Reads and Writes in SLC processors are quite similar to the instructions in the PLC-5. However, some differences exist between them, as shown in Table 7.5 on page 7-21.

Table 7.5 Block Transfer Comparison

	SLC	**PLC-5**
Control Block	3-element integer (N) type	5-element integer (N) type or 1-element block transfer (BT) type.
EN (Enable Bit)	Follow BT rung state.	Gets set when BT rung goes true. Remains set until the BT finishes or fails, and the BT rung goes false.
NR (No Response bit)	None	This bit is in control block word 0, bit 9.
CO (Continuous bit)	None	This bit is in control block word 0, bit 11.
FILE (File Number)	None	This word is control block word 3.
ELEM (Element Number)	None	This word is control block word 4.
Error Codes	7 error codes	11 error codes
BTR/BTW number limitation for one scanner/channel	32	64
BT Status Bits	Can only change when BT rung is scanned.	Can change at any point in the program scan.

IMPORTANT Do not manipulate the I/O image words of the RIO scanner for modules to which you are block transferring. These words are used by the RIO scanner and the remote device as block transfer handshake bits. Any manipulation of them by the user program while a block transfer is in progress causes the block transfer to fail.

Switchover Considerations

When planning programs for the SLC 500 backup system, you must first consider that the program scans of the two processors are not synchronized. This means that the program in the primary processor is not executing the exact same instruction at the same time as the program in the secondary processor. In addition, you must consider:

- timing
- divergence
- I/O forces
- Data Highway Plus switching
- remote I/O switching
- special sections of the data table
- data integrity

This chapter describes the switchover considerations listed above, as well as switchover diagnostics. It also describes how to detect a possible fault in the secondary module in order to avoid switchover problems in the backup system.

Timing Requirements

Switchover from one processor to another takes less than 50 ms.

TIP Total switchover time is a factor of the 1747-BSN module delay and one program scan of the secondary processor. The time required for one program scan is added because the secondary processor technically has not assumed control until it has completed one scan of its program and remote I/O after a switchover.

When a failure occurs in the primary system, the remote outputs remain in the state set by the primary processor prior to switchover, while the secondary processor assumes control of the process.

IMPORTANT You must program the backup system to account for the following time requirements:

- input signal update time
- time-out on remote I/O link
- data table transfer time over the HSSL

Input Signal Update Time

It is possible for the secondary processor to read different input conditions immediately after the switchover. To avoid a sudden change in the outputs during switchover, all of the input signals must be present for an amount of time equal to the sum of the following:

- program scan
- I/O scan
- input module delay
- 1747-BSN switchover time

If these conditions are not met, outputs may be assigned different states in the two processors. This can cause a sudden change of operation when switchover occurs.

Time-out on Remote I/O Link

The remote I/O chassis are updated one at a time. The total remote I/O update is 10 ms per chassis (typical) at 57.6K baud.

To keep the I/O chassis from faulting and losing control of the I/O, the switchover in the remote I/O link is accomplished in less than 50 ms. (The remote I/O link switchover starts when the remote I/O relays in the primary 1747-BSN module open and ends when the remote I/O relays in the secondary 1747-BSN module close.)

This switchover time is less than the 100 ms for the remote I/O adapter watchdog timeout. This permits continuous control of the I/O from the backup system processors.

Data Table Transfer Time on HSSL

Data table transfer time refers to the amount of time it takes to transfer critical data from the primary system to the secondary system. This time is dependent on the amount of data to be sent between the two systems, the number of remote I/O chassis, and the number of remote I/O block transfers that are being executed.

The 1747-BSN module is capable of transferring up to 100K Word per second (10 msec per K Word). This data rate is 2M bits/second.

Divergence

The synchronization of program execution in both processors limits divergence between the two systems by ensuring that the resulting output data in both processors is always identical.

Because the program scans of the processors are not synchronized, it is possible that the two separate systems read the same input at different times. Eventually, different input values are read, thus different decisions are made and the internal states of the two systems diverge. If the two systems diverge, the secondary system generates a “bump” when it takes control over the process.The “bump” time is a function of how fast the inputs of the machine or the process are changing, as well the amount of data to send between the primary and secondary units.

One possible method to avoid or reduce the bump is the “loosely synchronized backup” approach. This method requires special user programming techniques. With this approach, primary and the secondary SLC 5/0x execute the same program and are synchronized at a waiting point once per program scan.

To limit divergence, we recommend that the primary and secondary processors execute the same programs.

Forcing I/O

Backup communication functions provided by the SLC 500 backup system do not transfer the force tables from the primary to the secondary processors. Therefore, if the SLC 500 backup system has forces set in the primary processor, the forces do not transfer to the secondary processor. For this situation, if a switchover occurs, the secondary processor becomes active, but the forces set in the original primary processor are not carried over.

ATTENTION

Set the forces in the secondary processor first. Then set the forces in the primary processor. Likewise, when removing forces, remove forces in the secondary processor before removing the forces in the primary processor. If a switchover occurs and forces were not set in the secondary processor first, the secondary processor does not recognize the forced-on input or output.

Data Highway Plus Switching

You can aggregate a 1747-BSN with the communication channel configured as DH+ network to the system. In this situation, both primary and secondary processors must be set to the same node address. During normal operation, the other equipment in the DH+ network recognize the primary processor in its real node address and the secondary processor in the node address which follows.

When a switchover occurs, the new primary SLC 5/04 assumes its real node address in the DH+ network and the new secondary processor assumes the next node address.

The DH+ smart switch is designed to provide remote programming capability for the secondary processor with the following limitations:

- The secondary SLC 5/04 can be accessed only by one station in the primary link each time. If more than one station tries to access the secondary SLC 5/04 at the same time, the communication performance with this node is degraded significantly.
- The secondary SLC 5/04 only communicates with the secondary 1747-BSN. If a third device is connected to this link, it will not work properly.
- The secondary SLC 5/04 only answers to commands. It cannot start a MSG instruction.
- The primary SLC 5/04 cannot send messages through DH+ to the secondary SLC 5/04.

The secondary processor can be programmed through an access-point address which corresponds to the processor node address plus one (n+1). The secondary processor becomes part of the link only after a switchover. If the real node address is 5, the primary is the node 5 and the secondary is node 6. If the real node address is 77, the primary is node 77 and the secondary is node 0.

It is possible that the primary processor could have possession of the token during a switchover from the primary to the secondary processor. In this case, the token could be lost even though passage of the token from one station to the next is done as quickly as possible.

If the token is lost, all the stations on the link have an internal watchdog timeout (250 ms) and the nodes assume the token is lost.

The other stations on the link initiate a token recovery procedure which includes:

- recreating the active node table
- rebuilding the link, including the new primary processor
- ensuring that one node gains control of the link

During this time, while the stations reconstruct the link, communication is interrupted. The length of the interruption depends on the number of stations involved in the link and the number of messages received and/or transmitted. This time is typically less than 50 ms per station.

However, while the link is rebuilding, messages can collide, increasing the amount of time needed to reconstruct the link. As a result, on a typical DH+ link with 6 to 10 stations, the impact on the link due to switchover could be the loss of communication for as much as 1 to 3 seconds.

DH+ switching occurs whenever there is a transfer of control from the primary to the secondary processor. The switchover could result from a power failure or because of other system failures. The table below shows what can happen to the link if the token is lost.

IMPORTANT When a transfer of control from the primary to the secondary processor occurs, the processor that is now primary has the node address of *n* on the DH+ link. The processor that is now secondary has the node address of *n*+1.

If the new primary is:	the following could occur	causing this result
Polling the link	Message packet from another station disrupted	The source station turns on error bit in its MSG instruction. You provide programming at the source station to regain synchronization with the receiving station. You can do this by monitoring the message instruction error bit as a condition for retransmitting the message.
	Reply packet disrupted	Same as above
	Token pass packet disrupted	The DH+ link times out and must rebuild. (250ms watchdog timeout) + (50 ms x number of stations on the link).
Listening to the link	New primary processor takes the token	The new primary processor has nothing in its active node table. The DH+ link times out and must rebuild. (250 ms watchdog timeout) + (50 ms x number of stations on the link).
	New primary processor receives a message from another station	The link continues working normally.

IMPORTANT In the recommended configuration, set both SLC 500 processors to the same station address. The backup modules permit only one of the two processors to be connected to the DH+ at a time. In this configuration, remote PLC processors and operator stations communicate to the processor that currently has primary status. The secondary processor can also be accessed through the primary SLC 500 processor node address plus one (*n*+1).

Remote I/O Switching

The communications protocol for the remote I/O link is a master-slave half-duplex type. This means that the master station (processor) sends a command message addressed to a slave (remote I/O adapter), and the slave responds with a reply message addressed to the master. This is also called two-way alternating communication.

When the communication channel is configured as RIO, the primary 1747-BSN emulates a Series B 1747-SN module with all the features that it provides to the user. Additionally, the secondary 1747-BSN “spies” on the RIO link to acquire all the input data and use it to emulate the 1747-SN functionality to the secondary SLC 5/0x.

In the secondary system, the 1747-BSN simulates the entire scanner operation to the secondary SLC 5/04 with the following characteristics:

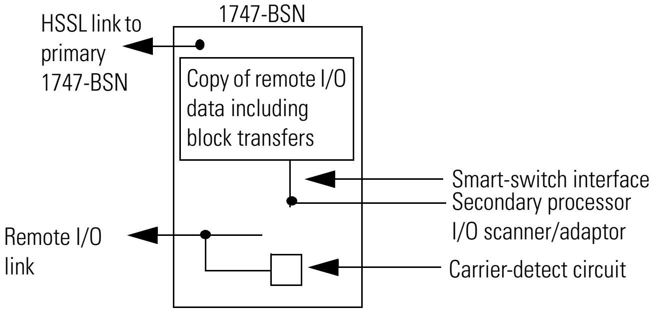
- Real discrete input data is acquired by the secondary BSN from the RIO network and transferred to the secondary SLC 5/04 through the input (I) file.
- Real BTR input data is acquired by the secondary BSN from the RIO network and transferred to the secondary SLC 5/04 through the M1 file. All the timing necessary for doing a BTR command in the SLC 5/04 is simulated by the secondary BSN.
- The discrete output data sent by the secondary SLC 5/04 through the output (O) file is stored in the output image table for using in case of a switchover.
- The BTW output data sent by the secondary SLC 5/04 through the M0 file is stored into the secondary BSN and all the timing for the BTW operation is simulated by the secondary BSN.
- In case of RIO communication errors seen by the BSN in the primary system (remote rack errors, BTR or BTW errors, etc.), the secondary BSN simulates the same errors for the secondary SLC 5/04. This simulation is done in order to assure that the secondary SLC 5/04 knows the real status of the remote devices.
- If a switchover occurs, all the input and output data inside the BSN is ready for the operation as the new primary scanner.

When the backup system is operating properly, the relays of the primary 1747-BSN are closed and the relays of the secondary 1747-BSN are open. Should a failure occur in the primary system, the relays of the primary 1747-BSN module open and the relays in the secondary 1747-BSN module close.

IMPORTANT The smart-switch interface communicates with the secondary processor only when the 1747-BSN module belongs to the secondary system. When the module is primary, the smart switch only monitors the remote I/O link to obtain input/output data.

The smart switch interface also includes a carrier-detect circuit to monitor the activity of the remote I/O link. The purpose of this circuit is to detect an idle remote I/O link before the secondary relays are closed. This guarantees that the former primary system was really disconnected from the link and the remote adapters have completed their response to the last poll from the primary scanner.

In addition, in the event of an HSSL break, the carrier-detect circuit prevents the secondary system from taking control of the link along with the primary system. The figure below shows the block diagram of the remote I/O switch.



Programming Techniques

Chapter Objectives

Read this chapter to familiarize yourself with techniques used to program your SLC 500 backup system.

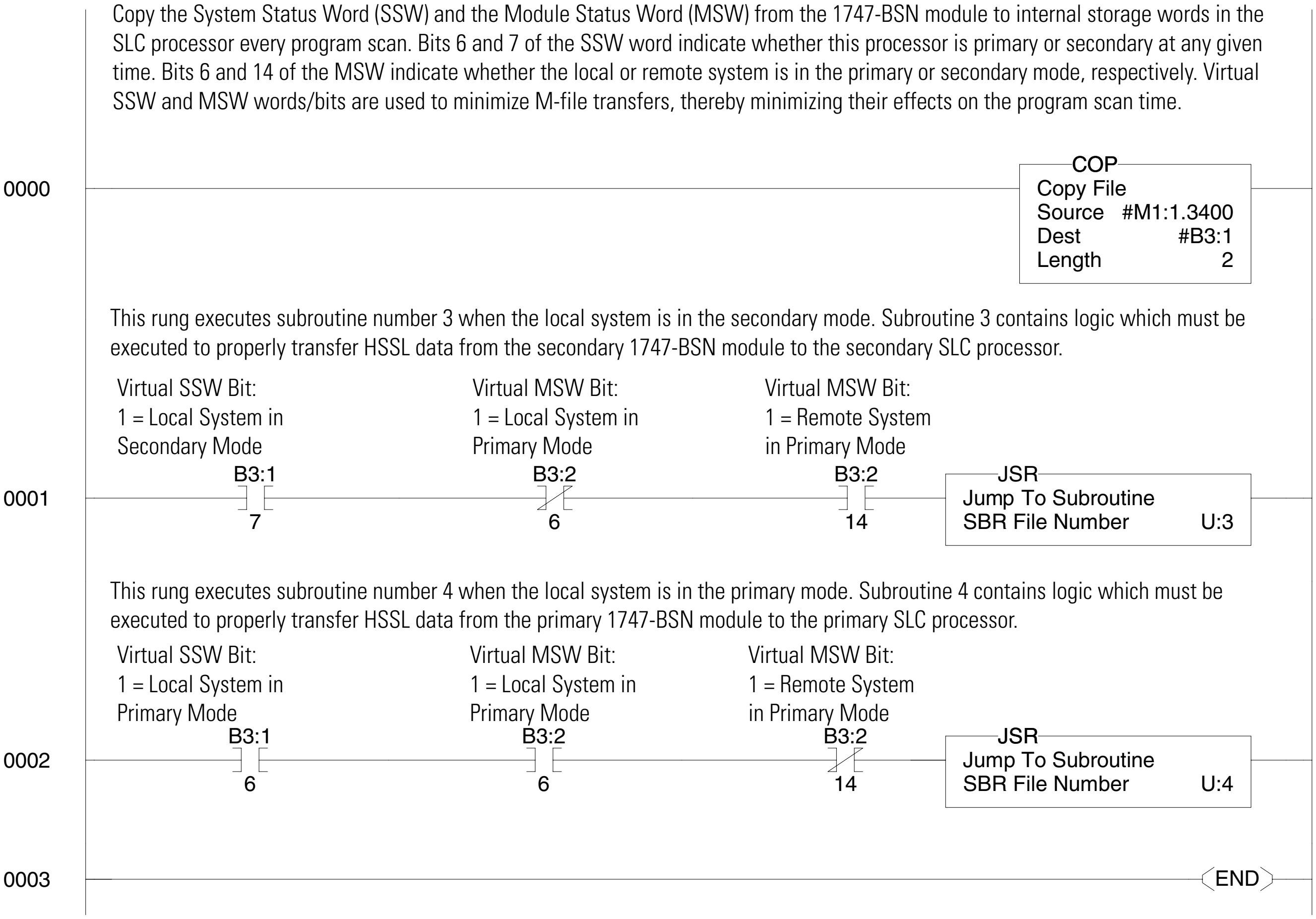
In this chapter we describe:

- how to get started with a program to transfer data table values
- two methods you can use to program the SLC 500 backup system to transfer data table values
- the behavior of specific instructions when used in your application program, and give you suggestions for dealing with these instructions.

Getting Started

The 1747-BSN modules control switchover and transfer remote I/O data without additional programming. However, if you want to transfer data table values from the primary to the secondary processor, you must provide a ladder program. The following figure shows the minimum ladder programming you must provide to transfer 2 Kwords of the data table between the two 1747-BSN modules, and to provide 1747-BSN module status. This program constantly transfers data as fast as possible. 2K words is the maximum per 1747-BSN module on the HSSL. With no local I/O, an SLC 5/02 or later processor supports up to 8 1747-BSN modules for a maximum of 16K words of retentive data transfer from primary processor to secondary processor. Refer to the following section, Data Transfer Schemes, for alternative approaches. You need to put this program in both processors.

Program File 2

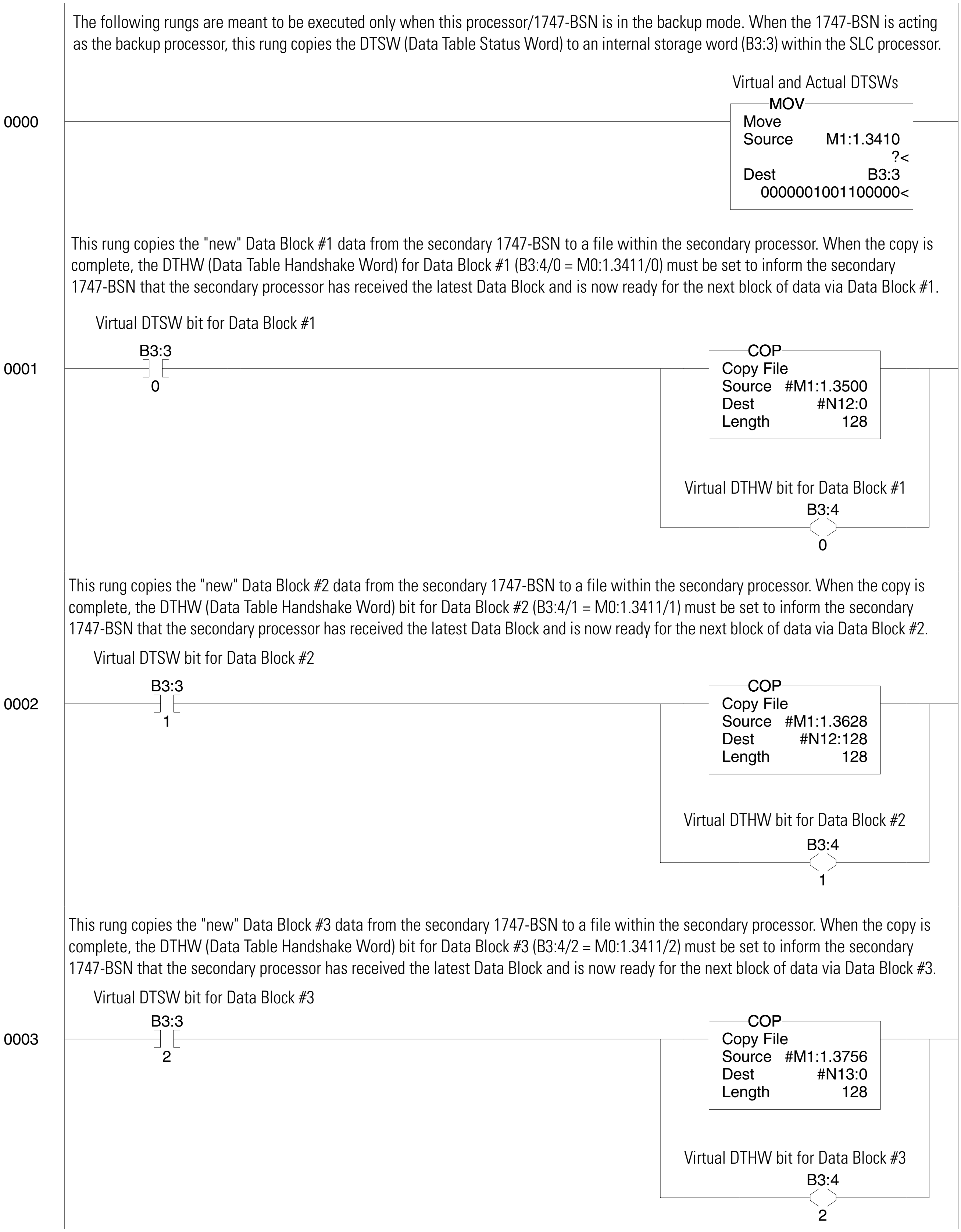


Copy the System Status Word (SSW) and the Module Status Word (MSW) from the 1747-BSN module to internal storage words in the SLC processor every program scan. Bits 6 and 7 of the SSW word indicate whether this processor is primary or secondary at any given time. Bits 6 and 14 of the MSW indicate whether the local or remote system is in the primary or secondary mode, respectively. Virtual SSW and MSW words/bits are used to minimize M-file transfers, thereby minimizing their effects on the program scan time.

This rung executes subroutine number 3 when the local system is in the secondary mode. Subroutine 3 contains logic which must be executed to properly transfer HSSL data from the secondary 1747-BSN module to the secondary SLC processor.

This rung executes subroutine number 4 when the local system is in the primary mode. Subroutine 4 contains logic which must be executed to properly transfer HSSL data from the primary 1747-BSN module to the primary SLC processor.

Program File 3



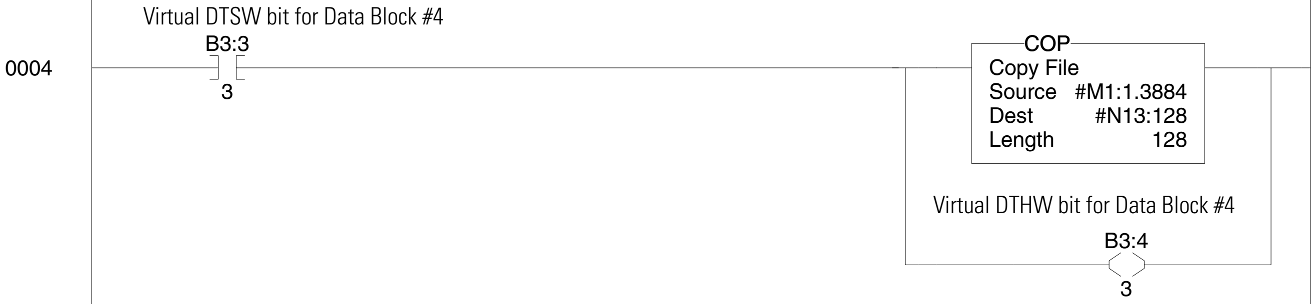
The following rungs are meant to be executed only when this processor/1747-BSN is in the backup mode. When the 1747-BSN is acting as the backup processor, this rung copies the DTSW (Data Table Status Word) to an internal storage word (B3:3) within the SLC processor.

This rung copies the "new" Data Block #1 data from the secondary 1747-BSN to a file within the secondary processor. When the copy is complete, the DTHW (Data Table Handshake Word) for Data Block #1 (B3:4/0 = M0:1.3411/0) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #1.

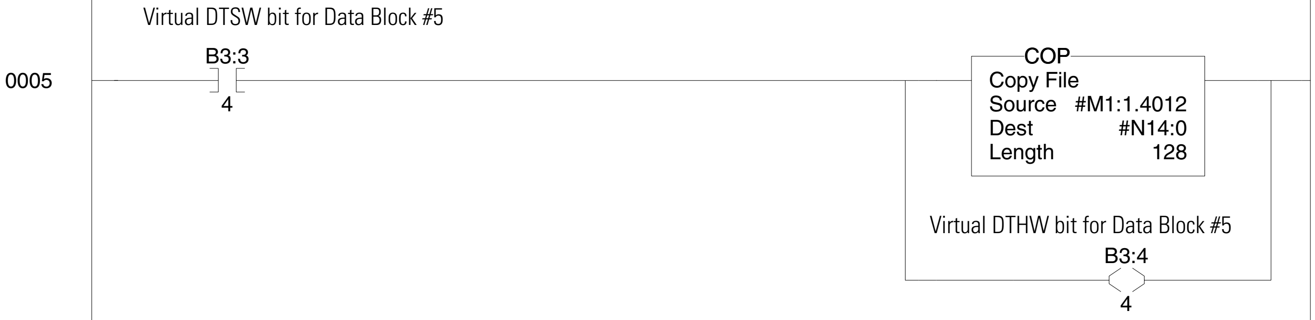
This rung copies the "new" Data Block #2 data from the secondary 1747-BSN to a file within the secondary processor. When the copy is complete, the DTHW (Data Table Handshake Word) bit for Data Block #2 (B3:4/1 = M0:1.3411/1) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #2.

This rung copies the "new" Data Block #3 data from the secondary 1747-BSN to a file within the secondary processor. When the copy is complete, the DTHW (Data Table Handshake Word) bit for Data Block #3 (B3:4/2 = M0:1.3411/2) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #3.

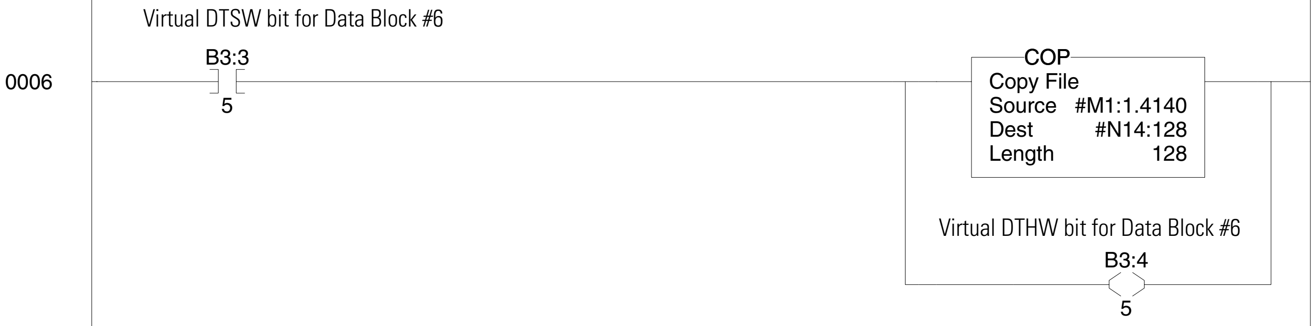
This rung copies the "new" Data Block #4 data from the secondary 1747-BSN to a file within the secondary processor. When the Copy is complete, the DTHW (Data Table Handshake Word) bit for Data Block #4 (B3:4/3 = M0:1.3411/3) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #4.



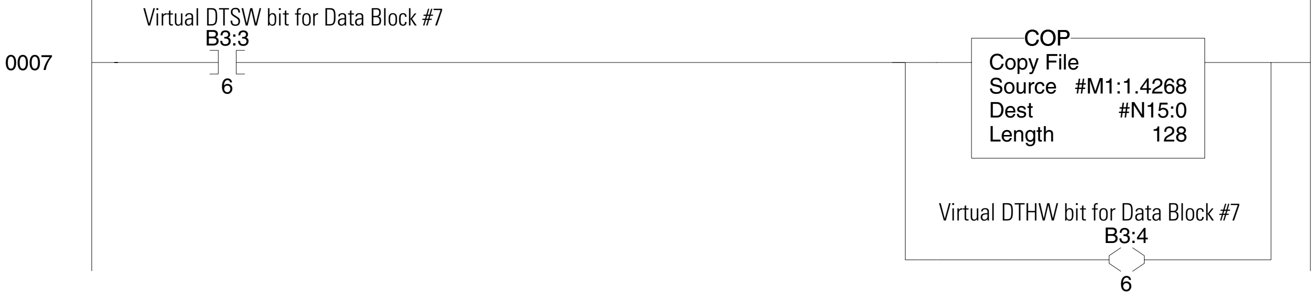
This rung copies the "new" Data Block #5 data from the secondary 1747-BSN to a file within the secondary processor. When the Copy is complete, the DTHW (Data Table Handshake Word) bit for Data Block #5 (B3:4/4 = M0:1.3411/4) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #5.



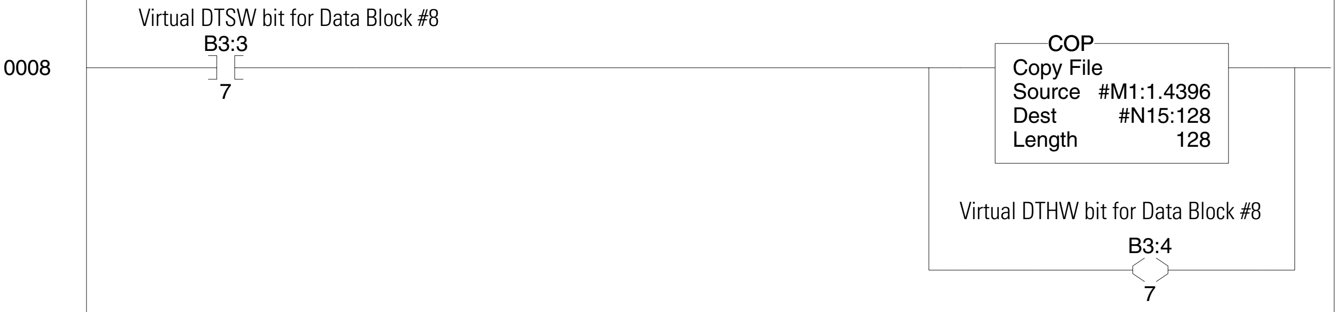
This rung copies the "new" Data Block #6 data from the secondary 1747-BSN to a file within the secondary processor. When the Copy is complete, the DTHW (Data Table Handshake Word) bit for Data Block #6 (B3:4/5 = M0:1.3411/5) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #6.



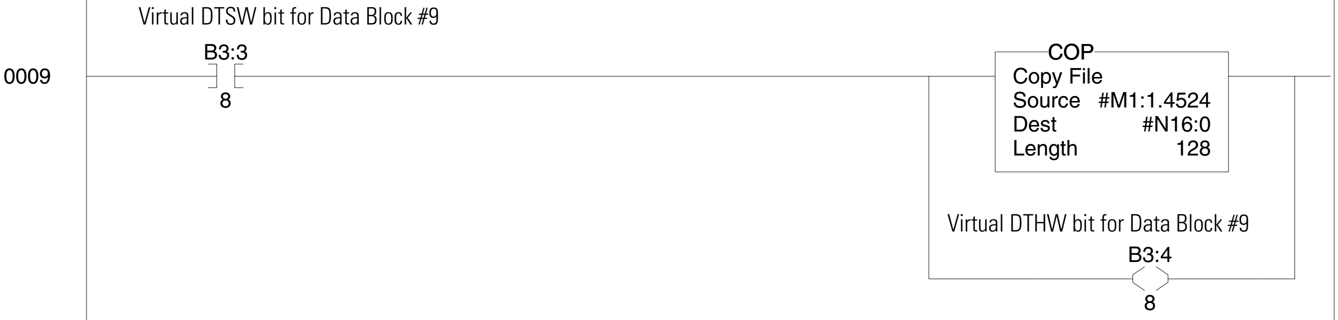
This rung copies the "new" Data Block #7 data from the secondary 1747-BSN to a file within the secondary processor. When the Copy is complete, the DTHW (Data Table Handshake Word) bit for Data Block #7 (B3:4/6 = M0:1.3411/6) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #7.



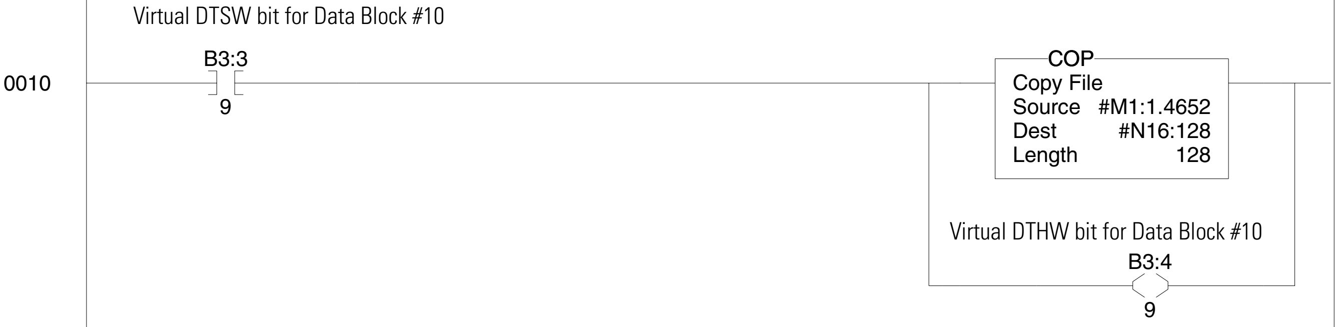
This rung copies the "new" Data Block #8 data from the secondary 1747-BSN to a file within the secondary processor. When the Copy is complete, the DTHW (Data Table Handshake Word) bit for Data Block #8 (B3:4/7 = M0:1.3411/7) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #8.



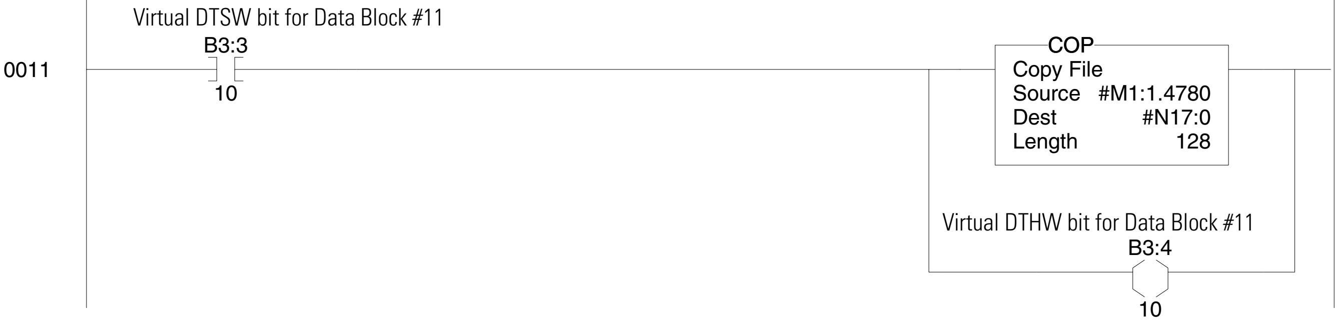
This rung copies the "new" Data Block #9 data from the secondary 1747-BSN to a file within the secondary processor. When the Copy is complete, the DTHW (Data Table Handshake Word) bit for Data Block #9 (B3:4/8 = M0:1.3411/8) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #9.



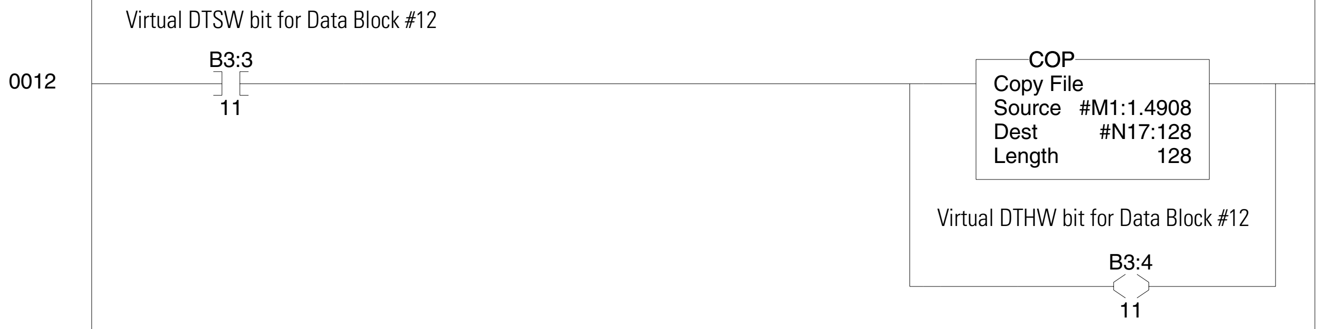
This rung copies the "new" Data Block #10 data from the secondary 1747-BSN to a file within the secondary processor. When the Copy is complete, the DTHW (Data Table Handshake Word) bit for Data Block #10 (B3:4/9 = M0:1.3411/9) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #10.



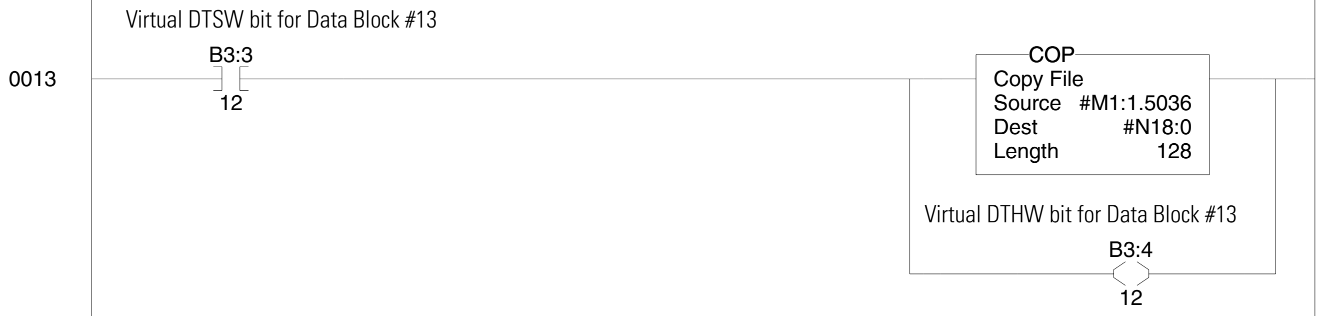
This rung copies the "new" Data Block #11 data from the secondary 1747-BSN to a file within the secondary processor. When the Copy is complete, the DTHW (Data Table Handshake Word) bit for Data Block #11 (B3:4/10 = M0:1.3411/10) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #11.



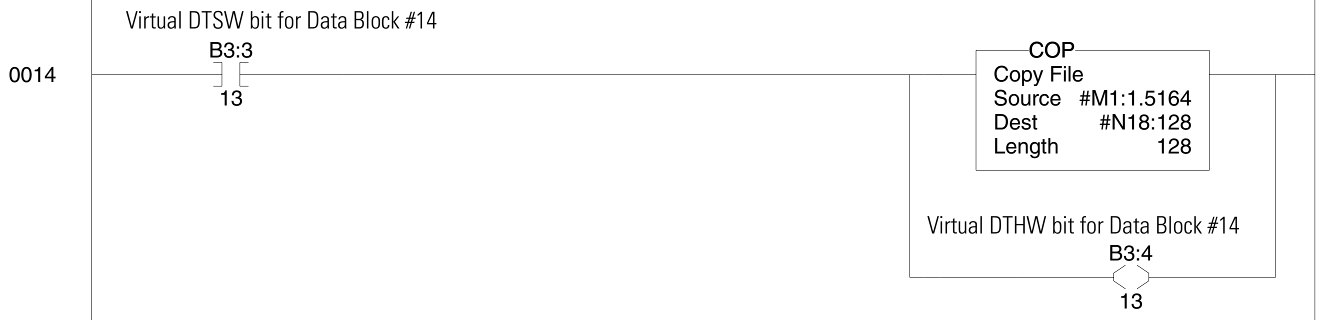
This rung copies the "new" Data Block #12 data from the secondary 1747-BSN to a file within the secondary processor. When the Copy is complete, the DTHW (Data Table Handshake Word) bit for Data Block #12 (B3:4/11 = M0:1.3411/11) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #12.



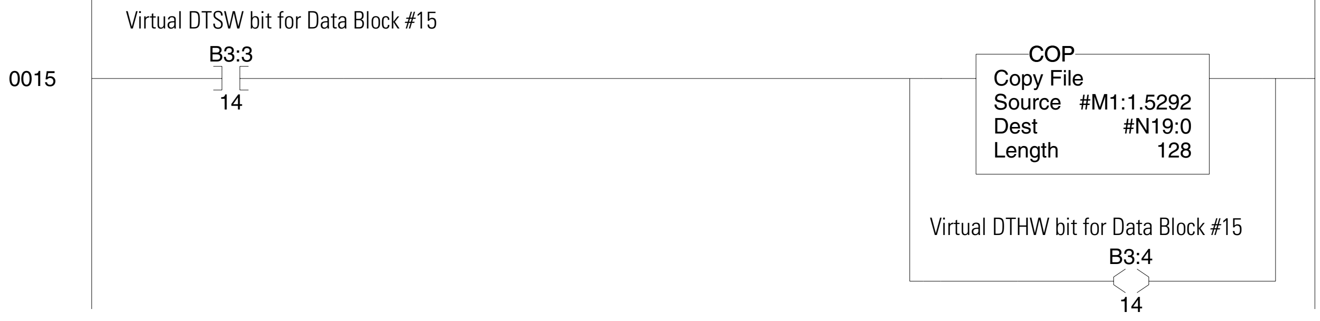
This rung copies the "new" Data Block #13 data from the secondary 1747-BSN to a file within the secondary processor. When the Copy is complete, the DTHW (Data Table Handshake Word) bit for Data Block #13 (B3:4/12 = M0:1.3411/12) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #13.



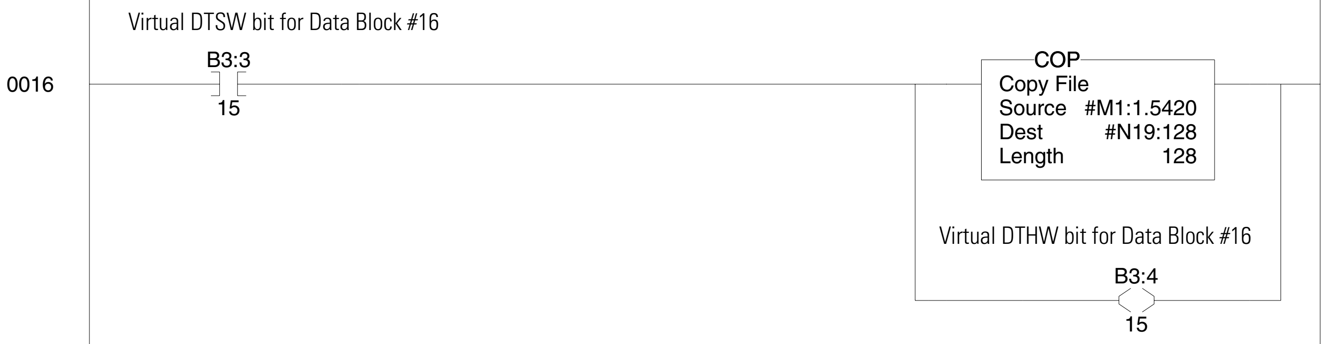
This rung copies the "new" Data Block #14 data from the secondary 1747-BSN to a file within the secondary processor. When the Copy is complete, the DTHW (Data Table Handshake Word) bit for Data Block #14 (B3:4/13 = M0:1.3411/13) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #14.



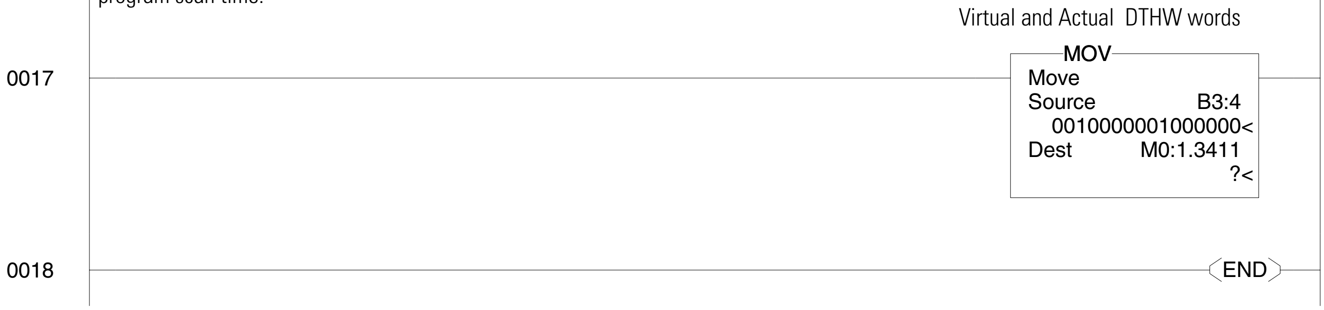
This rung copies the "new" Data Block #15 data from the secondary 1747-BSN to a file within the secondary processor. When the Copy is complete, the DTHW (Data Table Handshake Word) bit for Data Block #15 (B3:4/14 = M0:1.3411/14) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #15.



This rung copies the "new" Data Block #16 data from the secondary 1747-BSN to a file within the secondary processor. When the Copy is complete, the DTHW (Data Table Handshake Word) bit for Data Block #16 (B3:4/15 = M0:1.3411/15) must be set to inform the secondary 1747-BSN that the secondary processor has received the latest Data Block and is now ready for the next block of data via Data Block #16.

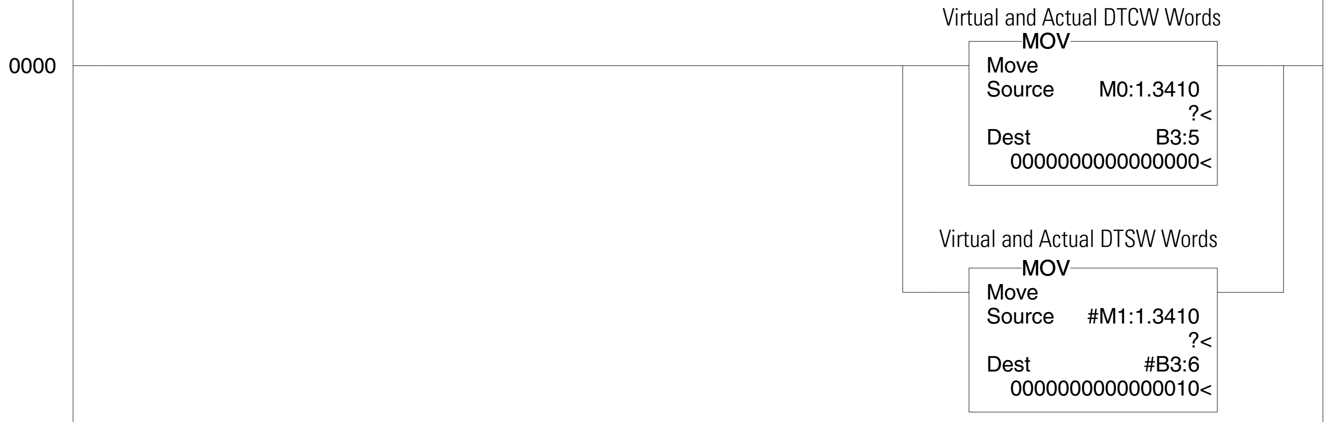


This rung copies the virtual DTHW word (B3:4) to the actual DTHW word (M0:1/3411) located in the 1747-BSN module. M-file accesses are interrupts to the processor. Using virtual words minimizes M-file accesses, thereby minimizing the effects of these transfers on the ladder program scan time.

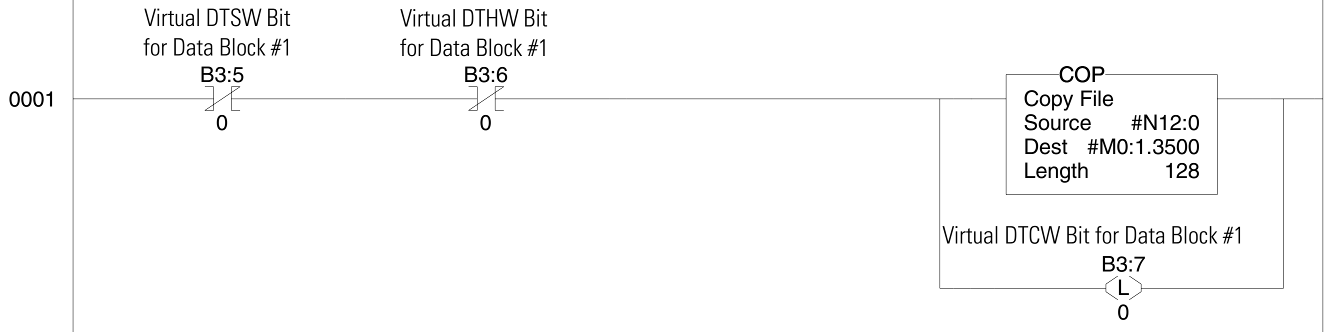


Program File 4

The following six rungs are meant to be executed only when this processor/1747-BSN is in the Primary mode. When the 1747-BSN is acting as the primary processor, this rung copies the DTCW and DTSW words to internal storage (B3:5 and B3:6 respectively) within the SLC processor.

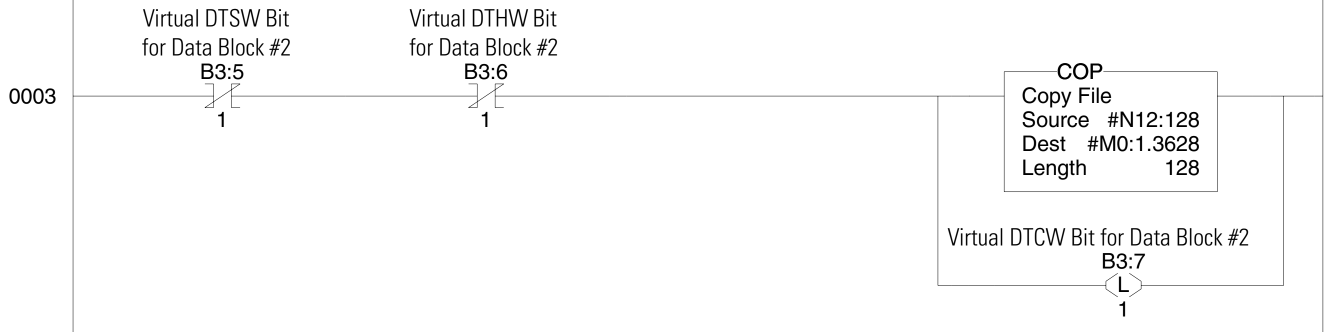


This rung monitors the DTCW and DTSW bits for Data Block #1. When a transfer of data using Data Block #1 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #1 in the 1747-BSN module.



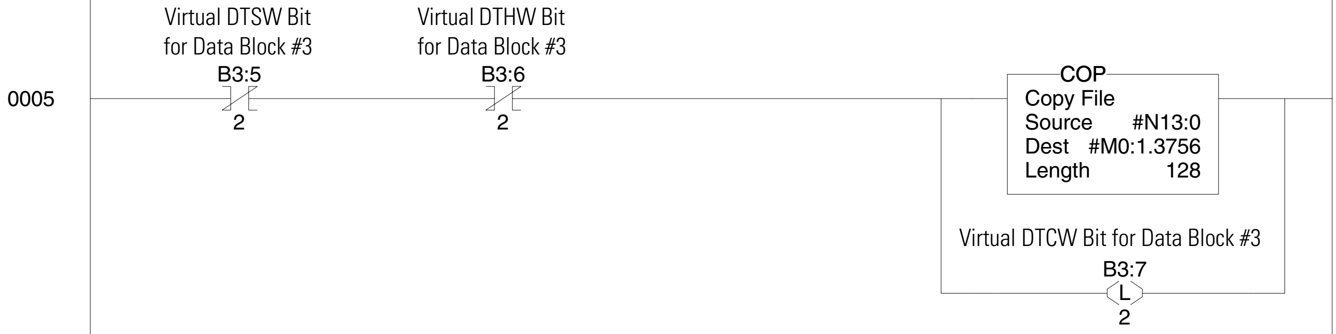
Unlatch the virtual DTCW bit for Data Block #1 when the comparable virtual DTSW bit is set by the 1747-BSN module to indicate that it has received the last data via Data Block #1. When B3:7/0 has been unlatched, the module is ready for the next data transfer via Data Block #1.

This rung monitors the DTCW and DTSW bits for Data Block #2. When a transfer of data using Data Block #2 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #2 in the 1747-BSN module.

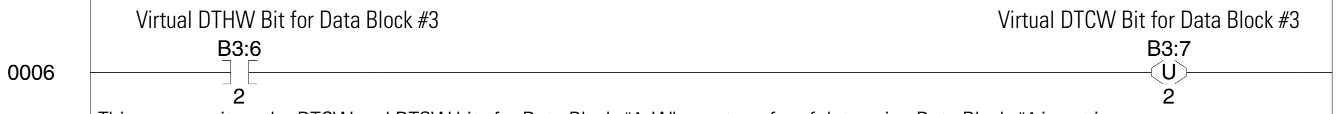


Unlatch the virtual DTCW bit for Data Block #2 when the comparable virtual DTSW bit is set by the 1747-BSN module to indicate that it has received the last data via Data Block #2. When B3:7/1 has been unlatched, the module is ready for the next data transfer via Data Block #2.

This rung monitors the DTCW and DTSW bits for Data Block #3. When a transfer of data using Data Block #3 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #3 in the 1747-BSN module.

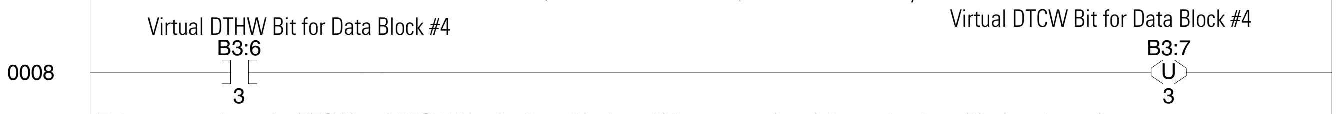


Unlatch the virtual DTCW bit for Data Block #3 when the comparable virtual DTSW bit is set by the 1747-BSN module to indicate that it has received the last data via Data Block #3. When B3:7/2 has been unlatched, the module is ready for the next data transfer via Data Block #3.

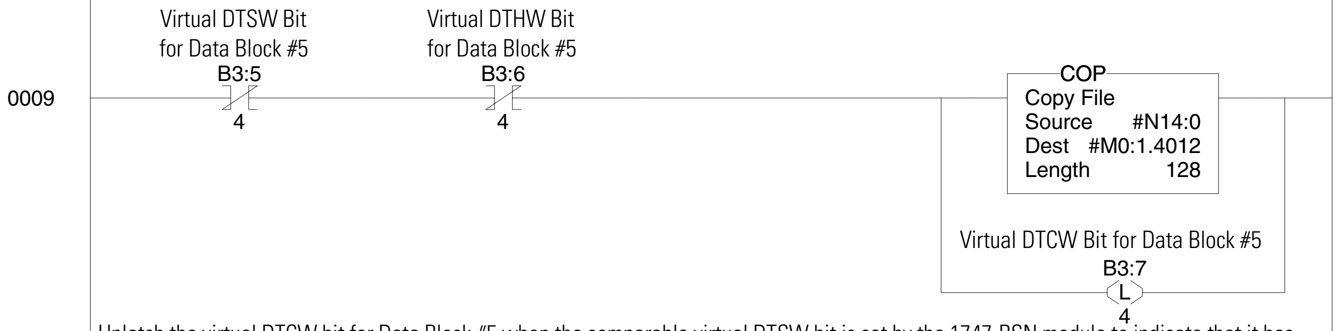


This rung monitors the DTCW and DTSW bits for Data Block #4. When a transfer of data using Data Block #4 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #4 in the 1747-BSN module.

Unlatch the virtual DTCW bit for Data Block #4 when the comparable virtual DTSW bit is set by the 1747-BSN module, to indicate that it has received the last data via Data Block #4. When B3:7/3 has been unlatched, the module is ready for the next data transfer via Data Block #4.



This rung monitors the DTCW and DTSW bits for Data Block #5. When a transfer of data using Data Block #5 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #5 in the 1747-BSN module.



Unlatch the virtual DTCW bit for Data Block #5 when the comparable virtual DTSW bit is set by the 1747-BSN module to indicate that it has received the last data via Data Block #5. When B3:7/4 has been unlatched, the module is ready for the next data transfer via Data Block #5.

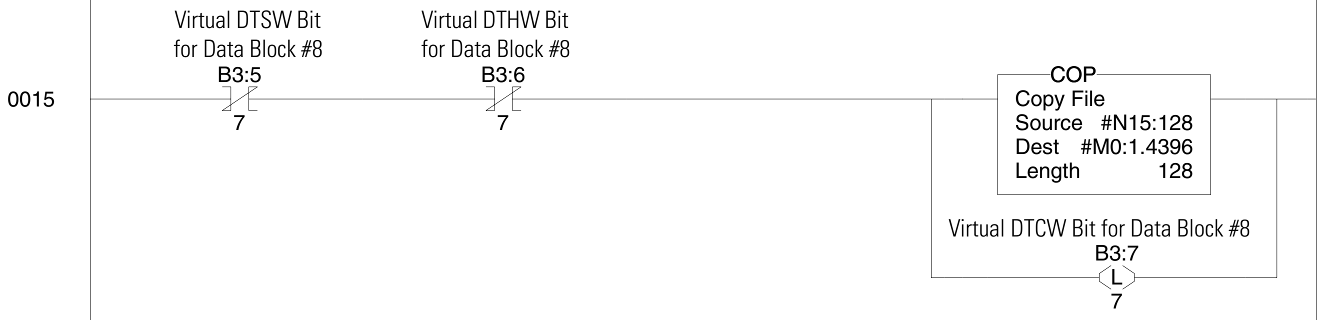
This rung monitors the DTCW and DTSW bits for Data Block #6. When a transfer of data using Data Block #6 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #6 in the 1747-BSN module.

Unlatch the virtual DTCW bit for Data Block #6 when the comparable virtual DTSW bit is set by the 1747-BSN module to indicate that it has received the last data via Data Block #6. When B3:7/5 has been unlatched, the module is ready for the next data transfer via Data Block #6.

This rung monitors the DTCW and DTSW bits for Data Block #7. When a transfer of data using Data Block #7 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #7 in the 1747-BSN module.

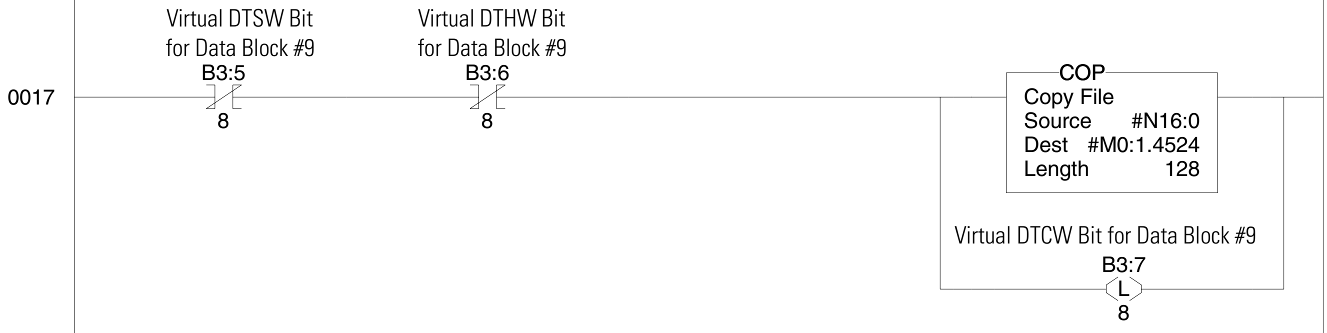
Unlatch the virtual DTCW bit for Data Block #7 when the comparable virtual DTSW bit is set by the 1747-BSN module to indicate that it has received the last data via Data Block #7. When B3:7/6 has been unlatched, the module is ready for the next data transfer via Data Block #7.

This rung monitors the DTCW and DTSW bits for Data Block #8. When a transfer of data using Data Block #8 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #8 in the 1747-BSN module.



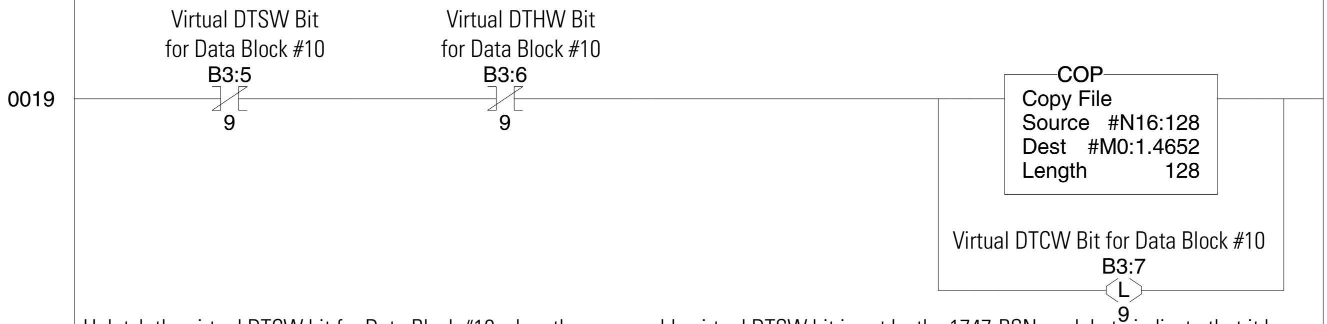
Unlatch the virtual DTCW bit for Data Block #8 when the comparable virtual DTSW bit is set by the 1747-BSN module to indicate that it has received the last data via Data Block #8. When B3:7/7 has been unlatched, the module is ready for the next data transfer via Data Block #8.

This rung monitors the DTCW and DTSW bits for Data Block #9. When a transfer of data using Data Block #9 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #9 in the 1747-BSN module.



Unlatch the virtual DTCW bit for Data Block #9 when the comparable virtual DTSW bit is set by the 1747-BSN module to indicate that it has received the last data via Data Block #9. When B3:7/8 has been unlatched, the module is ready for the next data transfer via Data Block #9.

This rung monitors the DTCW and DTSW bits for Data Block #10. When a transfer of data using Data Block #10 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #10 in the 1747-BSN module.

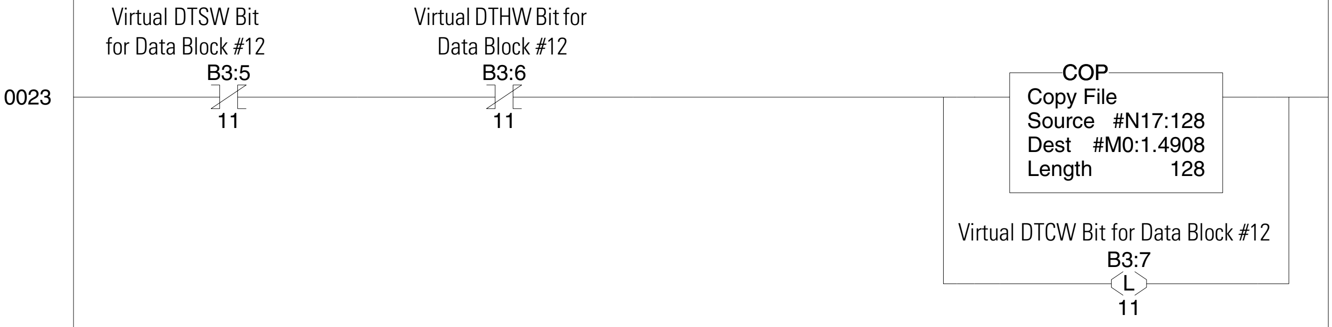


Unlatch the virtual DTCW bit for Data Block #10 when the comparable virtual DTSW bit is set by the 1747-BSN module to indicate that it has received the last data via Data Block #10. When B3:7/9 has been unlatched, the module is ready for the next data transfer via Data Block #10.

This rung monitors the DTCW and DTSW bits for Data Block #11. When a transfer of data using Data Block #11 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #11 in the 1747-BSN module.

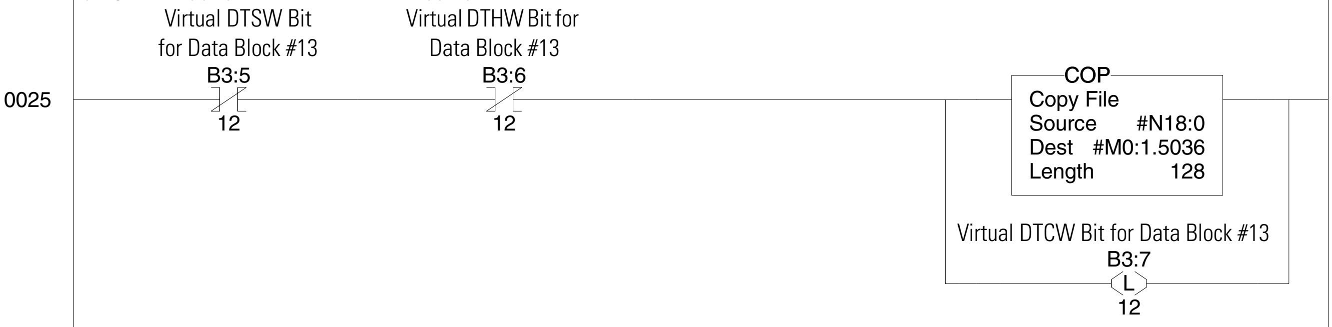
Unlatch the virtual DTCW bit for Data Block #11 when the comparable virtual DTSW bit is set by the 1747-BSN module to indicate that it has received the last data via Data Block #11. When B3:7/10 has been unlatched, the module is ready for the next data transfer via Data Block #11.

This rung monitors the DTCW and DTSW bits for Data Block #12. When a transfer of data using Data Block #12 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #12 in the 1747-BSN module.



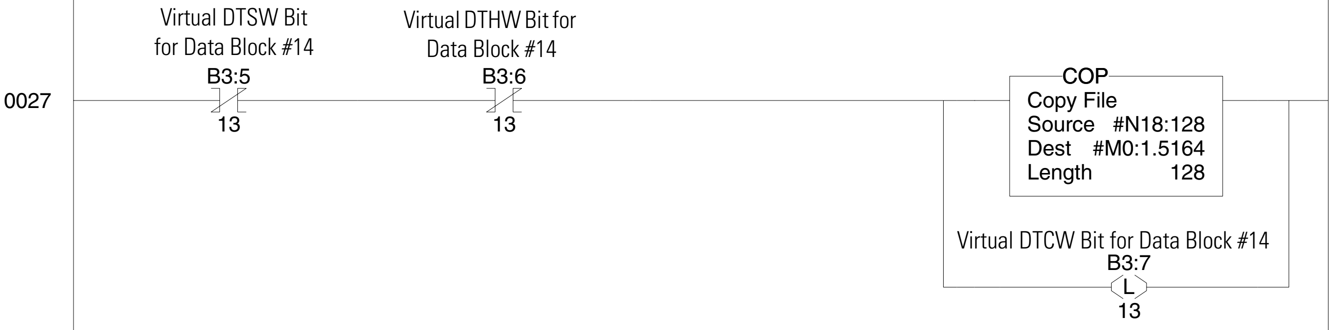
Unlatch the virtual DTCW bit for Data Block #12 when the comparable virtual DTSW bit is set by the 1747-BSN module to indicate that it has received the last data via Data Block #12. When B3:7/11 has been unlatched, the module is ready for the next data transfer via Data Block #12.

This rung monitors the DTCW and DTSW bits for Data Block #13. When a transfer of data using Data Block #13 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #13 in the 1747-BSN module.

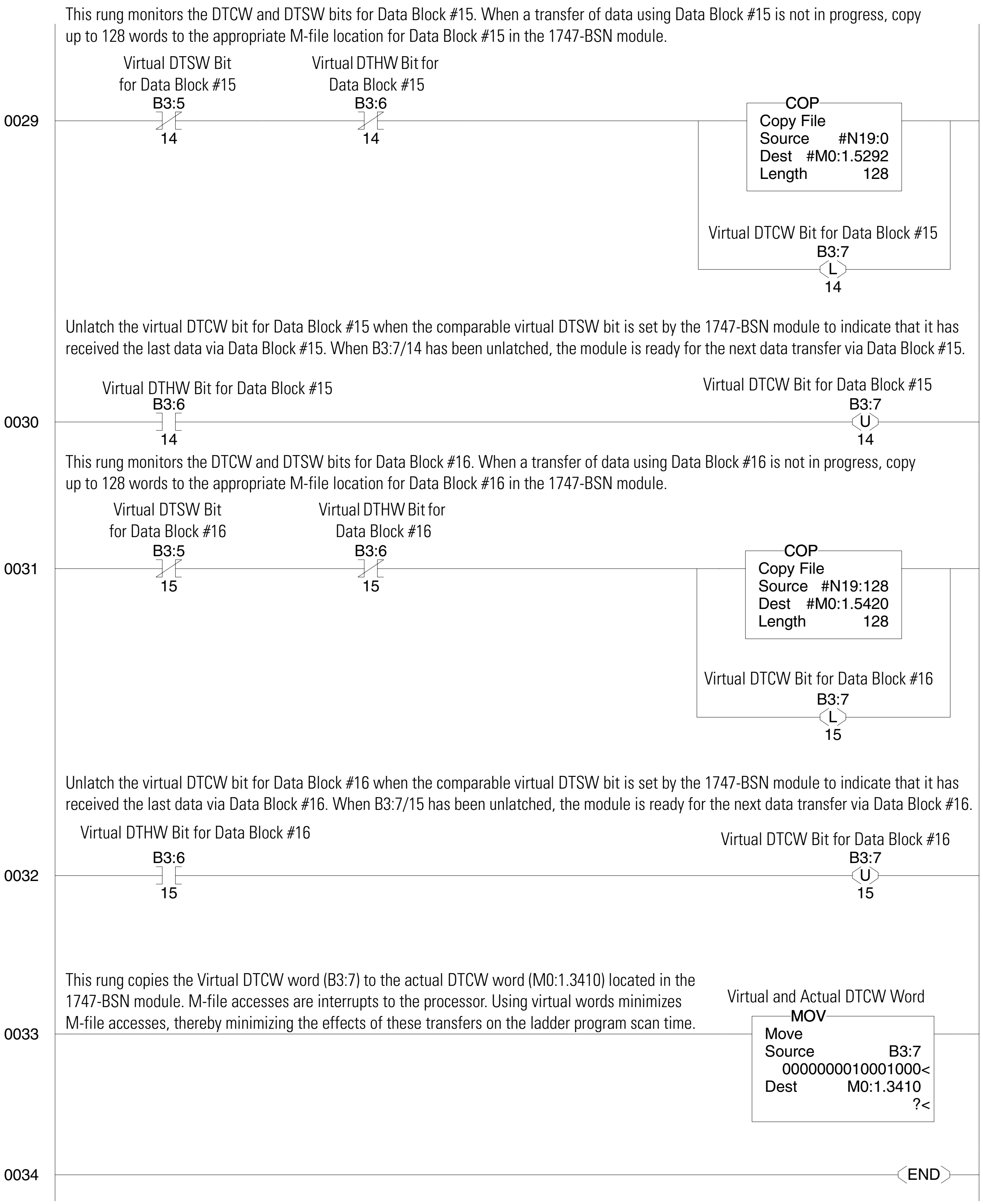


Unlatch the virtual DTCW bit for Data Block #13 when the comparable virtual DTSW bit is set by the 1747-BSN module to indicate that it has received the last data via Data Block #13. When B3:7/12 has been unlatched, the module is ready for the next data transfer via Data Block #13.

This rung monitors the DTCW and DTSW bits for Data Block #14. When a transfer of data using Data Block #14 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #14 in the 1747-BSN module.



Unlatch the virtual DTCW bit for Data Block #14 when the comparable virtual DTSW bit is set by the 1747-BSN module to indicate that it has received the last data via Data Block #14. When B3:7/13 has been unlatched, the module is ready for the next data transfer via Data Block #14.



This rung monitors the DTCW and DTSW bits for Data Block #15. When a transfer of data using Data Block #15 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #15 in the 1747-BSN module.

Unlatch the virtual DTCW bit for Data Block #15 when the comparable virtual DTSW bit is set by the 1747-BSN module to indicate that it has received the last data via Data Block #15. When B3:7/14 has been unlatched, the module is ready for the next data transfer via Data Block #15.

This rung monitors the DTCW and DTSW bits for Data Block #16. When a transfer of data using Data Block #16 is not in progress, copy up to 128 words to the appropriate M-file location for Data Block #16 in the 1747-BSN module.

Unlatch the virtual DTCW bit for Data Block #16 when the comparable virtual DTSW bit is set by the 1747-BSN module to indicate that it has received the last data via Data Block #16. When B3:7/15 has been unlatched, the module is ready for the next data transfer via Data Block #16.

This rung copies the Virtual DTCW word (B3:7) to the actual DTCW word (M0:1.3410) located in the 1747-BSN module. M-file accesses are interrupts to the processor. Using virtual words minimizes M-file accesses, thereby minimizing the effects of these transfers on the ladder program scan time.

Data Transfer Schemes

This manual provides two possible methods for transferring data from a primary to a secondary processor:

The first programming alternative transfers a data block per program scan. With this method, the application program defines the order that is used for transferring the data blocks. This programming scheme results in a lower overall data transfer throughput and has a minimal impact in the program scan time.

The second programming alternative transfers each data block at its maximum possible throughput. This method is shown in the ladder program example earlier in this chapter. With this method, the application program tries to make all the possible data transfers in each program scan, according to the DTSW bits.

Data Transfer Method 1

With this alternative, the primary SLC 5/0x uses a counter to select the 1747-BSN module to which the data block is sent. The primary SLC 5/0x also uses a second counter to select which block is sent to the 1747-BSN module. If the system has only one 1747-BSN, only one counter is used.

The following procedure is used in the primary SLC 5/0x for sending the data blocks:

1. Read all the DTSWs from the respective 1747-BSN modules.

2. Clear all DTCW bits corresponding to the data blocks that are marked as already transferred to secondary system in the DTSWs. Send the data blocks to their respective 1747-BSN modules.

3. Look at the counters and DTSWs to check whether the next data block that is sent is able to receive new data from the SLC 5/0x:
   - If the next data block is not able to receive new data, increment the data block counter and check whether it is time to select the next 1747-BSN module.
   - If the next data block is able to receive new data, increment the module counter, restart the data block counter and repeat step 3. (Repeat this step until all possible data blocks are scanned.)

4. Copy the data block from the SLC 5/0x to the 1747-BSN module indicated by the module counter.

5. Advise the 1747-BSN module that the data block is ready through the DTCW (set the bit corresponding to the data block in this word).

6. Increment the data block counter and check whether it is time to select the next 1747-BSN module. If it is time to select the next module, increment the module counter and restart the data block counter.

7. Start step 1 in the next program scan.

To receive the data blocks, the secondary SLC 5/0x application program uses the following procedure:

1. Read all the DTSWs from the respective 1747-BSN modules.

2. Clear all DTHW bits corresponding to the data blocks that were previously read and that no longer have a data ready bit set for them in the DTSWs. Send all the DTHWs to their respective 1747-BSN modules.

3. Look at the counters and DTSWs to check whether the next data block that is received is ready in the 1747-BSN module
    - If the next data block is not ready, increment the data block counter and check whether it is time to select the next 1747-BSN module.
    - If the next data block is ready, increment the module counter, restart the data block counter and repeat step 3. (Repeat this step until all possible data blocks are scanned.)

4. Copy the data block from the 1747-BSN module to the SLC 5/0x.

5. Advise the 1747-BSN module that the data block was already read through the DTHW (set the bit corresponding to the data block in this word).

6. Increment the data block counter and check whether it is time to select the next 1747-BSN module. If it is time to select the next module, increment the module counter and restart the data block counter.

7. Start step 1 in the next program scan.

The worst scenario with this transfer method is a system that has eight 1747-BSN modules in which each one is transferring 16 data blocks. The total number of data blocks is 128 and the backup system spends 128 program scans to transfer all the data blocks.

Data Transfer Method 2

With method 2, the application program in the SLC 5/0x considers each data block independent from the other data blocks. This is shown in the ladder program example earlier in this chapter for one set of 1747-BSN modules. Reproduce this logic for additional sets of 1747-BSN modules, being sure to update the slot number in the M-file addresses for each set. The following procedure is used in the primary SLC 5/0x for sending the data blocks:

1. Read all the DTSWs from the respective 1747-BSN modules.
2. Clear all DTCW bits corresponding to the data blocks that are marked as already transferred to secondary system in the DTSWs. Send the data blocks to their respective 1747-BSN modules.
3. For each data block, do the following:
   - Check whether the block is free to receive new data from the SLC 5/0x.
   - Copy the data block from the SLC 5/0x to the 1747-BSN module.
   - Advise the 1747-BSN module that the data block is ready through the DTCW (set the bit corresponding to the data block in this word).
4. Start step 1 in the next program scan.

To receive the data blocks, the secondary SLC 5/0x application program uses the procedure below:

1. Read all the DTSWs from the respective 1747-BSN modules.
2. Clear all DTHW bits corresponding to the data blocks that were previously read and that no longer have a data ready bit set for them in the DTSWs. Send all the DTHWs to their respective 1747-BSN modules. For each data block, do the following:
   - Check whether the data block is ready in the 1747-BSN module.
   - Copy the data block from the 1747-BSN to the SLC 5/0x.
   - Advise the 1747-BSN module that the data block was already read through the DTHW (set the bit corresponding to the data block in this word).
3. Start step 1 in the next program scan.

The worst scenario with this transfer method is a system with eight 1747-BSN modules in which each one is transferring 16 data blocks.

The total number of data blocks is 128. In each program scan, the system transfers all 128 data blocks. Total transfer time is approximately 700 ms.

Other transfer methods

Data transfer methods 1 and 2 were designed to be used as sample methods. In a real application, the user would develop other methods of transferring data. These alternatives would optimize the data transfer throughput without causing an excessive loss of time due to the M-file transfers.

Accounting for Instructions That Could Cause Problems During Switchover

Some instructions may operate unpredictably when a switchover occurs if you fail to observe certain programming considerations. For example, you must consider transferring the control element and data elements for instructions that are also being executed in the secondary processor. You also must consider transferring data files which contain more than one word.

The following instructions are discussed in this section:

- timer
- counter
- logical, arithmetic, compare and move
- diagnostic, sequencing, file
- file copy and fill
- FIFO
- block transfer
- message

In the remainder of this chapter, we describe the behavior of these instructions in the SLC 500 backup system.

IMPORTANT The primary and secondary systems do not necessarily execute the same instructions at the same time (the systems' program scans are not synchronized). The SLC 500 processor executes block transfers to the program scan asynchronously. Therefore, the processor interrupts the program scan asynchronously to access block transfer write (BTW) and block transfer read (BTR) files.

Timer Instructions

The SLC 500 processors maintain timers by keeping a copy of a hardware timer in a portion of the three-word structure (timer byte) used by each timer.

Control byte	Timer byte	1
Preset value Accumulated value		2
Accumulated value		3

The hardware timer of the primary processor is completely asynchronous to the hardware timer of the secondary processor. If the whole three-word timer structure, including the timer-byte, is transferred from the primary to the secondary processor during a switchover, then the timer could encounter a large positive increment in the timer accumulated value. The worst-case increment may be as much as 2.55 seconds.

Follow the guidelines below to help avoid this type of increment in the timer accumulated value:

- Do not send the first word of the timer control structure.
- Transfer only the accumulated value. (Transfer the preset value also, if you need to change it.)

Even though both processors are executing the same programs, your program should transfer the timer accumulated value from the primary processor to the secondary processor at least once after you:

- start the backup system
- switch the secondary SLC 500 from PROGRAM to RUN mode
- restart a repaired system

This allows the accumulated value in the secondary processor to track the accumulated value in the primary processor.

Counter Instructions

Counter instructions are similar in structure to timer instructions. However, when a switchover occurs, a counter that occurs once in a program could increment or decrement twice in one program scan. This problem arises when the following sequence of events occur:

1. The primary system executes a counter up/down instruction conditioned by a rung transition of not-true to true. This increments/decrements the counter accumulated value.
2. The 1747-BSN modules transfer this data into the counter file of the backup processor before the secondary's input image table is updated with the input transition information.
3. The backup processor updates its input image table.
4. The processor in the secondary system (which was at a different point in its program scan) executes the counter instruction again, thus incrementing/decrementing the counter again.

Programming Techniques

As with the timer instruction, your program should transfer the counter accumulated value from the primary to the secondary processor at least once after you:

- start the backup system
- switch the secondary SLC 500 from PROGRAM to RUN mode
- restart a repaired system

This allows the accumulated value in the secondary processor to track the accumulated value in the primary processor.

Diagnostic, Sequencing, File Copy and Fill Instructions

The type of data you are transferring, as well as how you transfer the data to the secondary processor, determines the behavior of the following file instructions:

- File Copy (COP)
- File Fill (FLL)
- Diagnostic (FBC and DDT)

You should place the instruction's control element data file within the same 62 words of data. If this is not done, the 1747-BSN module could transfer the control values responsible for control of the file at a different point than when it transfers the file. A time lag between when the file is updated and when the associated control element is updated in the secondary processor. As a result, the following may occur during switchover:

- a file operation that was running may be off by one word,
- a word could be operated on by the file twice, or
- a word could not be operated on at all.

FIFO Instructions

The type of data you are transferring, as well as how you transfer the data to the secondary processor, determines the behavior of the FIFO instructions.

If you are transferring only selected areas of the data files which do not include data for FIFO and bit instructions, the instructions are executed normally.

However, if you are transferring all of the data table files over the HSSL, you may encounter problems. For example, if you are shifting four bits of BCD data through several words bit by bit, the data table contains invalid data until each shift is complete. Should the 1747-BSN module transfer one or more of these values to the backup system and the primary system fails, the data table of the backup system contains intermediate values for an indefinite amount of time. The result is illegal BCD values.

Block Transfer Instructions

It is impossible to guarantee that both processors are executing a given BTW or BTR at the same time. This is because the remote I/O scans are not synchronized and the block transfer data is not transferred instantaneously by both SLC 500 processors. The BTR files in the secondary processor always reflect information contained in the primary processor, except with a time delay.

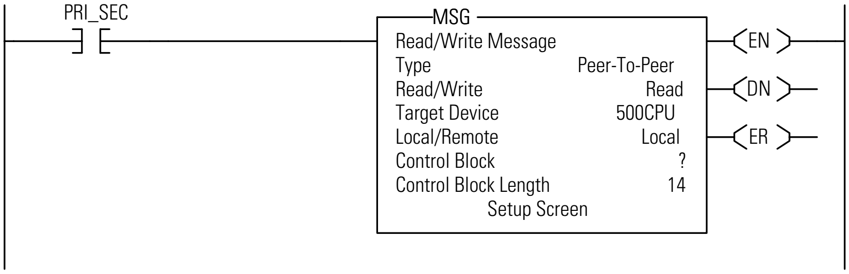
To avoid the possibility of block transfer data changing during the scan of the program, you can buffer data at the beginning of the ladder program to ensure continuity.

Another potential problem with block transfer instructions is that during switchover, the secondary scanner may skip one or more remote adapters. This may happen because the “new” primary scanner (previously the secondary) was in a different place in its remote I/O chassis scan.

Message Instructions

When programming, you must ensure that when a switchover occurs, that any messages which were running in the primary processor are subsequently enabled in the secondary processor. You can do this by conditioning message instruction rungs in both the primary and secondary processors using the primary/secondary bit of the 1747-BSN status word. If necessary, you can pass the results of a message addressed to the primary processor over the HSSL to the secondary processor.

Refer to the SLC 500 Instruction Set Reference Manual, publication number 1747-RM001, for more information on the Message instruction.



PID Control Files

When using PID instructions, you can send all or part of these control files, based on how you back up your PID instructions and the total amount of data to be sent by the 1747-BSN modules. Keep in mind that the control files contain such things as your setpoint, gains, and words that are used by the PID instruction for internal storage and should not be manipulated unnecessarily.

Summary of Programming Considerations

When developing a program for the SLC 500 backup system, you must always consider the following:

- non-synchronous I/O scans
- execution times of block transfer instructions
- time for transfer of data from the primary to the secondary processor
- the need to reduce the quantity of data to be transferred, and gathering all related data into a single block
- synchronization of discrete I/O data with regard to program scans even though remote I/O scans are performed asynchronously
- the need to disable message instructions in the secondary processor using the primary/secondary bit

Specifications

Backup Scanner Operating Specifications

This appendix provides system specifications, as well as throughput information, for the Backup Scanner Module. Topics include:

- scanner operating specifications
- network specifications
- throughput introduction
- calculating throughput

Backplane Current Consumption	800 mA at 5V
Operating Temperature	+32°F to +140°F (0°C to +60°C)
Storage Temperature	-40°F to +185°F (-40C to +85°C)
Humidity	5 to 95% without condensation
Noise Immunity	NEMA Standard ICS 2-230
Agency Certification (when product or packaging is marked)	UL listed C-UL listed - Class I, Division 2, Groups A, B, C, D Temp. Code T3C CE compliant for all applicable directives C-Tick marked for all applicable acts

Network Specifications

Baud Rate Determination of Maximum Cable Length and Terminating Resistor Size

	Baud Rate	Max. Cable Distance (Belden™ 9463)	Resistor Size
Using Extended Node Capability	57.6K baud	3048 m (10,000 ft.)	82Ω 1/2 Watt Gray-Red-Black-Gold
	115.2K baud	1524 m (5,000 ft.)	
	230.4K baud	762 m (2,500 ft.)	
Not Using Extended Node Capability	57.6K baud	3048 m (10,000 ft.)	150Ω 1/2 Watt Brown-Green-Brown-Gold
	115.2K baud	1524 m (5,000 ft.)	
	230.4K baud	762 m (2,500 ft.)	82Ω 1/2 Watt Gray-Red-Black-Gold

DIP Switch Position for Baud Rate Selection

Position 1	Position 2	Baud Rate
ON	ON	57.6K
ON	OFF	115.2K
OFF	ON	230.4K
OFF	OFF	Disabled

Throughput Introduction

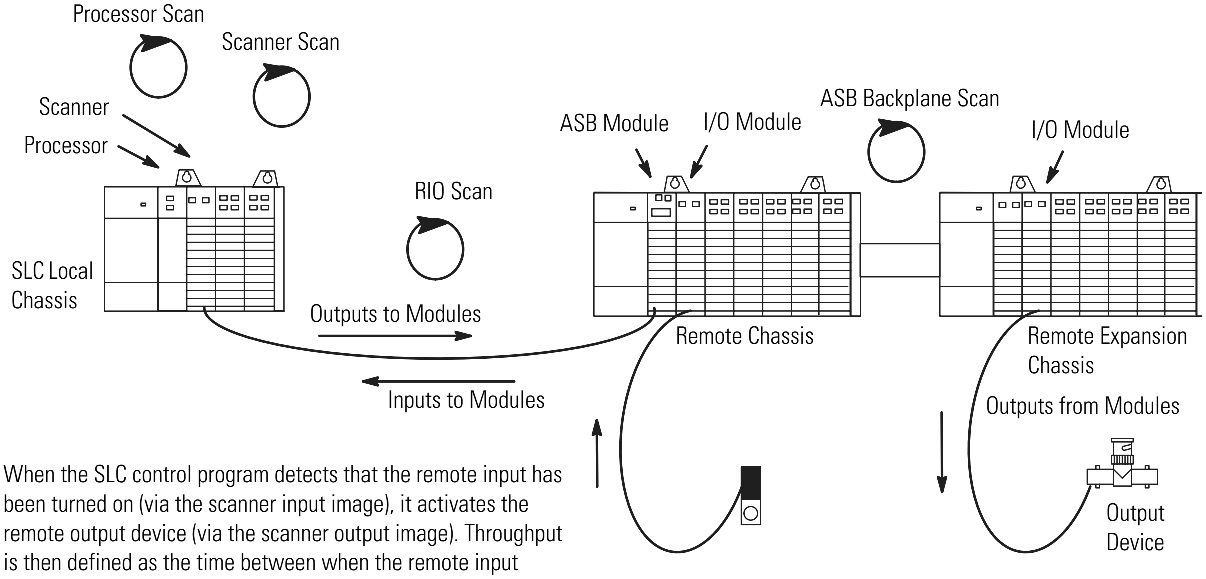
RIO throughput is defined as the time between when an input event occurs at an I/O module in an RIO chassis to when an output event occurs at an I/O module within the same RIO chassis. There are three types of throughput concerning the 1747-BSN Backup Scanner Module and its RIO network:

- discrete throughput (time from discretely mapped input to discretely mapped output) *without block transfers (BTs) present*
- discrete throughput (time from discretely mapped input to discretely mapped output) *with BTs present*
- BT throughput (time from when a BT is enabled to when the BT successfully completes)

RIO Network Throughput Components

The following components affect RIO network throughput:

- the total SLC processor scan time
- the total RIO link scan time
- adapter(s) backplane scan time(s)
- the backup scanner's output delay time
- the backup scanner's input delay time
- input module delay times
- output module delay times



When the SLC control program detects that the remote input has been turned on (via the scanner input image), it activates the remote output device (via the scanner output image). Throughput is then defined as the time between when the remote input device is activated to when the remote output turns on.

Calculating Throughput

The 1747-BSN Backup Scanner's throughput is determined by using the formulas provided in this section.

Discrete I/O Throughput without Block Transfers (T_{dm-nbt}) Present

The information in this section is used to calculate the discrete throughput of the 1747-BSN Backup Scanner if there are no BTs occurring on the RIO link to any chassis.

If BTs are present on the RIO link you *must* use the Discrete I/O Throughput with Block Transfers (T_{dm-bt}) Present section to determine your throughput.

The formula to calculate the maximum backup scanner discrete I/O throughput without BTs present is:

$T_{dm\text{-}nbt} = 2T_{ps} + 2T_{RIO} + T_{adp} + T_{SNo} + T_{SNi} + T_{id} + T_{od}$

$T_{dm\text{-}nbt}$ = The maximum discrete throughput *without* BTs in milliseconds (ms)

To calculate $T_{dm\text{-}nbt}$ throughput, substitute values for the variables in the formula above. Locate these values in the following documents:

Variable	**Variable Description**	**Location of Variable**
T_{ps}	The total processor scan time (ms)	Measured or estimated
T_{RIO}	The total RIO scan time (ms)	See the section *RIO Scan Time Calculation* (T_{RIO}) on page A-5.
T_{adp}	The adapter throughput delay. For a 1747-ASB, this is two ASB backplane scan times.	adapter user manual
T_{SNo}	The backup scanner module output delay time (ms)	See the section *Backup Scanner Output Delay Time* (T_{SNo}) on page A-14.
T_{SNi}	The backup scanner module input delay time (ms)	5 ms (constant value for all formulas in this appendix)
T_{id}	The input module delay time (ms)	I/O product data and I/O instruction sheets
T_{od}	The output module delay time (ms)	I/O product data and I/O instruction sheets

RIO Scan Time Calculation (T_{RIO})

The RIO scan time is calculated by identifying the baud rate and image size of *each* logical device on the RIO link. Locate the corresponding time value in the following table. If you are using multiple logical devices, add the time values together to determine the total RIO scan time (T_{RIO}).

$T_{RIO} = T_{adapter\ 1} + T_{adapter\ 2} + T_{adapter\ 3}$

RIO Scan Times for Adapters

Adapter Size	Baud Rate		
	57.6K	**115.2K**	**230.4K**
1/4 logical rack	6.0 ms	3.5 ms	2.5 ms
1/2 logical rack	6.5 ms	4.0 ms	2.75 ms
3/4 logical rack	7.5 ms	4.5 ms	3.0 ms
Full logical rack	9.5 ms	5.5 ms	3.5 ms

Example Discrete I/O Throughput without Block Transfers Present

An SLC 5/03 is controlling an RIO link running at 115.2K baud that has the following adapters:

- One 1747-ASB module is configured as a 1/2 logical rack starting at logical rack 0.
  I/O chassis slot 1 contains 1746-IB16, 16 point input module
  I/O chassis slot 2 contains 1746-OB16, 16 point output module
- Two adapters are each configured as full logical racks (logical racks 1 and 2).
- Three adapters are each configured as 1/4 logical racks (logical rack 3).

You need to calculate your $T_{dm\text{-}nbt}$: the RIO throughput time from when the input closes on the 1746-IN16 until the output on the 1746-OB16 is on.

1. Use the throughput formula to calculate the maximum throughput.

   $T_{dm\text{-}nbt} = 2T_{ps} + 2T_{RIO} + T_{adp} + T_{SNo} + T_{SNi} + T_{id} + T_{od}$

   $T_{ps} = 25.0$ ms

   T_{RIO} = The total RIO scan time (ms)

   T_{SNo} = See value in the table on page B-13, T_{SNo} without M0 File Writes (Normal Mode).

   $T_{SNi} = 5.0$ ms

   $T_{id} = 10.0$ ms, which is from I/O module instruction sheets

   $T_{od} = 1.0$ ms, which is from I/O module instruction sheets

   $T_{dm\text{-}nbt} = 2(25.0) + 2T_{RIO} + 8.0 + T_{SNo} + 5.0 + 10.0 + 1.0$

2. Calculate the total RIO scan time (T_{RIO}). Locate the baud rate (115.2K) and adapter size, which is found in the table on page B-4. Multiply the RIO scan times listed under the 115.2K heading by the number of each different type of rack that you have. Add those numbers together:

   $T_{RIO} = T_{adapter\ 1} + T_{adapter\ 2} + T_{adapter\ 3}$

   $T_{RIO} = 1(4.0\text{ ms}) + 2(5.5\text{ ms}) + 3(3.5\text{ ms})$

   $T_{RIO} = 25.5$ ms

3. Find T_{SNo} on page B-13 in the table T_{SNo} without M0 File Writes (Normal Mode). For this example $T_{upd} > T_{hold}$, and there are 4 logical racks configured. Therefore:

   $T_{SNo} = 7.0$ ms

4. Substitute all the values for variables in the throughput formula and solve for throughput:

   $T_{dm\text{-}nbt} = 2T_{ps} + 2T_{RIO} + 2T_{bp} + T_{SNo} + T_{SNi} + T_{id} + T_{od}$

   $T_{dm\text{-}nbt} = 2(25.0) + 2(25.5) + 8.0 + 7.0 + 5.0 + 10.0 + 1.0$

   $T_{dm\text{-}nbt} = 132.0$ ms = maximum throughput

Discrete I/O Throughput *with* Block Transfers ($T_{dm\text{-}bt}$) Present

The information in this section is used to calculate the discrete throughput of the 1747-BSN Backup Scanner if there are BTs occurring on the RIO link to *any* chassis.

If BTs are not present on the RIO link, you *must* use the Discrete I/O Throughput *without* Block Transfers (Tdm-nbt) Present section to determine your throughput. See page B-3.

The formula to calculate discrete I/O throughput with BTs present is:

$$T_{dm\text{-}bt} = 2T_{ps} + 2T_{RIO} + 2T_{btx} + T_{adp} + T_{SNo\text{-}bt} + T_{SNi} + T_{id} + T_{od}$$

$T_{dm\text{-}bt}$ = The maximum discrete throughput with BTs in milliseconds (ms)

To calculate $T_{dm\text{-}bt}$ throughput, substitute values for the variables in the formula above. Locate these values in the following documents:

Variable	Variable Description	Location of Variable
T_{ps}	The total processor scan time (ms)	measured or estimated
T_{RIO}	The total RIO scan time (ms)	see the section RIO Scan Time Calculation (T_{RIO}) on page A-5
T_{btx}	Additional time due to sending any BT data on the RIO link.	see the section Determining T_{btx} on page A-8
T_{adp}	The adapter throughput delay. For a 1747-ASB, this is two ASB backplane scan times.	adapter user manual
$T_{SNo\text{-}bt}$	Backup scanner output delay time with BTs present	see the section Determining $T_{SNo\text{-}bt}$ on page A-8
T_{SNi}	The backup scanner module input delay time (ms)	5 ms (constant value for all formulas in this appendix)
T_{id}	The input module delay time (ms)	I/O product data and I/O instruction sheets
T_{od}	The output module delay time (ms)	I/O product data and I/O instruction sheets

Determining $T_{SNo\text{-}bt}$

Use the following table to find $T_{SNo\text{-}bt}$ for your particular configuration.

IMPORTANT The times shown are, to the best of our knowledge, the maximum delay times of the backup scanner. However, in instances that throughput is an important consideration, test the application thoroughly first to ensure proper operation. Note that in most situations the average throughput is much better than the calculated maximum throughput.

Number of Logical Racks Configured	Normal Mode	Complementary Mode		
	All Baud Rates	57.6K baud	115.2K baud	230.4K baud
1 Logical Rack	16.0	19.0	24.0	32.0
2 Logical Racks	19.0	23.0	27.0	36.0
3 Logical Racks	22.0	26.0	30.0	39.0
4 Logical Racks	25.0	28.0	34.0	42.0

Determining T_{btx}

Before determining (T_{btx}), you need to establish the maximum BT write or read length that is to be processed by each logical rack on the RIO link. RIO scan time is increased each time an BT is sent to any logical device on the RIO network. The scan time increase depends on the number of words sent in the BT and the selected baud rate.

RIO link protocol allows for a maximum of one BT to be sent to each logical rack on the RIO link during any single RIO scan. Therefore, if multiple BTs are sent to devices within the same logical rack, only the longest BT to that logical rack needs to be considered to determine your maximum throughput. The RIO scan time increase (T_{ri}) for each logical rack is:

Baud Rate	RIO Scan Time Increase (T_{ri})
57.6K baud	0.300 x BT length + 5.0 ms
115.2K baud	0.150 x BT length + 3.5 ms
230.4K baud	0.075 x BT length + 2.0 ms

The total increase in the RIO scan time (T_{btx}) is equal to:

T_{btx} = sum of T_{ri} for all logical racks

Example Discrete I/O Throughput with Block Transfers Present

An SLC 5/03 is using a backup scanner to control a 115.2K baud RIO link that has 3 adapters and 4 logical devices.

Adapter #1 (1747-ASB module):

- starting logical rack 0, logical group 0
- 12 logical groups (1 1/2 logical racks)
- one 8 word and two 4 word BT write/read modules in logical rack 0
- one 2 word BT write/read module in logical rack 1

Adapter #2 (1771-ASB module):

- starting logical rack 2, logical group 0
- 2 logical groups (1/4 logical rack)
- one 64 word BT write/read module

Adapter #3 (1771-ASB module):

- starting logical rack 2, logical group 2
- 2 logical groups (1/4 logical rack)
- one 32 word BT write/read module

1. Use the throughput formula to calculate the maximum throughput of the 1747-ASB module.

   $T_{dm\text{-}bt} = 2T_{ps} + 2T_{RIO} + 2T_{btx} + T_{adp} + T_{SNo\text{-}bt} + T_{SNi} + T_{id} + T_{od}$

   T_{ps} = 25.0 ms, which is from the APS reference manual (assume for example)

   T_{RIO} = The total RIO scan time (ms)

   T_{btx} = Additional time due to sending any BT data on the RIO link

   T_{adp} = Two 1747-ASB module backplane scan times (calculated from ASB manual = 2(4.5) = 9.0 ms

   $T_{SNo\text{-}bt}$ = 22.0 ms from the table on page B-13, TSNo with Block Transfers (Normal Mode). There are 3 logical racks configured.

   T_{SNi} = 5.0 ms

   T_{id} = 10.0 ms, which is from I/O module instruction sheets

   T_{od} = 1.0 ms, which is from I/O module instruction sheets

   $T_{dm\text{-}bt} = 2(25.0) + 2T_{RIO} + 2T_{btx} + 9.0 + 22.0 + 5.0 + 10.0 + 1.0$

2. Calculate the total RIO scan time (T_{RIO}). Locate the baud rate (115.2K) and adapter size which is found in the table on page B-4. Multiply the RIO scan times listed under the 115.2K heading by the number of each different type of rack that you have. Add those number together.

   T_{RIO} = Tadapter1 + Tadapter2 + Tadapter3

   $T_{RIO} = 1(5.5) + 1(4.0) + 2(3.5)$

   T_{RIO} = 16.5 ms

3. Calculate the maximum T_{ri} time for each logical rack. Do this by determining the largest BT that occurs to any device within a logical rack and calculating the transfer time using the table on page A-8. Then add together the T_{ri} times for each logical rack to obtain T_{btx}.

   T_{ri} for rack 0 = 0.150(8) + 3.5 = 4.7 ms (maximum BT to rack 0 is 8 words)

   T_{ri} for rack 1 = 0.150(2) + 3.5 = 3.8 ms (maximum BT to rack 1 is 2 words)

   T_{ri} for rack 2 = 0.150(64) + 3.5 = 13.1 ms (maximum BT to rack 2 is 64 words)

   $T_{btx} = T_{ri0} + T_{ri1} + T_{ri2} = 4.7 + 3.8 + 13.1 = 21.6$ ms

4. Substitute all the values for variables in the throughput formula and solve for throughput.

   $T_{dm\text{-}bt} = 2(25.0) + 2(16.5) + 2(21.6) + 9.0 + 22.0 + 5.0 + 10.0 + 1.0$

   $T_{dm\text{-}bt} = 173.2$ ms = maximum throughput

Block Transfer Throughput

Block transfer throughput is the time from when the BT is enabled via the EN bit, until the DN bit is processed. The following BT timing explanations are based on the directional continuous BT example shown on page 7-19, where a BT is re-triggered automatically upon each completion.

BT throughput is *always* slower than discrete data transfer. Completing a BT is dependent on the time involved for the:

- SLC control program to enable the BT via an M0 file write(1)
- backup scanner to detect that a BT has been requested(2)
- BT to be waiting in the queue due to another BT already being processed on the same logical rack(3)
- backup scanner to schedule a pending bit(2)
- adapter to acknowledge the request(4)
- backup scanner to initiate the BT and transfer the data(2)
- SLC control program to detect that the BT has completed (DN flag set)(1)

The time to free up the BT buffer (by clearing the EN flag so another BT can be performed) depends on the:

- BT instruction time when the rung transitions from true to false, which clears the EN flag(5)
- time for the backup scanner to detect that the EN flag has been cleared(6)
- BT instruction time to detect that the DN flag has been cleared(5)

The formula to calculate BT throughput is:

$$T_{M0} + T_{SNo\text{-}bt}\,(\text{number of BTs} + 1) + T_{btwait} + 2T_{RIO} + 2T_{btx} + T_{adp\text{-}bt} + T_{ps}$$

The equation for freeing up the BT buffer is:

$$T_{M0} + TS_{No\text{-}bt}\,(\text{number of BTs}) + T_{ps}$$

---

(1) This is dependent on the SLC processor you are using.

(2) Refer to the equations that follow.

(3) The RIO network allows only one BT per logical rack (not logical device) per RIO scan. Therefore, if multiple BTs are performed on devices within the same logical rack, BTs have to wait in the queue until any previously scheduled BTs for the same logical rack are completed.

(4) This is dependent on the RIO adapter.

(5) This is dependent on the SLC processor you are using.

(6) Refer to the equations that follow.

Substitute values for the variables in the formulas above. Locate these values in the following documents:

Variable	Variable Description	Location of Variable
T_{M0}	Time to perform M0 file write to enable BT	Appendix B
$T_{SNo\text{-}bt}$	Backup Scanner Output Delay time with BTs present. There must be an output delay time added for each BT buffer that is being used since the backup scanner processes only one BT enable or disable every $T_{SNo\text{-}bt}$ (to minimize the impact on discrete I/O throughput).[1]	see the section Determining $T_{SNo\text{-}bt}$ on page A-8
T_{btwait}	Equals the sum of the throughput times for all BTs scheduled to the same logical rack (time waiting is queue), + $T_{SNo\text{-}bt}$ (time to schedule pending BT). If multiple BTs are not being performed to the same logical rack, this value equals zero.	calculated
T_{RIO}	RIO scan time without BTs	see the section RIO Scan Time Calculation (T_{RIO}) on page A-5
T_{btx}	Amount that the RIO scan time can be increased due to BTs. This includes the time for the backup scanner to initiate the BT and transfer the data.	see the section Determining T_{btx} on page A-8
$T_{adp\text{-}bt}$	Time for the adapter to acknowledge the BT request. For the 1747-ASB, the manual defines this as no more than one (ASB) backplane scan time and two RIO scans. However, the two RIO scans are already included in the above equation so only the ASB scan time needs to be added.	adapter user manual
T_{ps}	One processor scan time may occur before the SLC control program detects that the DN flag has been set or cleared	measured or estimated

(1) When calculating BT throughput, one $T_{SNo\text{-}bt}$ is also required to handle the BT response.

Backup Scanner Output Delay Time (T_{SNo}) Tables

The tables provided in this section show the maximum backup scanner output delay time (T_{SNo}) for specific applications. T_{SNo} is dependent on the following:

- processor scan time, or time between immediate outputs (if no BTs are present)
- number of logical racks configured
- whether normal or complementary I/O mode is selected
- RIO baud rate (if complementary I/O is selected)

Variable	Variable Description
T_{SNo}	The maximum scanner output delay time
T_{upd}	The time between SLC processor output scan updates or immediate output updates
T_{hold}	A constant time threshold that is dependent on your configuration. Refer to the table on page A-15.

T_{SNo} increases if the interval between T_{upd} decreases to the time threshold (T_{hold}). If T_{upd} is less than T_{hold}, then the larger T_{SNo} number must be used. Otherwise, either number may be used.

IMPORTANT The times shown in this section are, to the best of our knowledge, the maximum delay times of the backup scanner. However, in instances that throughput is an important consideration, test the application thoroughly first to ensure proper operation. In most situations the average throughput is much better than the calculated maximum throughput.

Determining the Number of Logical Racks Configured

The number of logical racks configured is determined by the number of racks that contain configured devices. For example, if there are four 1/4 rack devices in logical rack 0 and one full rack device in logical rack 3, there would be two logical racks configured. The number of logical devices on the RIO network affects only $T_{RIO,}$ and only affects T_{SNo} when additional logical racks are used.

When complementary mode is selected, the number of configured racks is also determined by the number of primary or complementary racks configured, but not by both. (The maximum number of configured racks is 4.) That is, if there is a primary rack configured with a corresponding complementary rack, that is considered *one* logical rack. If there is a primary rack configured without a complementary rack (or vice versa), that also is considered *one* logical rack.

T_{SNo} without M0 File Writes

Normal Mode			
Number of Logical Racks Configured	**All Baud Rates**		
	T_{SNo} if $T_{upd} \le T_{hold}$	T_{hold}	T_{SNo} if $T_{upd} > T_{hold}$
1 Logical Rack	5.0	5.0	2.5
2 Logical Racks	7.0	7.0	4.0
3 Logical Racks	9.0	9.0	5.5
4 Logical Racks	11.0	11.0	7.0

M0-M1 Files and G Files

This appendix contains important information about M0-M1 files and G files. The information is general in nature and supplements specific information contained in earlier chapters of this manual. Topics include:

- M0-M1 Files
- G Files

M0 and M1 files are data files that reside in specialty I/O modules only. There is no image for these files in the processor memory. The application of these files depends on the function of the particular specialty I/O module. With respect to the SLC processor (SLC 5/02 or later), the M0 file is a module output file (a write only file) and the M1 file is a module input file (a read only file). The opposite is true for specialty I/O modules, where the M0 file is a read only file, and the M1 file is a write only file.

M0-M1 Files

M0 and M1 files can be addressed in your ladder program and they can also be acted upon by the specialty I/O module - independent of the processor scan. It is important that you keep the following in mind in creating and applying your ladder logic:

IMPORTANT During the processor scan, the ladder program can address M0 and M1 data with bit, word, or file instructions. Each time an M0-M1 file address is encountered in the program, an immediate data transfer to or from the specialty I/O module occurs. The impact these immediate data transfers have on processor scan time is described in the *SLC 500 Instruction Set Reference Manual,* Publication 1747-RM001.

Configuring M0-M1 Files

M0 and M1 files are configured as part of the I/O configuration procedure for the processor file. After you have assigned the specialty I/O module to a slot (the procedure is the same as assigning other modules), you may click on the advanced configuration button in order to configure the number of M0 and M1 file words.

The specialty I/O module may also require that you configure the G file and specify an ISR (interrupt subroutine) number. G files are discussed on page B-9.

Addressing M0-M1 Files

The addressing format for M0 and M1 files is below:

	Mf:e.s/b		
Where	M	=	module
	f	=	file type (0 or 1)
	e	=	slot (1-30)
	s	=	word (0 to max. supplied by module)
	b	=	bit (0-15)

Using M0-M1 Data File Addresses

M0 and M1 data file addresses can be used in all instructions except the OSR instruction and the instruction parameters noted below:

Instruction	Parameter (uses file indicator #)
BSL, BSR	File (bit array)
SQO, SQC, SQL	File (sequencer file)
LFL, LFU	LIFO (stack)
FFL, FFU	FIFO (stack)

M0/M1 Monitoring

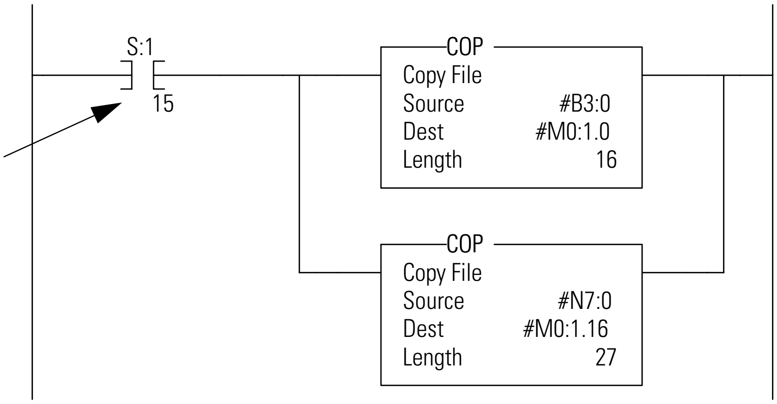
IMPORTANT This option is not supported by the SLC 5/02 processor.

The SLC 5/03, SLC 5/04, SLC 5/05 processors allow you to monitor the actual state of each addressed M0/M1 address (or data table). The highlighting appears normal when compared to the other processor data files. The processor's performance will be degraded to the degree of M0/M1 referenced screen data. For example, if your screen has only one M0/M1 element, degradation is minimal. If your screen has 69 M0/M1 elements, degradation is significant.

Transferring Data Between Processor Files and M0 or M1 Files

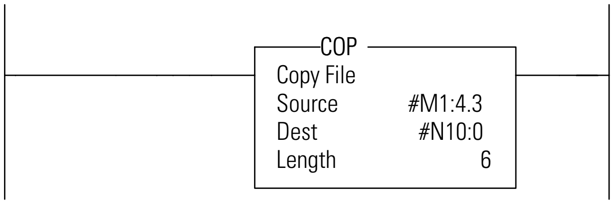
The processor does not contain an image of the M0 or M1 file. As a result, you must edit and monitor M0 and M1 file data via instructions in your ladder program. For example, you can copy a block of data from a processor data file to an M0 or M1 data file or vice versa using the COP instruction in your ladder program.

The COP instructions below copy data from a processor bit file and integer file to an M0 file. For the example, assume the data is configuration information affecting the operation of the specialty I/O module.



First scan bit. It makes this rung true only for the first scan after entering Run mode.

The COP instruction below copies data from an M1 data file to an integer file. This technique is used to monitor the contents of an M0 or M1 data file indirectly, in a processor data file.



Access Time

During the program scan, the processor must access the specialty I/O card to read/write M0 or M1 data. This access time must be added to the execution time of each instruction referencing M0 or M1 data. For the SLC 5/03 and SLC 5/04 processors, the instruction types vary in their execution times.

The following table shows approximate access times per instruction or word of data for the SLC 5/02, SLC 5/03, and SLC 5/04 processors.

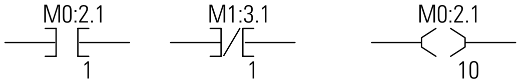
Processor	**Instruction Type**	**Access Time per Bit Instruction or Word of Data**	**Access Time per Multi-Word Instruction**
SLC 5/02 Series B	All types(1)	1930 µs	1580 µs plus 670 µs per word
SLC 5/02 Series C	All types(1)	1160 µs	950 µs plus 400 µs per word
SLC 5/03 (All Series)	XIC or XIO	782 µs	—
	OTU, OTE, or OTL	925 µs	—
	COP to M file	—	772 µs plus 23 µs per word
	COP from M file	—	760 µs plus 22 µs per word
	FLL	—	753 µs plus 30 µs per word
	MVM to M file	894 µs	—
	any source or Destination M file address	730 µs	—

Processor	**Instruction Type**	**Access Time per Bit Instruction or Word of Data**	**Access Time per Multi-Word Instruction**
SLC 5/04 SLC 5/05	XIC or XIO	743 µs	—
	OTU, OTE, or OTL	879 µs	—
	COP to M file	—	735 µs plus 23 µs per word
	COP from M file	—	722 µs plus 22 µs per word
	FLL	—	716 µs plus 30 µs per word
	MVM to M file	850 µs	—
	any source or Destination M file address	694 µs	—

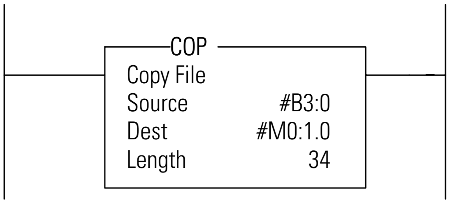
---

(1) Except the OSR instruction and the instruction parameters noted on page B-2.

SLC 5/02 Processor Example



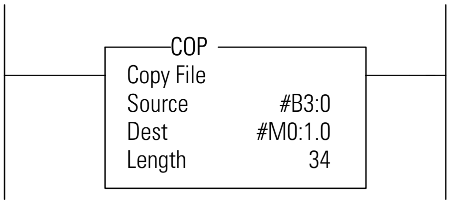
If you are using a SLC 5/02 Series B processor, add 1930 µs to the program scan time for each bit instruction addressed to an M0 or M1 data file. If you are using a SLC 5/03 Series C processor, add 1160 µs.



COP
Copy File
Source #B3:0
Dest #M0:1.0
Length 34

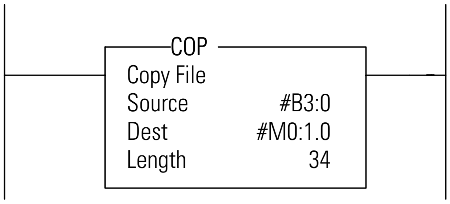
If you are using a SLC 5/02 Series B processor, add 1580 µs plus 670 µs per word of data addressed to the M0 or M1 file. As shown above, 34 words are copied from #B3:0 to M0:1.0. Therefore, this adds 24360 µs to the scan time of the COP instruction. If you are using a SLC 5/02 Series C processor, add 950 µs plus 400 µs per word. This adds 14550 µs to the scan time of the COP instruction.

SLC 5/03 Processor Example



The SLC 5/03 processor access times depend on the instruction type. Consult the table on page B-4 for the correct access times to add. As an example, if you use a COP to M file instruction like the one shown above, add 772 µs plus 23 µs per word. This adds 1554 µs to the SLC 5/03 processor scan time due to the COP instruction.

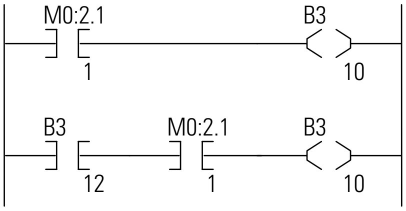
SLC 5/04 and SLC 5/05 Processor Example



The SLC 5/04 and SLC 5/05 processors access times depend on the instruction type. Consult the table on B-5 for the correct access times to add. As an example, if you use a COP to M file instruction like the one shown above, add 735 µs plus 23 µs per word. This adds 1517 µs to the SLC 5/04 and SLC 5/05 processor scan time due to the COP instruction.

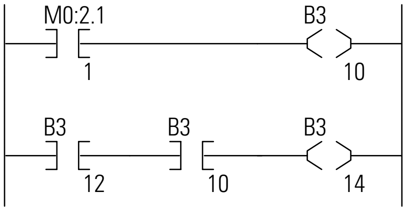
Minimizing the Scan Time

You can keep the processor scan time to a minimum by economizing on the use of instructions addressing the M0 or M1 files. For example, XIC instruction M0:2.1/1 is used in rungs 1 and 2 of the figure below, adding approximately 2 ms to the scan time if you are using an SLC 5/02 Series B processor.



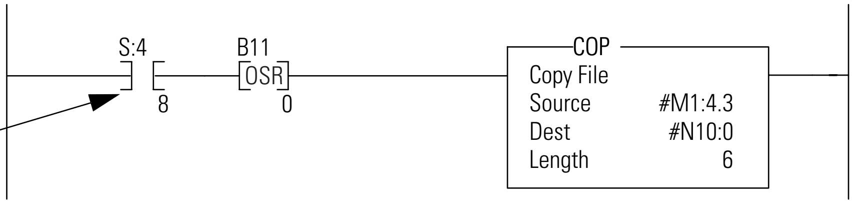
XIC instructions in rungs 1 and 2 are addressed to the M0 data file. Each of these instructions adds approximately 1 ms to the scan time (SLC 5/02 Series B Processor).

In the equivalent rungs of the figure below, XIC instruction M0:2.1/1 is used only in rung 1, reducing the SLC 5/02 scan time by approximately 1 ms.



These rungs provide equivalent operation to those of figure A by substituting XIC instruction B3/10 for XIC instruction M0:2.1/1 in rung 2. Scan time is reduced by approximately 1 ms (Series B processor).

The following figure illustrates another economizing technique. The COP instruction addresses an M1 file, adding approximately 4.29 ms to the scan time if you are using a SLC 5/02 Series B processor. Scan time economy is realized by making this rung true only periodically, as determined by clock bit S:4/8. (Clock bits are discussed in the *SLC 500 Instruction Set Reference Manual*, publication number 1747-RM001.) A rung such as this might be used when you want to monitor the contents of the M1 file, but monitoring need not be on a continuous basis.



S:4/8 causes the #M1:4.3 file to update the #N10:0 file every 2.56 seconds.

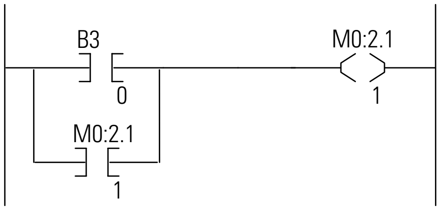
Capturing M0-M1 File Data

The first two ladder diagrams in the last section illustrate a technique allowing you to capture and use M0 or M1 data as it exists at a particular time. In the first figure, bit M0:2.1/1 could change state between rungs 1 and 2. This could interfere with the logic applied in rung 2. The second figure avoids the problem. If rung 1 is true, bit B3/10 captures this information and places it in rung 2.

In the second example of the last section, a COP instruction is used to monitor the contents of an M1 file. When the instruction goes true, the 6 words of data in file #M1:4.3 is captured as it exists at that time and placed in file #N10.0.

Specialty I/O Modules with Retentive Memory

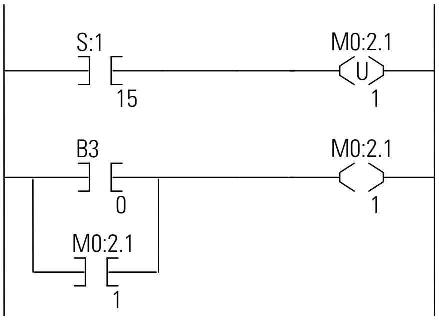
Certain specialty I/O modules retain the status of M0-M1 data after power is removed. See your specialty I/O module user's manual. This means that an OTE instruction having an M0 or M1 address remains on if it is on when power is removed. A "hold-in" rung as shown below will not function as it would if the OTE instruction were non-retentive on power loss. If the rung is true at the time power is removed, the OTE instruction latches instead of dropping out. When power is again applied, the rung will be evaluated as true instead of false.



ATTENTION

When used with a speciality I/O module having retentive outputs, this rung can cause unexpected start-up on powerup.

You can achieve non-retentive operation by unlatching the retentive output with the first pass bit at powerup:



This rung is true for the first scan after powerup to unlatch M0:2.1/1.

G Files

Some specialty I/O modules use G (configuration) files (indicated in the specific specialty I/O module user's manual). These files can be thought of as the software equivalent of DIP switches.

The content of G files is accessed and edited offline under the I/O Configuration function. Data you enter into the G file is passed on to the specialty I/O module when you download the processor file and enter the REM Run or any one of the REM Test modes.

Configuring G Files

The G file is configured as part of the I/O configuration procedure for the processor file. After you have assigned the specialty I/O module to a slot (the procedure is the same as assigning other modules), you may click on the advanced configuration button in order to configure the number of G file words.

Editing G File Data

Data in the G file must be edited according to your application and the requirements of the specialty I/O module. You edit the data offline under the I/O configuration function only.

IMPORTANT Word 0 of the G file is configured automatically by the processor according to the particular specialty I/O module. Word 0 cannot be edited.

Appendix C

RIO Configuration Worksheet

This appendix provides a worksheet to help you configure your RIO devices.

Use a photocopy of the worksheet so you retain a blank worksheet for future applications.

SLC Processor Input Image

	Bit Number-Decimal	15 High Byte 8	7 Low Byte 0	
Logical Rack 0	Group 0			I:e.0
	Group 1			I:e.1
	Group 2			I:e.2
	Group 3			I:e.3
	Group 4			I:e.4
	Group 5			I:e.5
	Group 6			I:e.6
	Group 7			I:e.7
Logical Rack 1	Group 0			I:e.8
	Group 1			I:e.9
	Group 2			I:e.10
	Group 3			I:e.11
	Group 4			I:e.12
	Group 5			I:e.13
	Group 6			I:e.14
	Group 7			I:e.15
Logical Rack 2	Group 0			I:e.16
	Group 1			I:e.17
	Group 2			I:e.18
	Group 3			I:e.19
	Group 4			I:e.20
	Group 5			I:e.21
	Group 6			I:e.22
	Group 7			I:e.23
Logical Rack 3	Group 24			I:e.24
	Group 25			I:e.25
	Group 26			I:e.26
	Group 27			I:e.27
	Group 28			I:e.28
	Group 29			I:e.29
	Group 30			I:e.30
	Group 31			I:e.31

SLC Processor Output Image

	Bit Number-Decimal	15 High Byte 8	7 Low Byte 0	
Logical Rack 0	Group 0			O:e.0
	Group 1			O:e.1
	Group 2			O:e.2
	Group 3			O:e.3
	Group 4			O:e.4
	Group 5			O:e.5
	Group 6			O:e.6
	Group 7			O:e.7
Logical Rack 1	Group 0			O:e.8
	Group 1			O:e.9
	Group 2			O:e.10
	Group 3			O:e.11
	Group 4			O:e.12
	Group 5			O:e.13
	Group 6			O:e.14
	Group 7			O:e.15
Logical Rack 2	Group 0			O:e.16
	Group 1			O:e.17
	Group 2			O:e.18
	Group 3			O:e.19
	Group 4			O:e.20
	Group 5			O:e.21
	Group 6			O:e.22
	Group 7			O:e.23
Logical Rack 3	Group 24			O:e.24
	Group 25			O:e.25
	Group 26			O:e.26
	Group 27			O:e.27
	Group 28			O:e.28
	Group 29			O:e.29
	Group 30			O:e.30
	Group 31			O:e.31

SLC Processor Input Image

	Bit Number-Decimal	15 High Byte 8	7 Low Byte 0	
Logical Rack 0	Group 0			I:e.0
	Group 1			I:e.1
	Group 2			I:e.2
	Group 3			I:e.3
	Group 4			I:e.4
	Group 5			I:e.5
	Group 6			I:e.6
	Group 7			I:e.7
Logical Rack 1	Group 0			I:e.8
	Group 1			I:e.9
	Group 2			I:e.10
	Group 3			I:e.11
	Group 4			I:e.12
	Group 5			I:e.13
	Group 6			I:e.14
	Group 7			I:e.15
Logical Rack 2	Group 0			I:e.16
	Group 1			I:e.17
	Group 2			I:e.18
	Group 3			I:e.19
	Group 4			I:e.20
	Group 5			I:e.21
	Group 6			I:e.22
	Group 7			I:e.23
Logical Rack 3	Group 24			I:e.24
	Group 25			I:e.25
	Group 26			I:e.26
	Group 27			I:e.27
	Group 28			I:e.28
	Group 29			I:e.29
	Group 30			I:e.30
	Group 31			I:e.31

SLC Processor Output Image

	Bit Number-Decimal	15 High Byte 8	7 Low Byte 0	
Logical Rack 0	Group 0			O:e.0
	Group 1			O:e.1
	Group 2			O:e.2
	Group 3			O:e.3
	Group 4			O:e.4
	Group 5			O:e.5
	Group 6			O:e.6
	Group 7			O:e.7
Logical Rack 1	Group 0			O:e.8
	Group 1			O:e.9
	Group 2			O:e.10
	Group 3			O:e.11
	Group 4			O:e.12
	Group 5			O:e.13
	Group 6			O:e.14
	Group 7			O:e.15
Logical Rack 2	Group 0			O:e.16
	Group 1			O:e.17
	Group 2			O:e.18
	Group 3			O:e.19
	Group 4			O:e.20
	Group 5			O:e.21
	Group 6			O:e.22
	Group 7			O:e.23
Logical Rack 3	Group 24			O:e.24
	Group 25			O:e.25
	Group 26			O:e.26
	Group 27			O:e.27
	Group 28			O:e.28
	Group 29			O:e.29
	Group 30			O:e.30
	Group 31			O:e.31

The following terms are used throughout this manual. Refer to the *Allen-Bradley Industrial Automation Glossary*, Publication Number AG-7.1, for a complete guide to Allen-Bradley technical terms.

Adapter

Any physical device that is a slave on the RIO link.

Adapter Image

That portion of the scanner image assigned to an individual adapter.

ASB Module

The Catalog Number 1747-ASB, 1771-ASB, or 1794-ASB Remote I/O Adapter Module. The ASB module is an adapter.

ASB Module Chassis

The chassis directly controlled by the ASB module. This includes the remote chassis and (if installed) two remote expansion chassis when using the 1747-ASB.

Block Transfer (BT)

See RIO Block Transfer.

Block Transfer Read (BTR)

A form of block transfer that occurs when a remote device transfers data to the SLC processor.

Block Transfer Write (BTW)

A form of block transfer that occurs when the SLC processor transfers data to a remote device.

Complementary I/O

Functionality that allows you to maximize I/O usage by pairing up I/O data from a primary and complementary chassis.

Discrete I/O

An input or output device that has corresponding bit locations in the scanner’s input or output file.

Discrete I/O Module

An I/O module used to sense or control two–state (ON/OFF) devices.

Data Transfer Control Word (DTCW)

This word is M0:s.3410 (where s=slot number of the 1747-BSN) and is used by the primary processor to initiate data transfers to the primary 1747-BSN. Eventually the primary processor initiates data transfers to the secondary 1747-BSN over the HSSL, and then to the secondary processor.

Data Transfer Handshake Word (DTHW)

This word is M0:S.3411 (where s=slot number of 1747-BSN) and is used by the secondary processor to inform the secondary 1747-BSN that it has received the latest data block and is ready for the next data block.

Data Transfer Status Word (DTSW)

This word is M1:s.3410 (where s=slot number of 1747-BSN and is used in both the primary and secondary systems to affect the transfer of data on the HSSL.

Extended Node Capability

Functionality that allows you to use an 82 Ohm termination resistor at both ends of the RIO link for all baud rates. This functionality also allows for up to 32 adapters to be connected to the RIO link.

G file

The SLC file used to configure the scanner. You enter configuration information into this file during SLC processor programming. This file is loaded to the scanner by the SLC processor upon entering run mode.

High-Speed Serial Link (HSSL)

2 Mbit/second link between primary and secondary 1747-BSN modules to provide the secondary SLC processor with input, block transfer, and user retentive data. Network status and control information are also transferred between 1747-BSN modules on this link.

Inhibit

A function by which the scanner stops communicating with a logical device. The logical device considers itself inhibited if it does not receive communications from the scanner within a certain period of time.

Input file

The scanner's input image file that is updated during the SLC processor input scan.

Local Expansion Chassis

A chassis that is connected to a local SLC chassis using a 1747-C9 (91.4 cm [36 in.]) or 1747-C7 (15.2 cm [6 in.]) cable.

Local SLC Chassis

The chassis that contains the SLC processor and scanner.

Logical Device

Any portion of a logical rack that is assigned to a single adapter. Adapters may appear as more than one logical device.

Logical Group

A logical group consists of one input and one output word within a logical rack. A word consists of 16 bits, each bit represents one terminal on a discrete I/O module.

Logical Rack

A fixed section of the scanner image comprised of eight input image words and eight output image words.

Logical Slot

A logical slot consists of one input and one output byte within a logical group. A byte consists of 8 bits, each bit represents one terminal on a discrete I/O module.

M files

The SLC M0 and M1 data files that reside in the scanner. M files contain RIO network status (M1) and control (M0) information. The contents of these files can be directly accessed by your application program. Also, the M files are used to control and monitor RIO block transfer operations.

Module Status Word (MSW)

This word is M1:s.3401 (where s=slot number of the 1747-BSN) and is used to monitor the status of the 1747-BSN itself and its counterpart in the remote system.

Output file

The scanner's output file that is updated during the SLC processor output scan.

Remote Chassis

The chassis containing an ASB module and connected to the local SLC chassis via the RIO link.

Remote Expansion Chassis

A chassis that is connected to a remote chassis using a 1747-C9 (91.4 cm [36 in.]) or 1747–C7 (15.2 cm [6 in.]) cable.

Reset, Adapter Decide

Commands sent by the scanner to a logical device during an RIO discrete transfer. These commands instruct the logical device to reset all of its discrete outputs if hold last state is not selected, or to hold all of its discrete outputs in their last state if hold last state is selected.

Reset, Adapter Reset

Commands sent by the scanner to a logical device during an RIO discrete transfer. These commands instruct the logical device to reset all of its discrete outputs, regardless of the hold last state selection.

RIO Block Transfer

The exchange of up to 64 words of data between the scanner and a remote device. RIO block transfers only occur if you program them in your processor control program.

RIO Discrete Transfer

The exchange of image data between the scanner and adapter. RIO discrete transfers occur continuously whenever the scanner and adapter are communicating on the RIO link.

RIO Link

An Allen-Bradley communication system supporting high-speed serial transfer of Remote I/O (RIO) control information. This link consists of one master and one or more slaves.

RIO Link Device

Refers to any Allen-Bradley or licensed third party product that connects to the RIO link as an adapter or slave device.

Scanner Image

The data table area within the scanner, used to exchange I/O information between the scanner and all the adapters on the RIO link. The scanner image is a portion of the SLC processor image.

SLC Chassis

A physical SLC rack that houses SLC processors and 1746 and 1747 I/O modules.

SLC Processor

The processor that controls the SLC chassis in which the scanner is installed.

Slot

The physical location in any SLC chassis used to insert I/O modules.

Specialty I/O Module

An I/O module other than a discrete I/O module (e.g., an analog module).

System Status Word (SSW)

This word is M1:s.3400 (where s=slot number of 1747-BSN and is used to monitor the status of the entire backup system.

Notes:

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation, Vorstlaan/Boulevard du Souverain 36, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Publication 1747-UM010B-EN-P - September 2003
Supersedes Publication 1747-6.22 - October 1999