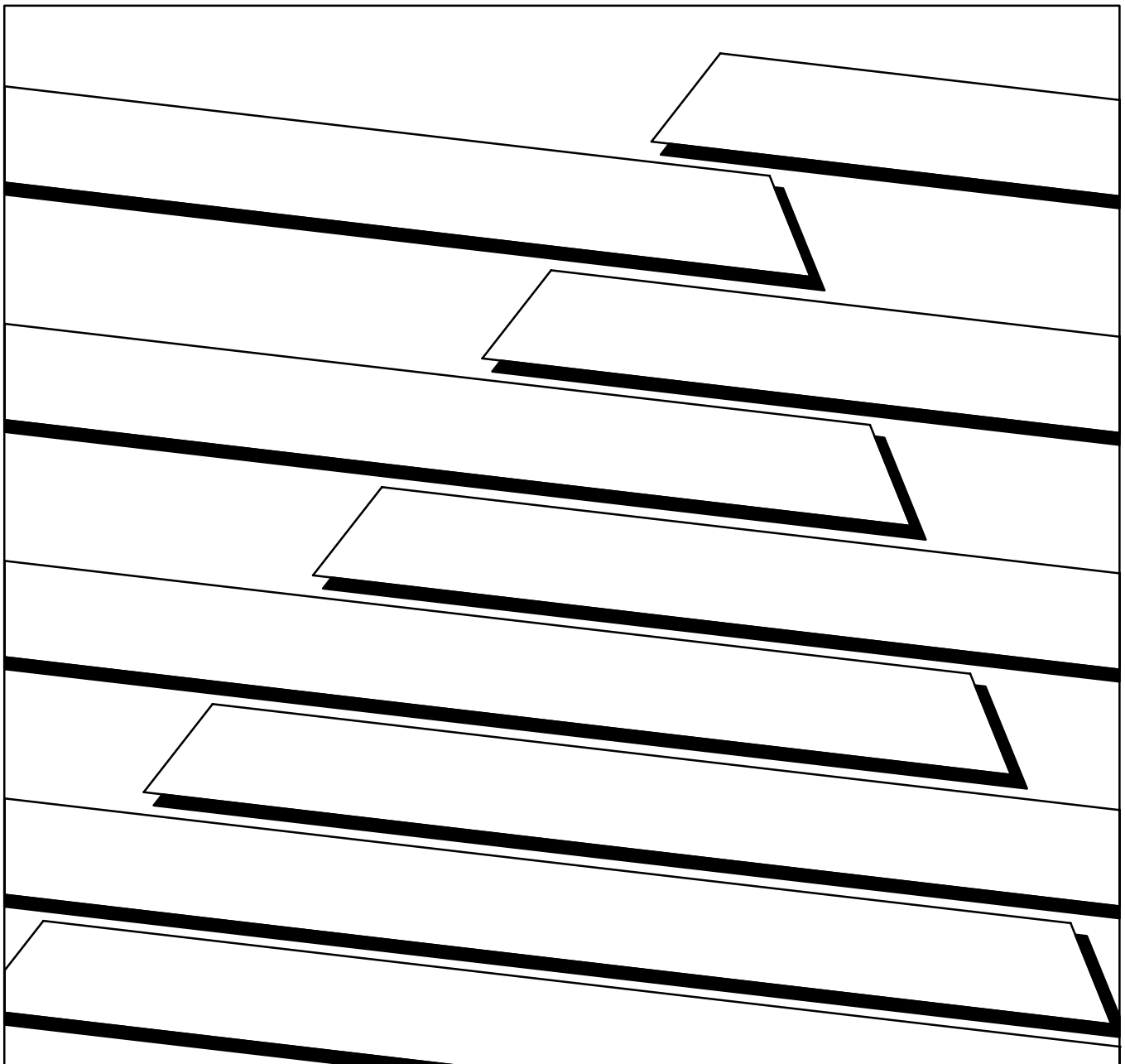




**ALLEN-BRADLEY**

# **Analog Input Module Cat. No. 1771-IFE**

User Manual



## Important User Information

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

The illustrations, charts, sample programs and layout examples shown in this guide are intended solely for example. Since there are many variables and requirements associated with any particular installation, Allen-Bradley does not assume responsibility or liability (to include intellectual property liability) for actual use based 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 Allen-Bradley office) describes some important differences between solid-state equipment and electromechanical devices which should be taken into consideration when applying products such as those described in this publication.

Reproduction of the contents of this copyrighted publication, in whole or in part, without written permission of Allen-Bradley Company, Inc. is prohibited.

Throughout this manual we make notes to alert you to possible injury to people or damage to equipment under specific circumstances.



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

---

Attention helps you:

- Identify a hazard.
- Avoid the hazard.
- Recognize the consequences.

**Important:** Identifies information that is especially important for successful application and understanding of the product.

**Important:** We recommend you frequently backup your application programs on appropriate storage medium to avoid possible data loss.

## Summary of Changes

### Summary of Changes

This release of the publication contains updated information from the last release.

### Updated Information

This release includes information previously included in a documentation update (publication 1771-6.5.90–RN1 dated March 1993).

In addition, many areas in this publication have been restructured or rewritten.

To help you find new and updated information in this release of the publication, we have included change bars as shown to the right of this paragraph.

# Table of Contents

---

<b>Summary of Changes</b> .....	<b><a href="#">S-1</a></b>
<b>Using This Manual</b> .....	<b><a href="#">P-1</a></b>
Purpose of Manual .....	<a href="#">P-1</a>
Audience .....	<a href="#">P-1</a>
Vocabulary .....	<a href="#">P-1</a>
Manual Organization .....	<a href="#">P-1</a>
Related Products .....	<a href="#">P-2</a>
Product Compatibility .....	<a href="#">P-2</a>
Related Publications .....	<a href="#">P-3</a>
<b>Overview of the Analog Input Module</b> .....	<b><a href="#">1-1</a></b>
Chapter Objectives .....	<a href="#">1-1</a>
Module Description .....	<a href="#">1-1</a>
Features .....	<a href="#">1-1</a>
How Analog Modules Communicate with Programmable Controllers .....	<a href="#">1-2</a>
Accuracy .....	<a href="#">1-3</a>
Chapter Summary .....	<a href="#">1-3</a>
<b>Installing the Input Module</b> .....	<b><a href="#">2-1</a></b>
Chapter Objectives .....	<a href="#">2-1</a>
Before You Install Your Input Module .....	<a href="#">2-1</a>
Electrostatic Damage .....	<a href="#">2-1</a>
Power Requirements .....	<a href="#">2-2</a>
Module Location in the I/O Chassis .....	<a href="#">2-2</a>
Module Keying .....	<a href="#">2-2</a>
Wiring Your Input Module .....	<a href="#">2-3</a>
Grounding .....	<a href="#">2-8</a>
Changing the Module's Configuration .....	<a href="#">2-9</a>
Module Installation .....	<a href="#">2-12</a>
Indicator Lights .....	<a href="#">2-12</a>
Chapter Summary .....	<a href="#">2-12</a>
<b>Module Programming</b> .....	<b><a href="#">3-1</a></b>
Chapter Objectives .....	<a href="#">3-1</a>
Block Transfer Programming .....	<a href="#">3-1</a>
PLC-2 Programming .....	<a href="#">3-2</a>
PLC-3 Programming .....	<a href="#">3-3</a>
PLC-5 Programming .....	<a href="#">3-4</a>
Module Scan Time .....	<a href="#">3-5</a>

Chapter Summary .....	<a href="#">3-5</a>
<b>Configuring Your Module .....</b>	<b><a href="#">4-1</a></b>
Chapter Objectives .....	<a href="#">4-1</a>
Configuring Your Input Module .....	<a href="#">4-1</a>
Input Range Selection .....	<a href="#">4-2</a>
Input Type .....	<a href="#">4-3</a>
Data Format .....	<a href="#">4-3</a>
Digital Filtering .....	<a href="#">4-4</a>
Real Time Sampling .....	<a href="#">4-5</a>
Scaling .....	<a href="#">4-6</a>
Default Configuration .....	<a href="#">4-8</a>
Chapter Summary .....	<a href="#">4-9</a>
<b>Module Status and Input Data .....</b>	<b><a href="#">5-1</a></b>
Chapter Objectives .....	<a href="#">5-1</a>
Reading Data From Your Module .....	<a href="#">5-1</a>
Block Transfer Read Format .....	<a href="#">5-2</a>
Chapter Summary .....	<a href="#">5-2</a>
<b>Calibrating Your Module .....</b>	<b><a href="#">6-1</a></b>
Chapter Objectives .....	<a href="#">6-1</a>
Tools and Equipment .....	<a href="#">6-1</a>
Calibration Procedure .....	<a href="#">6-1</a>
Chapter Summary .....	<a href="#">6-3</a>
<b>Troubleshooting Your Input Module .....</b>	<b><a href="#">7-1</a></b>
Chapter Objective .....	<a href="#">7-1</a>
Diagnostics Reported by the Module .....	<a href="#">7-1</a>
Chapter Summary .....	<a href="#">7-3</a>
<b>Specifications .....</b>	<b><a href="#">A-1</a></b>
<b>Programming Examples .....</b>	<b><a href="#">B-1</a></b>
Sample Programs for the Analog Input Module .....	<a href="#">B-1</a>
PLC-2 Family Processors .....	<a href="#">B-1</a>
PLC-3 Family Processor .....	<a href="#">B-2</a>
PLC-5 Family Processors .....	<a href="#">B-4</a>
<b>Data Table Formats .....</b>	<b><a href="#">C-1</a></b>
4-Digit Binary Coded Decimal (BCD) .....	<a href="#">C-1</a>
Signed-magnitude Binary .....	<a href="#">C-2</a>
Two's Complement Binary .....	<a href="#">C-3</a>

---

<b>Block Transfer (Mini-PLC-2 and PLC-2/20 Processors) . . .</b>	<b><a href="#">D-1</a></b>
Multiple GET Instructions - Mini-PLC-2 and PLC-2/20 Processors .	<a href="#">D-1</a>
Setting the Block Length (Multiple GET Instructions only) . . . . .	<a href="#">D-3</a>
<b>Forms . . . . .</b>	<b><a href="#">E-1</a></b>
Analog Block Transfer Read . . . . .	<a href="#">E-2</a>
Analog Block Transfer Write . . . . .	<a href="#">E-3</a>

## Using This Manual

### Purpose of Manual

This manual shows you how to use your Analog Input module with an Allen-Bradley programmable controller. It helps you install, program, calibrate, and troubleshoot your module.

### Audience

You must be able to program and operate an Allen-Bradley programmable controller to make efficient use of your input module. In particular, you must know how to program block transfers.

We assume that you know how to do this in this manual. If you do not, refer to the appropriate programming and operations manual before you attempt to program this module.

### Vocabulary

In this manual, we refer to:

- Each individual analog input module as the “input module”
- The Programmable Controller as the “controller”

### Manual Organization

This manual is divided into seven chapters. The following chart shows each chapter with its corresponding title and a brief overview of the topics covered in that chapter.

Chapter	Title	Topics Covered
1	Overview of the input modules	Description of the module including general and hardware features
2	Installing the module	Module power requirements, keying, chassis location Wiring of the field wiring arm
3	Module programming	Sample programs
4	Module configuration	Hardware and software configuration Input range selection Data format
5	Module status and input data	Reading data from the module Read block format
6	Calibration	Information on calibrating your module
7	Troubleshooting your module	Troubleshooting guide for problem diagnosis

Chapter	Title	Topics Covered
Appendix	Title	Topics Covered
A	Specifications	
B	Programming Examples	
C	Data Formats	Information on BCD, 2s complement binary, signed magnitude (12-bit) binary
D	Block transfer with Mini-PLC-2 and PLC-2/20 processors	How to use GET-GET instructions
E	Forms	Useful forms for identifying your data table

## Related Products

You can install your input module in any system that uses Allen-Bradley programmable controllers with block transfer capability and the 1771 I/O structure.

Contact your nearest Allen-Bradley office for more information about your programmable controllers.

## Product Compatibility

The 1771-IFE module can be used with any 1771 I/O chassis. Communication between the discrete analog module and the processor is bidirectional; the processor block-transfers output data through the output image table to the module and block-transfers input data from the module through the input image table. The module also requires an area in the data table to store the read block transfer data and write block transfer data. I/O image table use is an important factor in module placement and addressing selection. Compatibility and data table use is listed in Table P.A.

**Table P.A**  
**Compatibility and Use of Data Table**

Catalog Number	Use of Data Table				Compatibility			
	Input Image Bits	Output Image Bits	Read Block Words	Write Block Words	Addressing			Chassis Series
					1/2-Slot	1-Slot	2-Slot	
1771-IFE	8	8	20	37	Y	Y	Y	A, B

A = Compatible with 1771-A1, -A2, -A4  
 B = Compatible with 1771-A1B, -A2B, -A3B, -A3B1, -A4B  
 Y = Compatible without restriction.

You can place your input module in any I/O module slot of the I/O chassis. You can put two input modules in the same module group. You can put an input and an output module in the same module group.



Do not put the module in the same module group as a discrete high density module. Avoid placing analog input modules close to ac modules or high voltage dc modules.

## **Related Publications**

For a list of publications with information on Allen-Bradley programmable controller products, consult our publication index (SD499).

## Overview of the Analog Input Module

### Chapter Objectives

This chapter gives you information on:

- features of the module
- how the input module communicates with programmable controllers

### Module Description

The Analog input module is an intelligent block transfer module that interfaces analog input signals with any Allen-Bradley programmable controllers that have block transfer capability. Block transfer programming moves input data words from the module's memory to a designated area in the processor data table in a single scan. It also moves configuration words from the processor data table to module memory.

The input module is a single-slot module and requires no external power supply. (If using passive transducers for input, the user must supply loop power.) After scanning the analog inputs, the input data is converted to a specified data type in a digital format to be transferred to the processor's data table on request. The block transfer mode is disabled until this input scan is complete. Consequently, the minimum interval between block transfer reads is the same as the total input update time for each analog input module.

### Features

The Analog input module senses up to 16 single-ended or 8 differential analog inputs and converts them to a proportional four-digit BCD or twelve-bit binary value. You can select from five voltage or three current input ranges. Each input can be configured as a current or voltage input with internal jumpers.

This module's features include:

- 16 single-ended or 8 differential inputs on one card
- User program selectable input ranges on a per channel basis (Table 1.A)
- Selectable real-time sampling
- Selectable scaling to engineering units
- Selectable digital filtering
- Selectable data format

**Table 1.A**  
**Program Selectable Input Ranges**

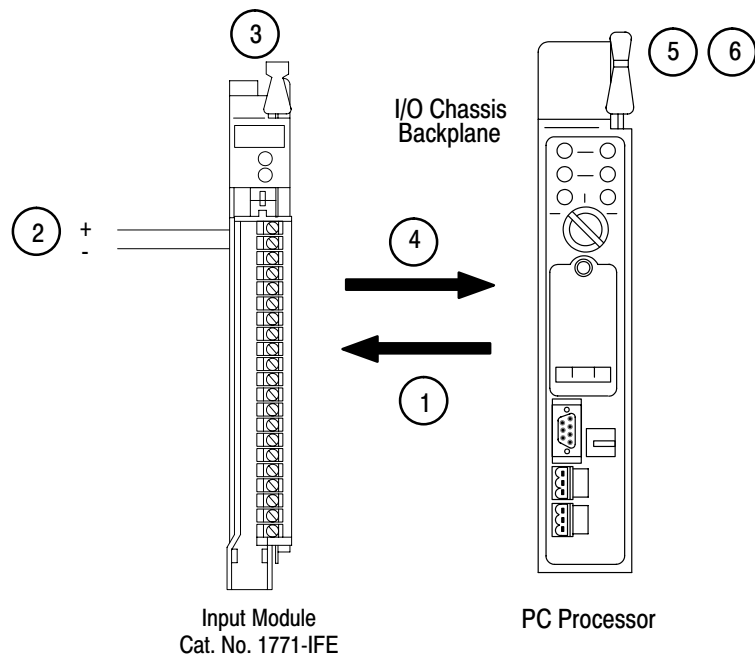
Voltage	Current
1 to 5V dc	4 to 20mA
0 to 5V dc	0 to 20mA
-5 to +5V dc	-20 to +20mA
-10 to +10V dc	
0 to 10V dc	

**How Analog Modules Communicate with Programmable Controllers**

The processor transfers data to the module (block transfer write) and from the module (block transfer read) using BTW and BTR instructions in your ladder diagram program. These instructions let the processor obtain input values and status from the module, and let you establish the module's mode of operation (Figure 1.1).

1. The processor transfers your configuration data to the module via a block transfer write instruction.
2. External devices generate analog signals that are transmitted to the module.

**Figure 1.1**  
**Communication Between Processor and Module**



3. The module converts analog signals into binary or BCD format, and stores these values until the processor requests their transfer.
4. When instructed by your ladder program, the processor performs a read block transfer of the values and stores them in a data table.
5. The processor and module determine that the transfer was made without error, and that input values are within specified range.
6. Your ladder program can use and/or move the data (if valid) before it is written over by the transfer of new data in a subsequent transfer.
7. Your ladder program should allow write block transfers to the module only when enabled by operator intervention or at power-up.

### **Accuracy**

The accuracy of your input module is described in Appendix A.

### **Chapter Summary**

In this chapter you read about the functional aspects of the input module and how the module communicates with the programmable controller.

## Installing the Input Module

### Chapter Objectives

This chapter gives you information on:

- calculating the chassis power requirement
- choosing the module's location in the I/O chassis
- keying a chassis slot for your module
- wiring the input module's field wiring arm
- configuring your module configuration plugs
- installing the input module

### Before You Install Your Input Module

Before installing your input module in the I/O chassis:

You need to:	As described under:
Calculate the power requirements of all modules in each chassis.	Power Requirements, page 2-2.
Determine where to place the module in the I/O chassis.	Module Location in the I/O Chassis, page 2-2.
Key the backplane connector in the I/O chassis.	Module Keying, page 2-2.
Make connections to the wiring arm.	Wiring Your Input Module, page 2-3 and Grounding, page 2-8.

### Electrostatic Damage

Electrostatic discharge can damage semiconductor devices inside this module if you touch backplane connector pins. Guard against electrostatic damage by observing the following precautions:



**ATTENTION:** Electrostatic discharge can degrade performance or cause permanent damage. Handle the module as stated below.

- Wear an approved wrist strap grounding device, or touch a grounded object to rid yourself of electrostatic charge before handling the module.
- Handle the module from the front, away from the backplane connector. Do not touch backplane connector pins.
- Keep the module in its static-shield bag when not in use.

## **Power Requirements**

Your module receives its power through the 1771 I/O power supply. The module requires 750mA from the backplane.

Add this current to the requirements of all other modules in the I/O chassis to prevent overloading the chassis backplane and/or backplane power supply.

## **Module Location in the I/O Chassis**

Place your module in any I/O module slot of the I/O chassis except for the extreme left slot. This slot is reserved for PC processors or adapter modules.

Group your modules to minimize adverse affects from radiated electrical noise and heat. We recommend the following.

- Group analog input and low voltage dc modules away from ac modules or high voltage dc modules to minimize electrical noise interference.
- Do not place this module in the same I/O group with a discrete high-density I/O module when using 2-slot addressing. This module uses a byte in both the input and output image tables for block transfer.

After determining the module's location in the I/O chassis, connect the wiring arm to the pivot bar at the module's location.

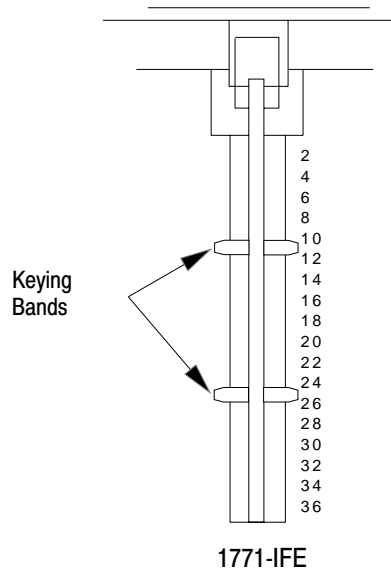
## **Module Keying**

Use the plastic keying bands, shipped with each I/O chassis, for keying I/O slots to accept only one type of module.

The module is slotted in two places on the edge of the rear circuit board. The position of the keying bands on the backplane connector must correspond to these slots to allow insertion of the module. You can key any connector in an I/O chassis to receive this module except for the leftmost connector reserved for adapter or processor modules. Place keying bands between the following numbers labeled on the backplane connector (Figure 2.1):

- between 10 and 12
- between 24 and 26

**Figure 2.1**  
**Keying Positions**



12676

## Wiring Your Input Module

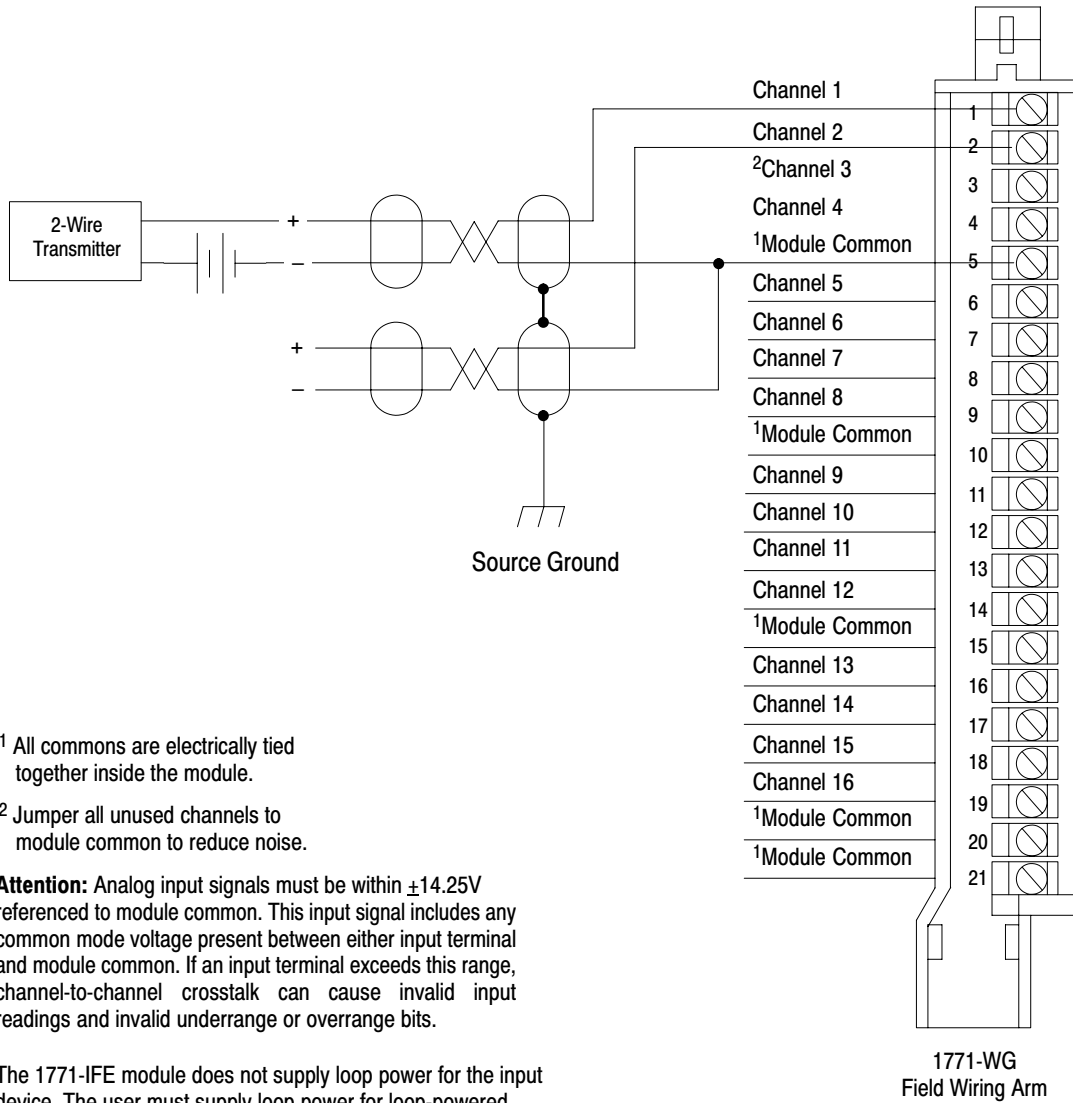
Connect your I/O devices to the cat. no. 1771-WG wiring arm shipped with the module. Attach the wiring arm to the pivot bar at the bottom of the I/O chassis. It pivots upward and connects with the module so you can install or remove the module without disconnecting the wires.

Input connections for the 1771-IFE with single-ended inputs are shown in Figure 2.2 and Figure 2.3. Input connections for the 1771-IFE with differential inputs are shown in Figure 2.4 and Figure 2.5.

Recommended maximum cable length for voltage-mode input devices is 50 feet. This recommendation is based on considerations of signal degradation and electrical noise immunity in typical industrial environments. Cable length for current-mode input devices need not be as restrictive because analog signals from these devices are less sensitive to electrical noise interference.

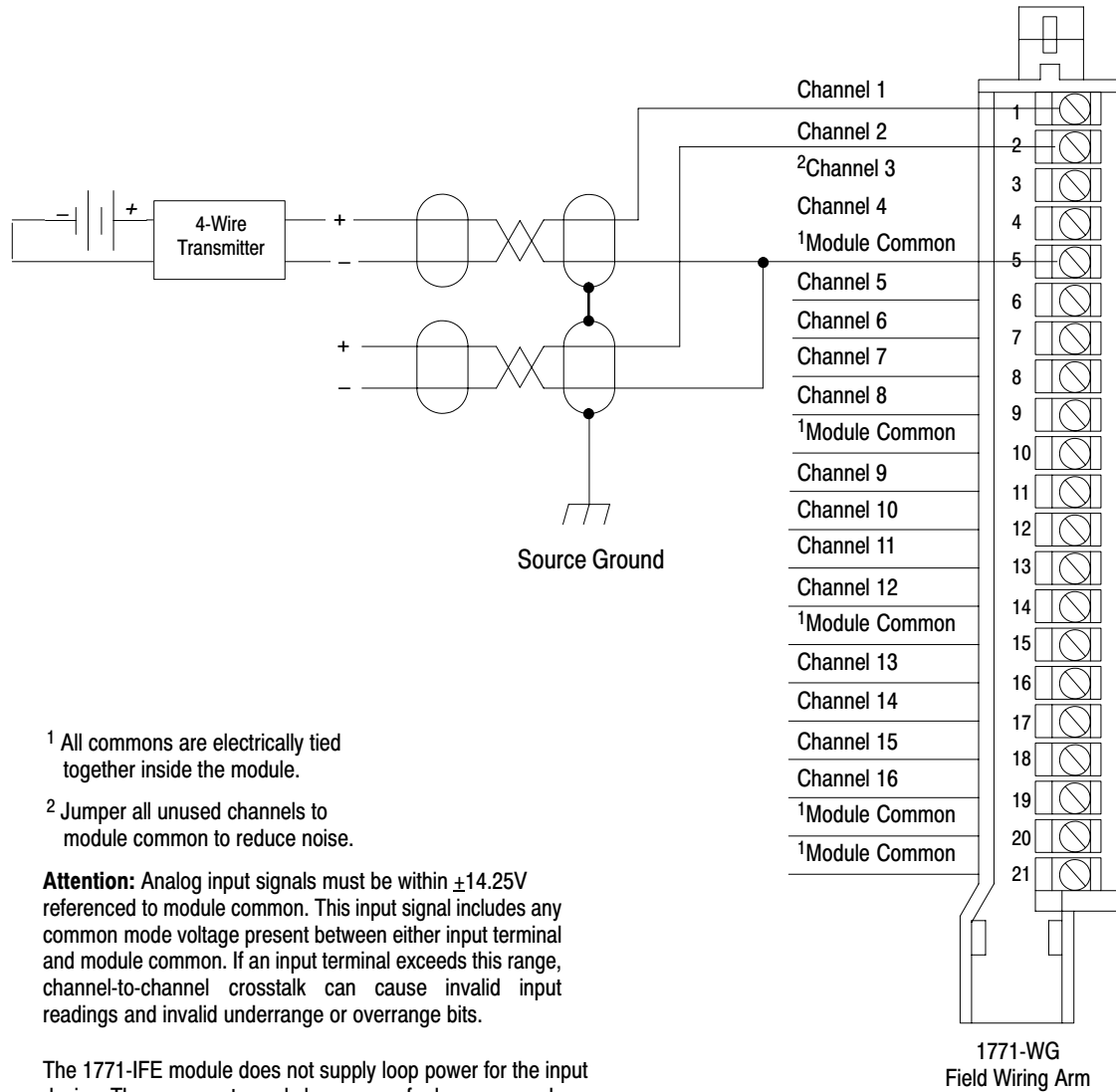
**The 1771-IFE module is shipped from the factory set for a 1 to 5V DC voltage input.** Refer to “Changing Your Module’s Configuration” on page 2-9 for other combinations of current and voltage inputs.

**Figure 2.2**  
Connection Diagram for 16 Single-ended Inputs and Two-Wire Transmitters



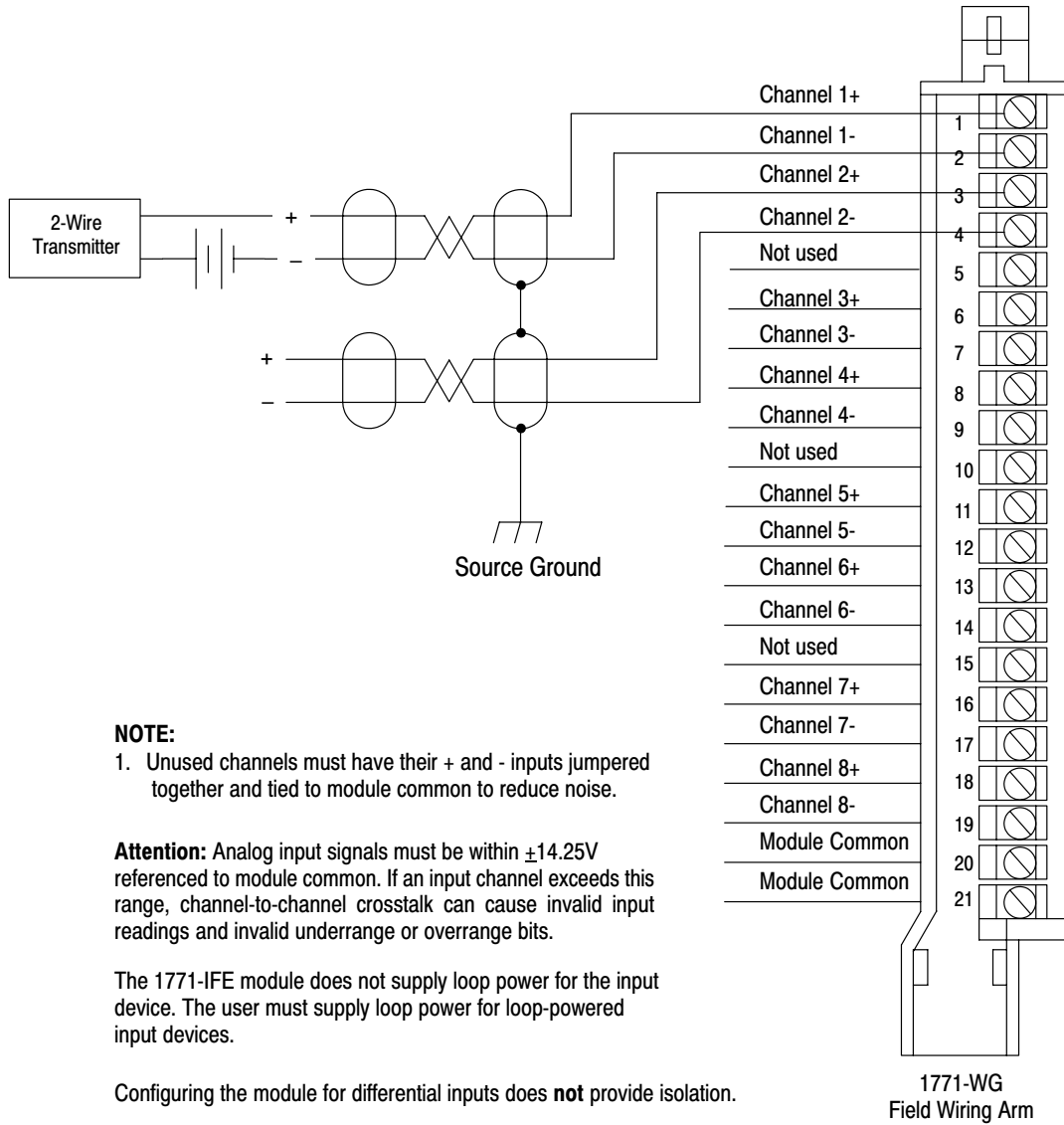


**Figure 2.3**  
**Connection Diagram for 16 Single-ended Inputs and Four-Wire Transmitters**



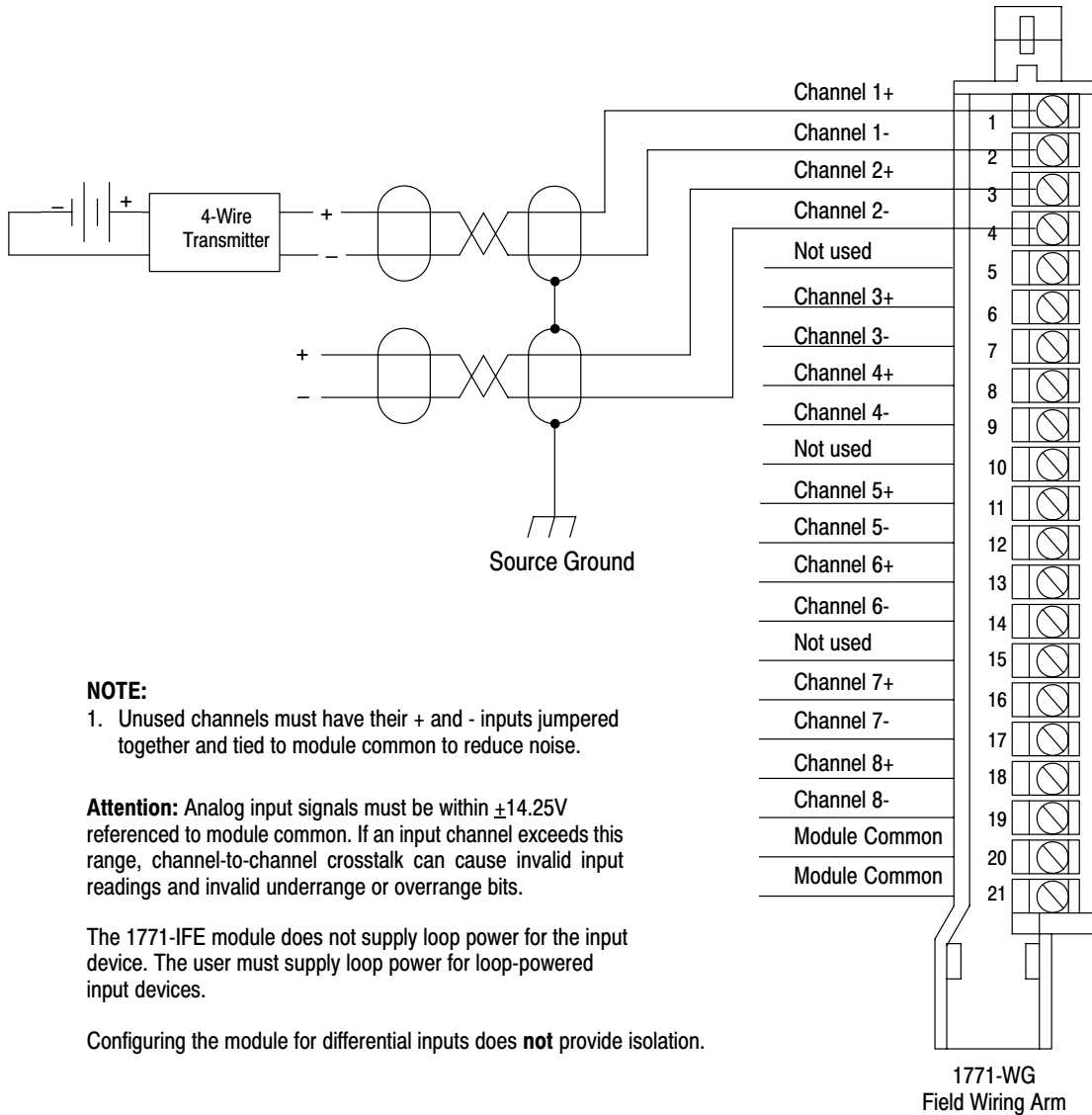
10948-I

**Figure 2.4**  
**Connection Diagram for 8 Differential Inputs and Two-Wire Transmitters**



10949-1

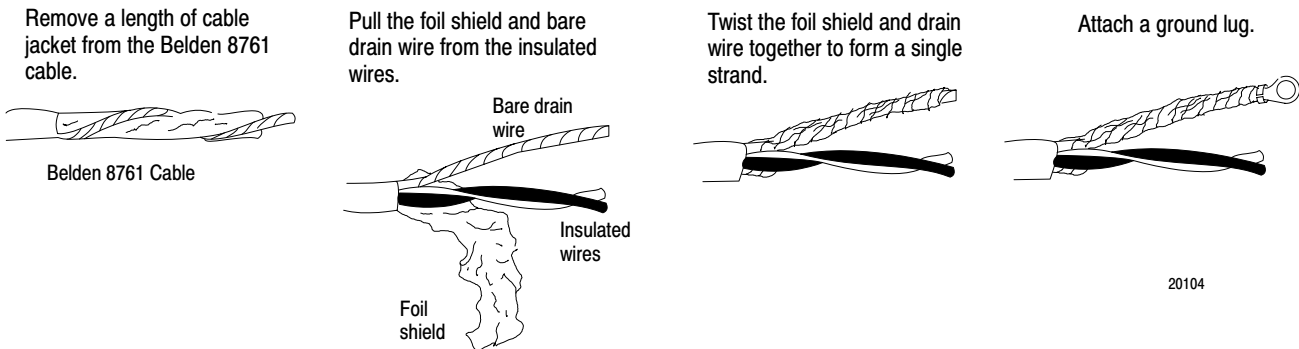
**Figure 2.5**  
**Connection Diagram for 8 Differential Inputs and Four-Wire Transmitters**



## Grounding

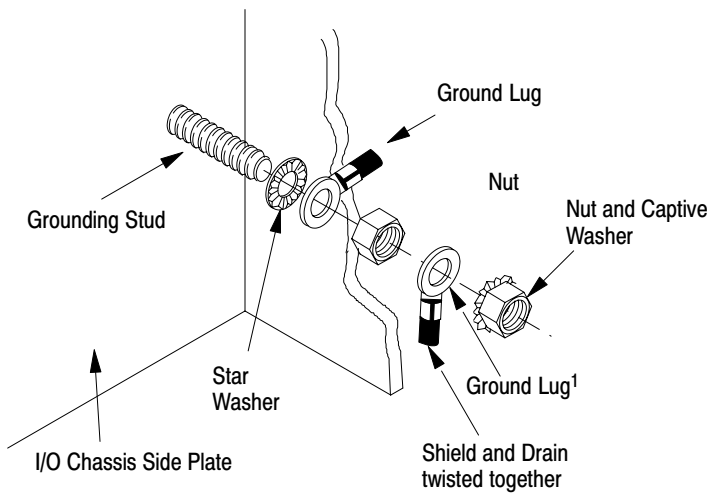
When using shielded cable wire, ground the foil shield and drain wire only at one end of the cable. We recommend that you wrap the foil shield and drain wire together and connect them to a chassis mounting bolt (Figure 2.6). At the opposite end of the cable, tape exposed shield and drain wire with electrical tape to insulate it from electrical contact.

**Figure 2.6**  
**Cable Grounding**

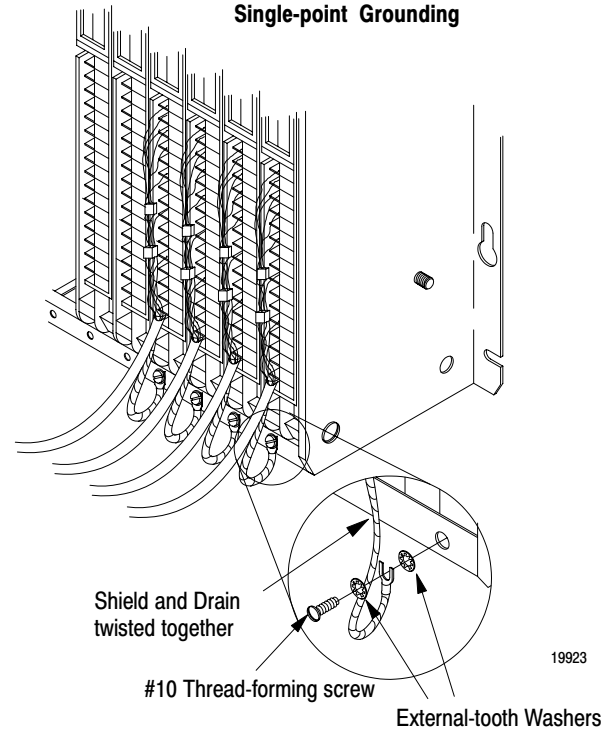


### Chassis Ground

When you connect grounding conductors to the I/O chassis grounding stud, place a star washer under the first lug, then place a nut with captive lock washer on top of each ground lug.



### Single-point Grounding



Refer to Wiring and Grounding Guidelines, publication 1770-4.1 for additional information.

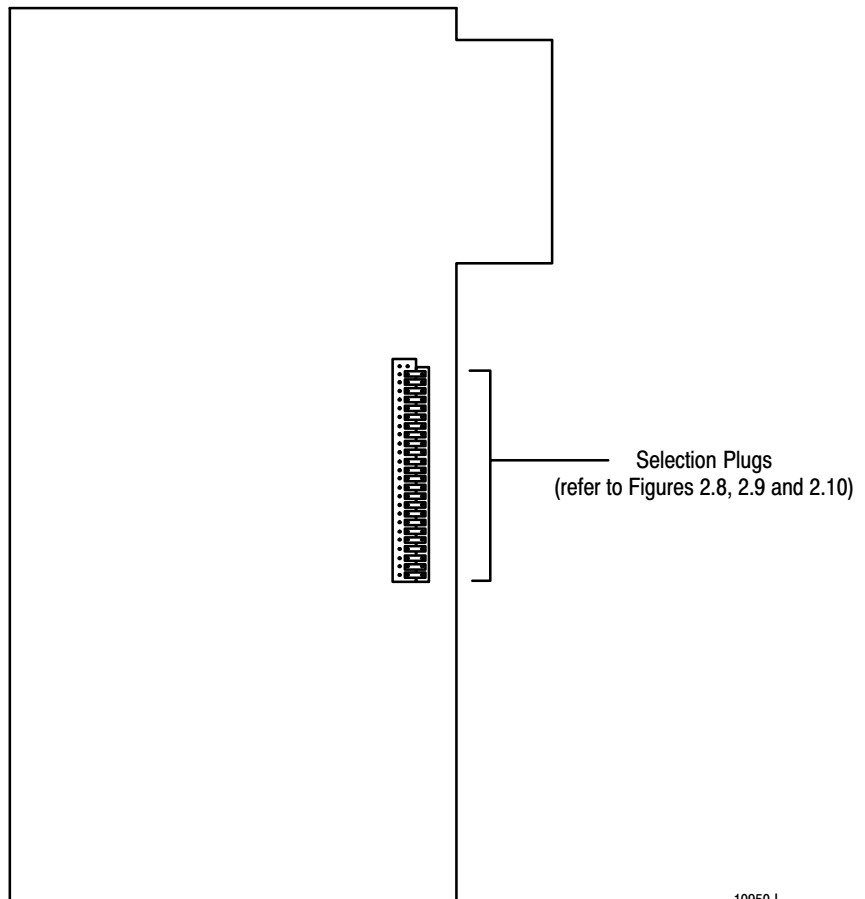
## Changing the Module's Configuration

The analog input module (1771-IFE) has configuration plugs for determining the input type (voltage or current) desired for each input. The module comes from the factory with the plugs positioned for voltage inputs.

To set the configuration plugs for your desired inputs, proceed as follows:

1. Remove the module's covers by removing the four screws securing the covers to the module.
2. Locate the selection plugs (Figure 2.7).

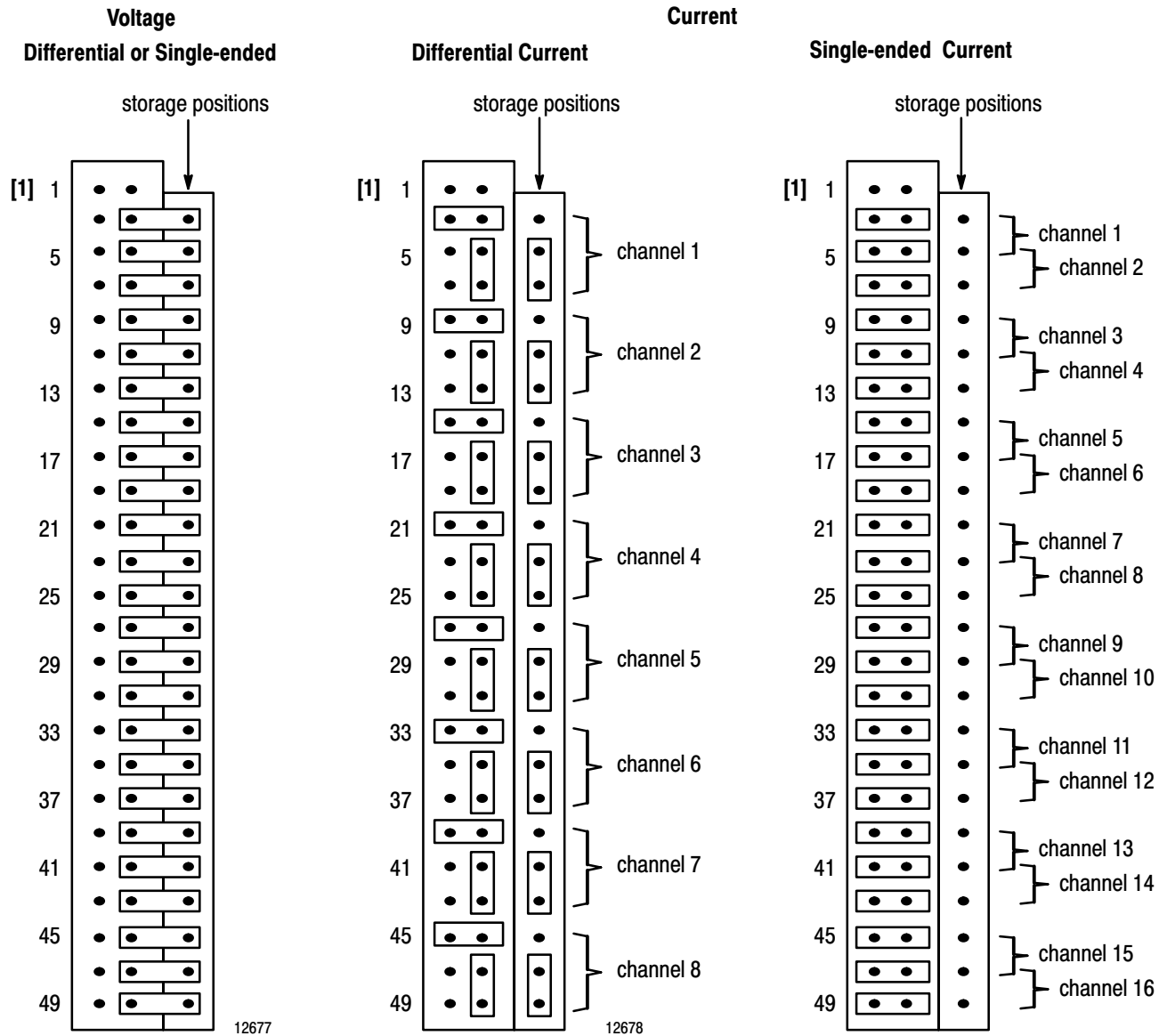
**Figure 2.7**  
Configuration Plug Locations



10950-1

3. Position the plugs as shown in the Figures for your particular module (Figures 2.8, 2.9 and 2.10).
4. Reassemble the module after you have finished checking and/or setting the selection plugs.

**Figure 2.8**  
Selection Plug Settings for Differential or Single-ended Voltage or  
Differential or Single-ended Current Inputs

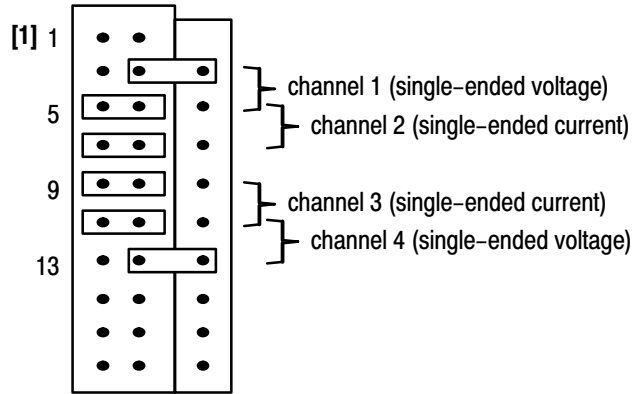


[1] Positions 1 and 2 are not used.

**Note:** Plugs are not needed for operation in the voltage mode.

10951-I

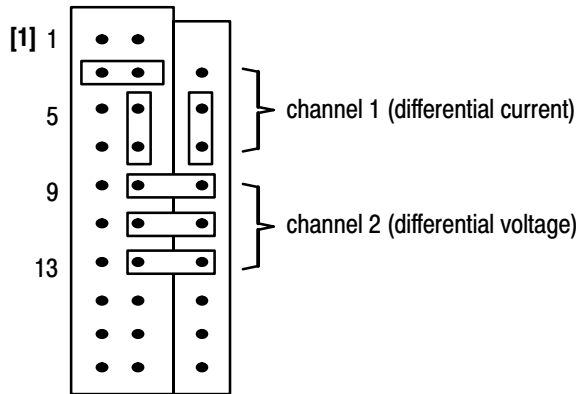
**Figure 2.9**  
**Configuration Plug Settings for Single-ended Voltage and Current Inputs on Adjacent Channels**



[1] positions 1 and 2 are not used

10952-I

**Figure 2.10**  
**Configuration Plug Settings for Differential Voltage and Current Inputs on Adjacent Channels**



[1] positions 1 and 2 are not used

10953-I

**Note:** Either differential and single-ended configurations must be selected for the entire module.

## Module Installation

When installing your module in an I/O chassis:

1. First, turn off power to the I/O chassis.



**ATTENTION:** Remove power from the 1771 I/O chassis backplane and wiring arm before removing or installing an I/O module.

Failure to remove power from the backplane could cause injury or equipment damage due to possible unexpected operation.

Failure to remove power from the backplane or wiring arm could cause module damage, degradation of performance, or injury.

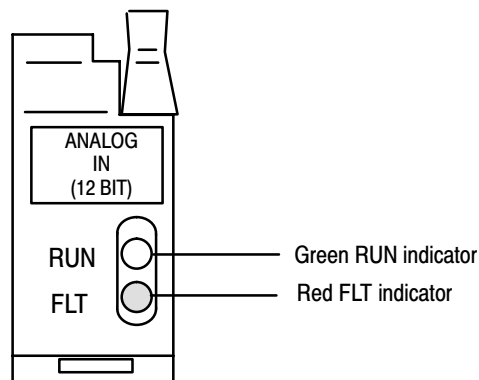
---

2. Place the module in the plastic tracks on the top and bottom of the slot that guides the module into position.
3. Do not force the module into its backplane connector. Apply firm even pressure on the module to seat it properly.
4. Snap the chassis latch over the top of the module to secure it.
5. Connect the wiring arm to the module.

## Indicator Lights

The front panel of the input module contains a green RUN and a red FLT (fault) indicator (Figure 2.11). At power-up an initial module self-check occurs. If there is no fault, the red indicator turns off. The green indicator will be on when the module is powered. If a fault is found initially or occurs later, the red FLT indicator lights. Possible module fault causes and corrective action is discussed in Chapter 7, Troubleshooting.

**Figure 2.11**  
**Diagnostic Indicators**



10528-1

## Chapter Summary

In this chapter you learned how to install your input module in an existing programmable controller system and how to wire to the field wiring arm.



## Module Programming

### Chapter Objectives

In this chapter we describe:

- block transfer programming
- sample programs in the PLC-2, PLC-3 and PLC-5 processors
- module scan time issues

### Block Transfer Programming

Your module communicates with your processor through bidirectional block transfers. This is the sequential operation of both read and write block transfer instructions.

The block transfer write (BTW) instruction is initiated when the analog module is first powered up, and subsequently only when the programmer wants to write a new configuration to the module. At all other times the module is basically in a repetitive block transfer read (BTR) mode.

The application programs for the three processor families were written to accomplish this handshaking in the described manner. They are minimum programs; all the rungs and conditioning must be included in your application program. If you wish to disable BTRs for any reason, or add interlocks to the BTW rung to prevent writes from happening at certain times, you are allowed to do it. You may **not** eliminate any storage bits or interlocks that are included in our examples. If interlocks are removed, the program may not work properly.

**The analog input module will work with a default configuration of zeroes entered in all five words of a five word BTW configuration block.** See the configuration default section to understand what this configuration will look like. Also, refer to Appendix B for example configuration blocks and instruction addresses to get started.



**ATTENTION:** In PLC-2 family processors you must not enable both the read and write instructions at the same time. Undesirable data could transfer, resulting in unpredictable machine operation. Using the prescribed programs will prevent this situation.

---

**PLC-2 Programming**

The PLC-2 program example regulates when each block transfer will be initiated to eliminate problems caused by limited regulation of bidirectional block transfers. Both storage bits are needed, as shown in the example, to accomplish this task in all PLC-2 systems, local or remote, with long or short program scans. Therefore, the program as shown is the minimum required. Note that PLC-2 processors that do not have the block transfer instruction must use the GET-GET block transfer format which is outlined in Appendix E.

**Figure 3.1**  
**PLC-2 Family Sample Program Structure**

**Rung 1**

Block transfer read buffer: the file-to-file move instruction holds the block transfer read (BTR) data (file A) until the processor checks the data integrity. If the data was successfully transferred, the processor energizes the BTR done bit, initiating a data transfer to the buffer (file R) for use in the program.

If the data is corrupted during the BTR operation, the BTR done bit is not energized and data is not transferred to the buffer file. In this case, the data in the BTR file will be overwritten by data from the next BTR.

**Rungs 2 and 3**

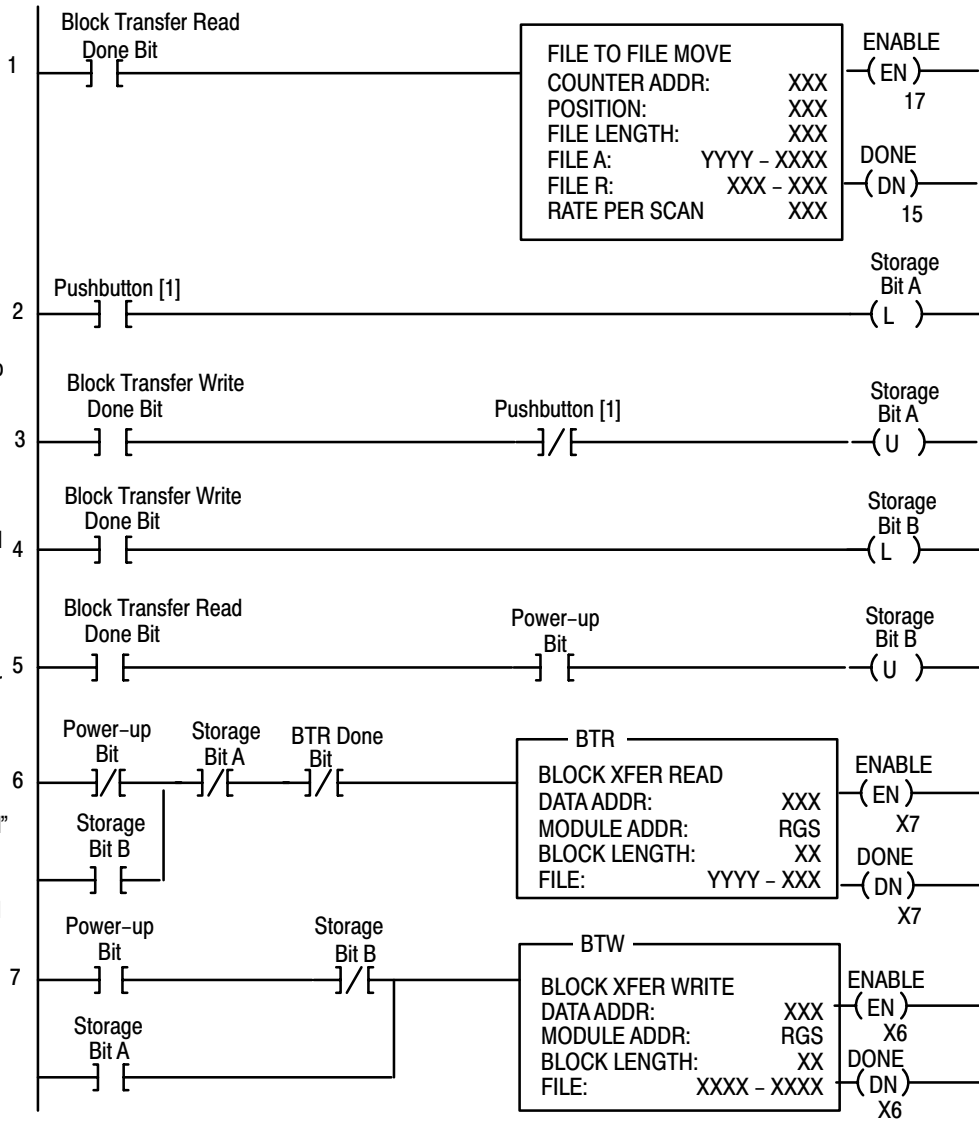
These rungs provide for a user-initiated block transfer write (BTW) after the module is initialized at power-up. Pressing the pushbutton locks out BTR operation and initiates a BTW that reconfigures the module. Block transfer writes will continue for as long as the pushbutton remains closed.

**Rungs 4 and 5**

These rungs provide a “read-write-read” sequence to the module at power-up. They also make sure that only one block transfer (read or write) is enabled during a particular program scan.

**Rungs 6 and 7**

These rungs are the conditioning block transfer rungs. Include all the input conditioning shown in the example program.



[1] You can replace the pushbutton with a timer “done” bit to initiate the block transfer write on a timed basis. You can also use any storage bit in memory.

## PLC-3 Programming

Block transfer instructions with the PLC-3 processor use one binary file in a data table section for module location and other related data. This is the block transfer control file. The block transfer data file stores data that you want transferred to your module (when programming a block transfer write) or from your module (when programming a block transfer read). The address of the block transfer data files are stored in the block transfer control file.

The industrial terminal prompts you to create a control file when a block transfer instruction is being programmed. **The same block transfer control file is used for both the read and write instructions for your module.** A different block transfer control file is required for every module.

A sample program segment with block transfer instructions is shown in Figure 3.2, and described below.

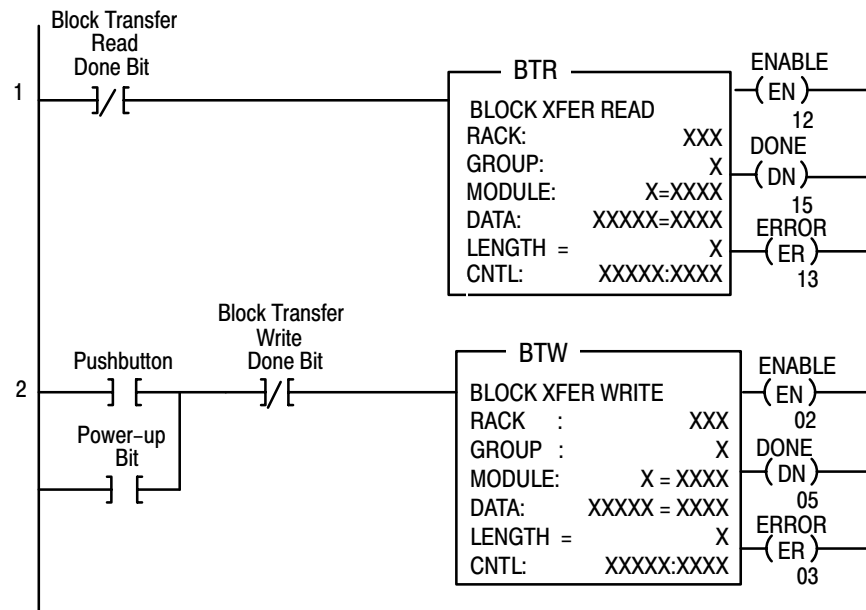
**Figure 3.2**  
**PLC-3 Family Sample Program Structure**

### Program Action

At power-up, the user program examines the BTR done bit in the block transfer read file, initiates a block transfer write to configure the module, and then does consecutive block transfer reads continuously. The power-up bit can be examined and used anywhere in the program.

### Rungs 1 and 2

Rungs 1 and 2 are the block transfer read and write instructions. The BTR done bit in rung 1, being false, initiates the first read block transfer. After the first read block transfer, the module performs a block transfer write and then does continuous block transfer reads until the pushbutton is used to request another block transfer write. After this single block transfer write is performed, the module returns to continuous block transfer reads automatically.



**PLC-5 Programming**

The PLC-5 program is very similar to the PLC-3 program with the following exceptions:

1. You must use enable bits instead of done bits as the conditions on each rung.
2. A separate control file must be selected for each of the block transfer instructions. Refer to Appendix B.

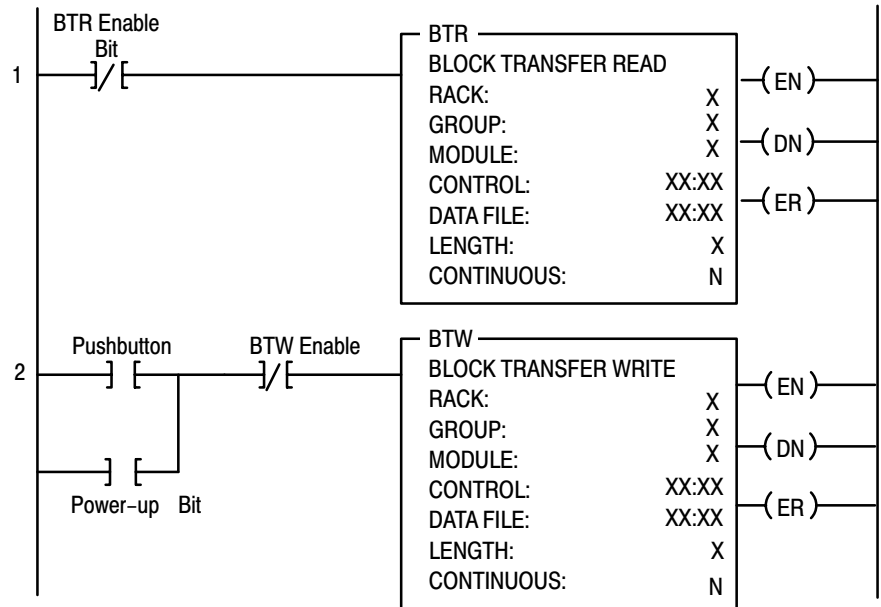
**Figure 3.3**  
**PLC-5 Family Sample Program Structure**

**Program Action**

**Rungs 1 and 2**

At power-up, the program enables a block transfer read and examines the power-up bit in the BTR file (rung 1). Then, it initiates one block transfer write to configure the module (rung 2). Thereafter, the program continuously reads data from the module (rung 1).

A subsequent BTW operation is enabled by a pushbutton switch (rung 2). Changing processor mode will not initiate a block transfer write.



10956-I

## Module Scan Time

Scan time is defined as the amount of time it takes for the input module to read the input channels and place new data into the data buffer. Scan time for your module is shown in Appendix A.

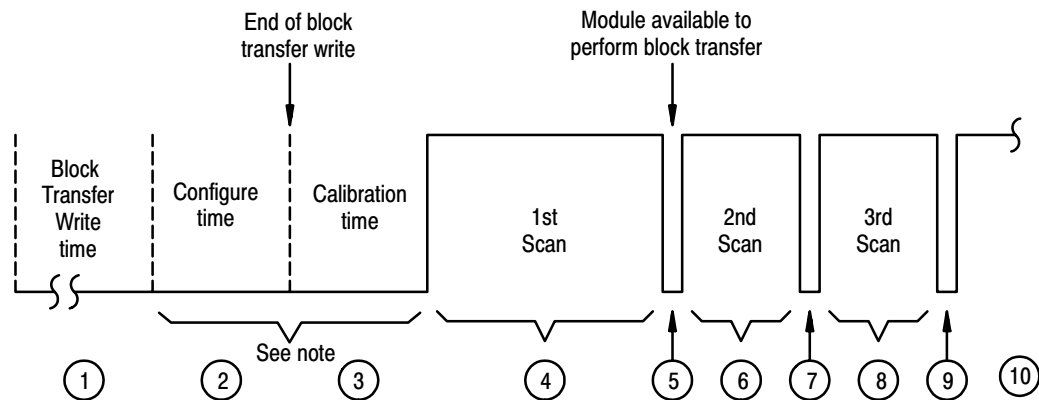
The following description references the sequence numbers in Figure 3.4.

Following a block transfer write “1” the module inhibits communication until after it has configured the data “2,” performed self-calibration “3,” scanned the inputs “4,” and filled the data buffer “5.” Write block transfers, therefore, should only be performed when the module is being configured or calibrated.

Any time after the second scan begins “6,” a BTR request “7” can be acknowledged. This interrupts the scan and the BTR empties the buffer.

Following the BTR, the input module inhibits block transfer communications with the programmable controller until it has scanned its inputs “8” and new data is ready “9.” The input module repeats the scan sequence “10,” updating the input values until another block transfer request is received. Therefore, BTRs will only be completed as frequently as the total scan time of the input module.

**Figure 3.4**  
**Block Transfer Time**



**Note: Configure/Calibration time:**

Single-ended mode = 100ms without filter, 102ms with filter  
Differential mode = 56ms without filter, 58ms with filter

**Scan time:**

= 12.5 ms for 8 differential inputs (no scaling or digital filter);  
= 25 ms for 16 single-ended inputs (no scaling or digital filter)

12689

## Chapter Summary

In this chapter, you learned how to program your programmable controller. You were given sample programs for your PLC-2, PLC-3 and PLC-5 family processors.

You also read about module scan time.

## Configuring Your Module

### Chapter Objectives

In this chapter you will read how to configure your module's features, condition your inputs and enter your data.

### Configuring Your Input Module

Because of the many analog devices available and the wide variety of possible configurations, you must configure your module to conform to the analog device and specific application that you have chosen. Data is conditioned through a group of data table words that are transferred to the module using a block transfer write instruction. Before continuing, make sure you read "Setting Module Selection Plugs" in chapter 2.

The software configurable features available with the Analog Input Module (cat. no. 1771-IFE) are:

- input range selection
- input type
- data format
- digital filtering
- real time sampling
- scaling to engineering units

**Note that digital filtering and scaling values must be entered in BCD format only. Change your display format to BCD in the PLC-5 and PLC-3 to accomplish this.**

**Note:** Programmable controllers that use 6200 software programming tools can take advantage of the IOCONFIG utility to configure this module. IOCONFIG uses menu-based screens for configuration without having to set individual bits in particular locations. Refer to your 6200 software literature for details.

**Note:** Programmable controllers that use process configuration and operation software (cat. no. 6190-PCO) can take advantage of those development and runtime tools used for the application of programmable controllers in process control. The PCO worksheets and the menu-driven configuration screens and faceplates let you configure, test/debug and operate the I/O module. Refer to your 6190-PCO software literature for details.

## Input Range Selection

You can configure the module to operate with any of five voltage or three current ranges. You can select individual channel ranges using the designated words of the write block transfer instruction (Table 4.A). Use BTW word 1 for range selection of channels 1 through 8, and BTW word 2 for channels 9 through 16. Two bits are allocated for each channel. For example, for channel 1, set word 1 bits 00-01 as shown in Table 4.A.

**Table 4.A**  
**Input Range Selection Bits**

Bit 01	Bit 00	Voltage or current input
0	0	1 to 5 V DC, 4 to 20 mA <sup>1</sup>
0	1	0 to 5 V DC, 0 to 20 mA <sup>1</sup>
1	0	-5 to +5 V DC, -20 to +20 mA <sup>1,2</sup>
1	1	-10 to +10 V DC <sup>2</sup> , 0 to 10 V DC

<sup>1</sup> Current input mode selected by configuration plug.

<sup>2</sup> Configurable using bi-polar scaling.

Table 4.B shows the incremented voltage or current assigned to each bit for the seven different input ranges. For example, if the channel 1 input range is 0 to +5V and the actual incoming signal is at mid-range (+2.5V) the value in the module's data word would be 0000 1000 0000 0000 (binary) or 2048 (decimal). The input is 2048/4096, or 1/2 of full scale.

**Table 4.B**  
**Input Voltage and Current Ranges for the Analog Input Module**

Nominal Voltage or Current Range	Corresponding 4-Digit BCD Output Range	Corresponding 12-Bit Binary Output Range	Voltage or Current Per Bit
+1 to +5V	0000 to +4095	0000 to + 4095	0.98mV
0 to 5V	0000 to +4095	0000 to +4095	1.22 mV
-5 to +5V	-4095 to +4095	-4095 to +4095	1.22mV
-10 to +10V	-4095 to +4095	-4095 to +4095	2.44mV
0 to +20mA	0000 to +4095	0000 to +4095	.0049mA
+4 to +20mA	0000 to +4095	0000 to +4095	.0039mA
-20 to +20mA	-4095 to +4095	-4095 to +4095	.0049 mA

**Note:** Voltage and current input ranges are selectable on a per channel basis.

## Input Type

You can select single-ended or differential inputs using the designated bit in the configuration file. Inputs to a particular module must be all single-ended or all differential. Set BTW word 3, bit 08 (bit 10 octal) as shown in Table 4.C.

**Table 4.C**  
**Selecting Single-ended or Differential Inputs**

Decimal Bit 8 (Octal Bit 10)	Input type
1	differential inputs
0	single-ended inputs

## Data Format

You must also indicate what format will be used to read data from your module. Typically, you select BCD with PLC-2 processors, and 2's complement binary with PLC-3 and PLC-5 processors. See Appendix C for details on data format. You use BTW word 3 bits 09-10 (11-12 octal) to set the data format (Table 4.D).

**Table 4.D**  
**Selecting the Data Format**

Decimal Bit 10 (Octal Bit 12)	Decimal Bit 09 (Octal Bit 11)	Data Format
0	0	BCD
0	1	not used
1	0	two's complement binary
1	1	signed magnitude binary



## Digital Filtering

The module has hardware-based high frequency filters on all channels to reduce the effect of electrical noise on the input signal. Software digital filtering is meant to reduce the effect of process noise on the input signal. Digital filtering is selected using BTW word 3, bits 00-07.

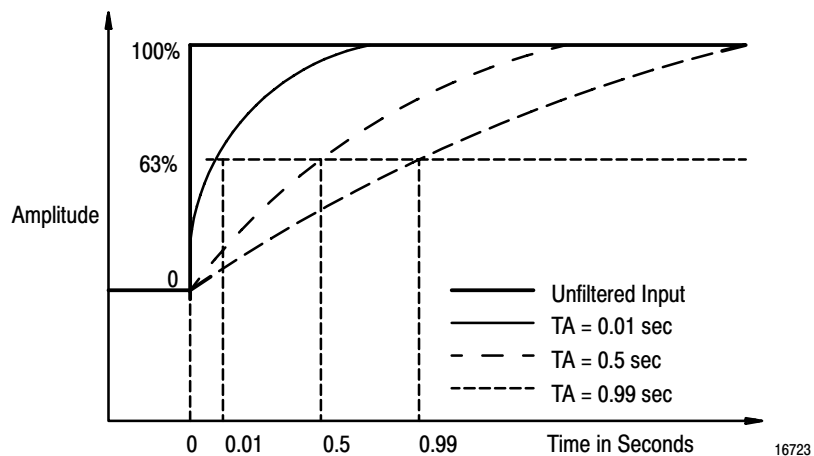
The digital filter equation is a classic first order lag equation (Figure 4.1). Using a step input change to illustrate the filter response (Figure 4.2), you can see that when the digital filter constant time elapses, 63.2% of the total response is reached. Each additional time constant achieves 63.2% of the remaining response.

**Figure 4.1**  
**Digital Filter Equation**

$$Y_n = Y_{n-1} + \left[ \frac{\Delta t}{\Delta t + TA} \right] (X_n - Y_{n-1})$$

**Where:**  
 $Y_n$  = present output, filtered peak voltage (PV)  
 $Y_{n-1}$  = previous output, filtered PV  
 $\Delta t$  = module channel update time (seconds)  
 $TA$  = digital filter time constant (seconds)  
 $X_n$  = present input, unfiltered PV

**Figure 4.2**  
**Digital Filter Lag Equation Illustration**



Digital filter time constant values of 0.00 BCD to 0.99 BCD (0.00 BCD = no filter; 0.99 BCD = maximum filter) are set in bits 00 through 07 of word 3 of the block transfer write instruction. If an invalid digital filter value is entered (i.e., 0.1F), bit 02, word 1 of the block transfer read instruction will be set. If an invalid digital filter value is entered, the module will not perform digital filtering. If you use the digital filtering feature, the filter time constant value chosen will apply to all input signals.

## Real Time Sampling

The real time sampling (RTS) mode of operation provides data gathered at precisely timed intervals for use by the processor. BTW word 3 bits 11–15 (13–17 octal) are used to set the real time sampling interval.

RTS is invaluable for time based functions (such as PID and totalization) in the PLC. It allows accurate time based calculations in local or remote I/O racks. In the RTS mode the module scans and updates its inputs at a user defined time interval (T) instead of the default interval. The module ignores block transfer read (BTR) requests for data until the sample time period elapses. The BTR of a particular data set occurs only once at the end of the sample period and subsequent requests for transferred data are ignored by the module until a new data set is available. If a BTR does not occur before the end of the next RTS period, a time-out bit is set in the BTR status area. When set, this bit indicates that at least one data set was not transferred to the processor. (The actual number of data sets missed is unknown.) The time-out bit is reset at the completion of the BTR.

Set appropriate bits in the BTW data file to enable the RTS mode. You can select RTS periods ranging from 100 milliseconds (ms) to 3.1 seconds. Refer to Table 4.E below for actual bit settings. Note that the default mode of operation is implemented by placing all zeroes in bits 11–15 (13–17 octal).

**Table 4.E**  
**Bit Settings for the Real Time Sample Mode**

Decimal Bits	15	14	13	12	11	Sample Time Period	15	14	13	12	11	Sample Time Period
Octal Bits	17	16	15	14	13		17	16	15	14	13	
	0	0	0	0	0	No RTS, Default settings	1	0	0	0	0	1.6s
	0	0	0	0	1	100ms	1	0	0	0	1	1.7s
	0	0	0	1	0	200ms	1	0	0	1	0	1.8s
	0	0	0	1	1	300ms	1	0	0	1	1	1.9s
	0	0	1	0	0	400ms	1	0	1	0	0	2.0s
	0	0	1	0	1	500ms	1	0	1	0	1	2.1s
	0	0	1	1	0	600ms	1	0	1	1	0	2.2s
	0	0	1	1	1	700ms	1	0	1	1	1	2.3s
	0	1	0	0	0	800ms	1	1	0	0	0	2.4s
	0	1	0	0	1	900ms	1	1	0	0	1	2.5s
	0	1	0	1	0	1.0s	1	1	0	1	0	2.6s
	0	1	0	1	1	1.1s	1	1	0	1	1	2.7s
	0	1	1	0	0	1.2s	1	1	1	0	0	2.8s
	0	1	1	0	1	1.3s	1	1	1	0	1	2.9s
	0	1	1	1	0	1.4s	1	1	1	1	0	3.0s
	0	1	1	1	1	1.5s	1	1	1	1	1	3.1s

Default Settings = Single-ended inputs - 25 ms  
Differential inputs - 12.5 ms

## Scaling

Your module can perform linear conversion of unscaled data to engineering units, (for example; gallons/minute, degrees C/degrees F and pounds/square inch). Unscaled data in the module has a range of : 0 through 4095 for the polar ranges (0 to 5V DC/0 to 20mA and 1 to 5V DC/4 to 20mA); and -4095 to +4095 (8190) for the bipolar ranges ( $\pm 5V/\pm 20mA$  and  $\pm 10V$ ). BTW words 6 through 37 are the scaling words for channels 1 through 16. Channel 1 minimum scaling values are set in word 6, and maximum scaling values are set in word 7. Channel 2 minimum scaling values are set in word 8, and maximum scaling values are set in word 9, and so on for the other channels.

The format of this data is 4-digit BCD or 12-bit binary. The resolution at the module of scaled values is the same as for unscaled data: one part in 4095 for 0 to 5V DC/0 to 20mA and 1 to 5V DC/4 to 20mA ranges; and one part in 8190 for the  $\pm 5V/\pm 20mA$  and  $\pm 10V$  ranges. Resolution at the processor, however, is determined by the scaled ranges (i.e., if 0 = minimum and 500 = maximum, resolution is now 1 part in 500). Each input channel can be scaled independently of the other channels.

**Note:** To achieve the 0 to +10V range you must use bipolar scaling. Select the  $\pm 10V$  range and scale for  $\pm$  the actual intended range. If you need 0 to 100 gpm, set scaling values at -100 and +100. You will effectively be creating a 0 to 10V range that is scaled from 0 to 100.

### Implementing the Scaling Feature

You implement the scaling feature by:

1. Inserting minimum and maximum scaled values in the appropriate configuration words
2. If any of the minimum or maximum values are negative, set the appropriate sign bits in the minimum or maximum sign bit word
3. If a single channel is scaled, all channels must be scaled, and all 37 configuration words must be written to the module.

### Scaling Ranges

The maximum range of the scaling values is  $\pm 9999$  BCD. **These values must be entered in BCD.**

Typically, invalid values are “minimum greater than maximum,” or “minimum equal to maximum.” **If invalid values are entered into the scaling words, the corresponding input in the BTR data will be zero and the invalid scaling bit will be set.**

**Important: Scaling values must always be entered in BCD format,** even if the data format chosen is binary. If scaling is selected for any channel, all channels must be scaled. If scaling is not required on certain channels, set those to the default input range: 0 to 4095 for 0 to + voltage or current ranges, and -4095 to +4095 for - to + voltage or current ranges.

**If scaling is not selected, the module requires specific minimum BTR file lengths for the number of channels used.** The BTW file length can be set to 3 words. Table 4.F shows the required BTW and BTR file lengths.

**Table 4.F**  
**Block Transfer Read and Write File Lengths**

Channels Used	BTR File Length	BTW File Length
1	5	7
2	6	9
3	7	11
4	8	13
5	9	15
6	10	17
7	11	19
8	12	21
9	13	23
10	14	25
11	15	27
12	16	29
13	17	31
14	18	33
15	19	35
16	20	37

**Important:** Use decimally addressed bit locations for PLC-5 processors.

**Default Configuration**

If a write block of five words, with all zeroes, is sent to the Analog Input Module (cat. no. 1771-IFE), default selections will be:

- 1 to 5V DC or 4 to 20mA (dependent on configuration jumper setting)
- BCD data format
- no real time sampling (RTS)
- no filtering
- no scaling
- single-ended inputs

**Figure 4.3**  
**Analog Input Module (1771-IFE) Block Transfer Write Configuration Block**

Decimal Bits	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00	Description
Octal Bits	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00	
Word 1	8		7		6		5		4		3		2		1		Range Selection - Channels 1 thru 8
2	16		15		14		13		12		11		10		9		Range Selection - Channels 9 thru 16
3	Real Time Sampling				Data Format		Input Type		Digital Filter								Real time sampling, data format, input type and digital filter
4	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Sign Bits, minimum scaling values
5	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Sign Bits, maximum scaling values
6																	Channel 1 - minimum scaling
7																	Channel 1 - maximum scaling
8																	Channel 2 - minimum scaling
9																	Channel 2 - maximum scaling
10																	Channel 3 - minimum scaling
<div style="display: flex; justify-content: space-around;"> <span>⇓</span> <span>⇓</span> <span>⇓</span> <span>⇓</span> </div>																	
37																	Channel 16 - maximum scaling

### Bit/Word Descriptions for the Analog Input Module Block Transfer Write Configuration Block

Note that decimal bits are shown, with octal bits in parentheses.

Word	Decimal Bit (Octal Bit)	Description
Word 1 and 2	Bits 00-15 (00-17)	Input range selections allow the user to configure the inputs for any of 7 input voltage or current ranges. Two bits are required for each channel. See Table 4.A.
Word 3	Bits 00-07 (00-07)	Digital filter reduces effect of noise on input. See "Digital Filtering" on page 4-4.
	Bit 08 (10)	Input type, set bit for differential mode on all channels. Reset (0) = single-ended. Refer to Table 4.C
	Bits 09-10 (07-10)	Data format matches format of processor. See Table 4.D.
	Bits 11-15 (13-17)	Real time sampling will default to 12.5ms for differential mode and 25ms for single-ended, greater with filtering selected. See appendix A for timing details. See Table 4.E for other real time intervals.
Word 4	Bits 00-15 (00-17)	Minimum sign bits, when set, designate negative minimum scaling values for the corresponding input channels. Bit 00 corresponds to channel 1, bit 01 corresponds to channel 2, etc.
Word 5	Bits 00-15 (00-17)	Maximum sign bits, when set, designate maximum scaling values that are negative. Maximum scaling value must be greater than minimum on any particular channel. Bit 00 corresponds to channel 1, bit 01 corresponds to channel 2, etc.
Words 6-37	Bits 00-15 (00-17)	Minimum and maximum scaling values for each channel. Enter in BCD format.

### Chapter Summary

In this chapter you learned how to configure your module's features, condition your inputs and enter your data.

## Module Status and Input Data

### Chapter Objectives

In this chapter you will read about:

- reading data from your module
- block transfer read block format

### Reading Data From Your Module

Block transfer read programming moves status and data from the input module to the processor's data table in one I/O scan (Figure 5.1). The processor's user program initiates the request to transfer data from the input module to the processor.

**Figure 5.1**  
Word Assignments for Analog Input Module (1771-IFE)  
Block Transfer Read

Decimal Bits	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00	Description
Octal Bits	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00	
Word 1	Not Used											Diagnostic Bits				Diagnostics	
2	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Data underrange for channels 1-16 <sup>1</sup>
3	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Data overrange for channels 1-16 <sup>1</sup>
4	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Data polarity for channels 1-16
5	Channel 1 Input											Channel 1 Input					
6	Channel 2 Input											Channel 2 Input					
7	Channel 3 Input											Channel 3 Input					
8	Channel 4 Input											Channel 4 Input					
				↓						↓							↓
20	Channel 16 Input											Channel 16 Input					

<sup>1</sup> These bits are set (1) at approximately the input range limits selected (Table 5.B).

## Block Transfer Read Format

The bit/word description for the block transfer read of the Analog Input Module is described below in Table 5.A.

**Table 5.A**  
**BTR Word Format for the Analog Input Module**

Word	Decimal Bit (Octal Bit)	Description
Word 1	Bit 00	<b>Power up bit</b> is used by the module to tell the processor that it is alive but not yet configured. It is a key element in the application program.
	Bit 01	<b>Out of range bit</b> is sent to tell the processor that one or more channels are either over or under range. <sup>1</sup>
	Bit 02	<b>Invalid scaling bit</b> reports that the scaling is somehow invalid. Usually, both values are equal or minimum is greater than maximum when this bit comes on. Can also be an <b>invalid filter</b> value.
	Bit 03	Real time sample fault bit. This bit is set if the module is configured for RTS and a block transfer read has not occurred within the user-programmed period.
Word 2	Bits 00–15 (00–17)	Individual <b>underrange bits</b> for each channel. Bit 00 for channel 1, bit 01 for channel 2, etc. <sup>1</sup> These bits are set (1) at approximately the input range limits selected from Table 5.B.
Word 3	Bits 00–15 (00–17)	Individual <b>overrange bits</b> for each channel. Bit 00 for channel 1, bit 01 for channel 2, etc. <sup>1</sup> These bits are set (1) at approximately the input range limits selected from Table 5.B.
Word 4	Bits 00–15 (00–17)	<b>Polarity bits</b> are set when input is less than zero. Bit 00 for channel 1, bit 01 for channel 2, etc.
Word 5 thru 20		<b>Input values.</b> Word 5 for channel 1, word 6 for channel 2, etc.

<sup>1</sup> **Attention:** If an input terminal's voltage exceeds  $\pm 14.25V$  as referenced to module common, channel-to-channel crosstalk can cause invalid input readings and invalid underrange/overrange bits.

**Table 5.B**  
**Input Range Selection**

Voltage input	Current input <sup>1</sup>
1 to 5V dc	4 to 20mA
0 to 5V dc	0 to 20mA <sup>1</sup>
-5 to +5V dc	-20 to +20mA <sup>2</sup>
-10 to +10V dc <sup>2</sup>	
0 to 10V dc	

<sup>1</sup> Current input mode selected by configuration plug.  
<sup>2</sup> Configurable using bi-polar scaling.

## Chapter Summary

In this chapter you learned the meaning of the status information that the input module sends to the processor.



## Calibrating Your Module

### Chapter Objectives

In this chapter we tell you what tools you need and how to calibrate your module.

### Tools and Equipment

In order to calibrate your input module you will need the following tools and equipment:

Equipment	Description
Digital voltmeter	5-1/2 digit, 0.01% accuracy minimum: Keithley191 or Fluke 8300A or equivalent
Alignment tool	P/n 35F616, for pot adjustment: Newark Electronics, 500 N.Pulaski Rd., Chicago, IL
Potentiometer sealant	Torque Seal: Organic Products, P.O. Box 928, Irving, TX
Industrial terminal	Cat. no. 1770-T3 and program panel interconnect cable (cat. no. 1772-TC) for PLC-2 family processors: Allen-Bradley, Highland Hts., OH
Backplane extender card	Cat. no. 1771-EZ

### Calibration Procedure

The analog input module is shipped from the factory **already calibrated**. If necessary to recalibrate the module, you must calibrate the module in an I/O chassis. The module must communicate with the processor and industrial terminal. Calibration consists of adjusting the 10V reference and nulling the input offset.

**Important:** The module must be powered up for at least 30 minutes before attempting to calibrate.



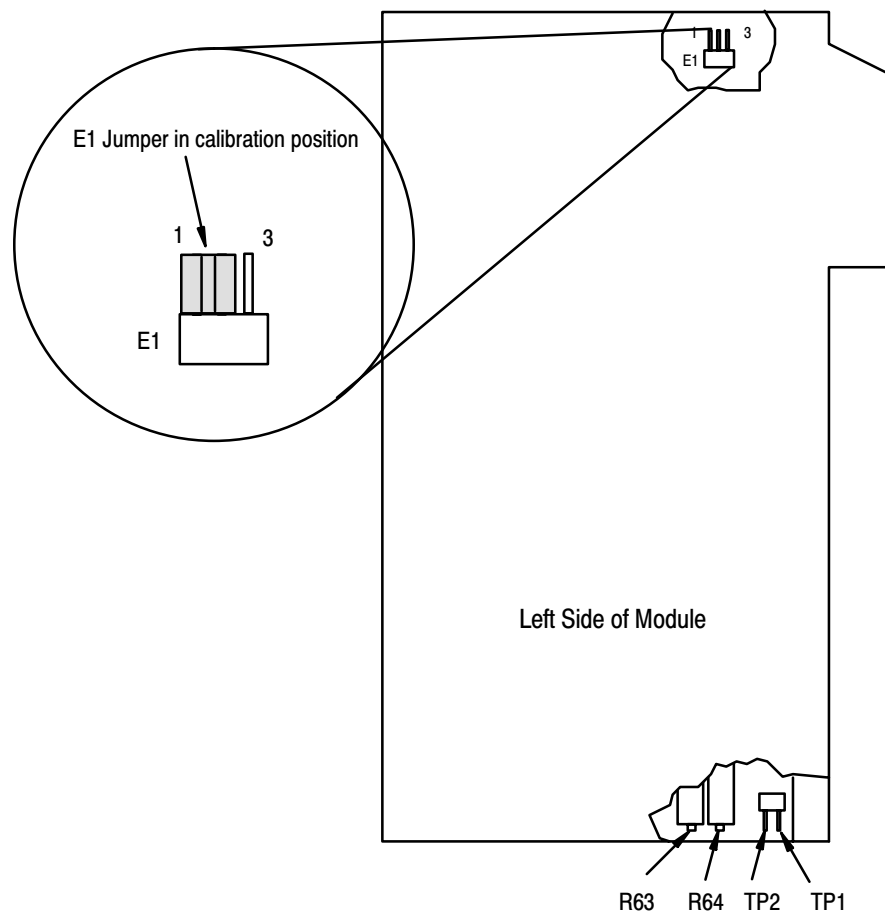
**ATTENTION:** Do not attempt to calibrate your module until you have read and thoroughly understand this procedure. Also, do not attempt to calibrate this module in an operating system. Damage to the equipment or personal injury may result.

---

### Adjusting the 10V Reference

1. Turn off power to your processor and I/O chassis.
2. Swing the field wiring arm out of the way.
3. Remove the module from the I/O chassis.
4. Plug the module into the extender card, and insert the extender card into the I/O chassis.
5. Attach the negative lead to an analog common (pin 5, 10, 15, 20 or 21) of the wiring arm.
6. Attach the positive lead of your voltmeter to TP1.
7. To set the on-board +10V reference, adjust potentiometer R64 (Figure 6.1) until the value at TP1 = 10.0000V ( $\pm 0.0002$ V maximum).

**Figure 6.1**  
Test Points and Potentiometers for Analog Input Module (1771-IFE)



10956-1

### **Nulling the Input Offset**

After completing the 10V reference adjustment, turn off power to your processor and I/O chassis and complete the following steps.

1. Move jumper E1 (Figure 6.1) from the default position (connecting the center and right posts) to the calibration position (connecting the center and left posts).
2. Attach the negative lead of your voltmeter to an analog common (pin 5, 10, 15, 20 or 21) of the field wiring arm.
3. Attach the positive lead of your voltmeter to TP2.
4. Turn on power to your processor and I/O chassis. Check to make sure the red FLT indicator is lit and the green RUN indicator is off. If the red indicator is off, check the position of E1.
5. Adjust potentiometer R63 (Figure 6.1) until the value at TP2 = 0.0000V ( $\pm 0.0002$ V maximum).
6. After completing the adjustment, remove power from the I/O chassis and return jumper E1 to the default position.

### **Chapter Summary**

In this chapter you learned how to calibrate your module. This included the necessary tools, adjusting the 10V reference, and nulling the offset.

## Troubleshooting Your Input Module

### Chapter Objective

In this chapter, we describe how to troubleshoot your module by observing the indicators and by monitoring status bits reported to the processor.

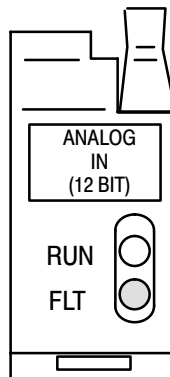
### Diagnostics Reported by the Module

At power-up, the module momentarily turns on the red indicator as a lamp test, then checks for:

- correct RAM operation
- firmware errors

Thereafter, the module lights the green RUN indicator when operating without fault, or lights the red FAULT indicator when it detects fault conditions. The module also reports status and specific faults (if they occur) in every transfer of data (BTR) to the PC processor. Monitor the green and red indicators and status bits in word 1 of the BTR file when troubleshooting your module.

**Figure 7.1**  
**Diagnostic Indicators**



10528-1

### Diagnostic Bits Reported By the Analog Input Module

Diagnostic bits in the read block transfer status words provide diagnostic capabilities.

**Word 1** provides power-up and valid data status. **Words 2, 3 and 4** provide channel data status.

If a module on-board self test fault occurs, block transfers will be inhibited, the red fault (FLT) will light, and the green run (RUN) light will go out.

### **Word 1**

Diagnostics word 1 is the first data word in the read block transfer file for transfer to the central processor. It contains a power-up bit (bit 00) that is set (1) when the module is first powered up. It is reset (0) after a write block transfer. It also contains an under-range or over-range bit (bit 01) that is set when any input is under or over-range.

An invalid scaling data bit (bit 02) will be set if invalid scaling data is entered into any of the minimum/maximum scaling value words. **Note that minimum equal to maximum is an invalid value.** If invalid values are entered into the minimum or maximum scaling words the corresponding read block transfer input channel word will be set to 0000.

Bit 02 will also be set if an invalid digital filter value is entered (e.g., 1F). If an invalid digital filter value is entered, the module will not perform digital filtering.

The real time sample (RTS) fault bit (bit 03) is set if the module is configured for RTS and a block transfer read has not occurred within the user-programmed period.

### **Word 2**

Word 2 provides for under-range conditions. When a particular channel input is under-range, the associated bit will be set. As long as inputs are under range the associated bit will remain set. Bit 00 corresponds to channel 1, bit 01 to channel 2, etc.

### **Word 3**

Word 3 provides for over-range conditions. When a particular channel input is over-range, the associated bit will be set. As long as inputs are in range the associated bit will remain reset. Bit 00 corresponds to channel 1, bit 01 to channel 2, etc.

### **Word 4**

Word 4 provides an indication of a particular channel's input polarity (set, or 1 = negative; reset, or 0 = positive). Bit 00 corresponds to channel 1, bit 01 to channel 2, etc.

Table 7.A lists the probable cause and recommended actions for a number of common trouble indications.

**Table 7.A**  
**Troubleshooting Chart for Analog Input Module (1771-IFE)**

**Legend**

- Off
- On

Indicators	Probable Cause	Recommended Action
● RUN (green) ○ FLT (red)	Normal operation	None
○ RUN (green) ● FLT (red)	If out-of-range bit is set (BTR word 1, bit 02) and all 8 under-range bits are set (BTR word 2, bits 00 through 07).	Return module for repair
	If incorrect data in final storage word locations in processor's data table, possible severed or disconnected input cable associated with the affected channels.  or  Input module is conditioned for BCD instead of binary or the reverse, incorrect scaling, sign bits missing, wrong range.	Repair/replace cable.  Condition module for desired format (BCD or binary), enter correct data and initiate another write block transfer.  If module connections are intact and configuration data is correct check calibration procedure.
	Hardware failure in module	Return module for repair.
○ RUN ○ FLT    Neither LED comes on	No power PICO fuse is bad.	Turn off power. Remove and reinsert module into chassis. Return power. If problem still exists, and chassis power supply is functioning properly, return the module for repair.

**Chapter Summary**

In this chapter you learned how to interpret the indicator lights, and troubleshoot your input module.

## Specifications

Inputs per module	16 single-ended; 8 differential low level
Module Location	1771 I/O rack - 1 slot
Input voltage ranges (nominal)	+1 to +5V dc 0 to 5V dc -5 to +5V dc -10 to +10V dc 0 to +10V dc
Input current ranges (nominal)	+4 to +20mA 0 to +20mA -20 to +20mA
Resolution	12-bit binary 12 bits plus sign on bipolar ranges
Accuracy	0.1% of full scale range @ 25°C
Linearity	±1 LSB
Repeatability	±1 LSB
Isolation Voltage	±1500V, (transient)
Input overvoltage protection	200V (voltage mode) <sup>1</sup> 8V (current mode) <sup>2</sup>
Input overcurrent protection (current ranges)	30mA
Common mode voltage	±14.25 Volts
Input impedance	100 Megohms for voltage ranges; 250 ohms for current ranges
Common mode rejection	80 db, DC-120 Hz
Current Requirements	0.75A @ +5V from I/O chassis backplane
Power Dissipation	3.75 Watts (maximum)
Thermal Dissipation	12.8 BTU/hr (maximum)
Unscaled BCD and binary output to processor	0000 to +4095 <sub>10</sub> for polar ranges (0 to 5V, +1 to +5V, 0 to +20mA, and +4 to +20mA) -4095 <sub>10</sub> to 4095 <sub>10</sub> for bipolar ranges ( ±5V, ±10V, ±20mA)
Engineering units sent to processor	±9999 <sub>10</sub> with selectable scaling
Internal scan rate	12.5 ms for 8 differential inputs (no digital filtering) -add 2.12ms for filtering 25 ms for 16 single-ended input (no digital filtering) -add 4.24 for filtering
Environmental conditions operational temperature: storage temperature: relative humidity:	0 to 60°C (32 to 140°F) -40 to 85°C (-40 to 185°F) 5 to 95% (without condensation)
Conductors Wiring Category	14 gauge stranded (max.) 3/64 inch insulation (max.) Category 2 <sup>3</sup>
Keying	between 10 and 12 between 24 and 26
Wiring Arm	Catalog Number 1771-WG
Field Wiring Arm Screw Torque	7-9 inch-pounds

<sup>1</sup> The inputs are protected to 200V. However, if an input terminal's voltage exceeds ±14.25V as referenced to module common, channel-to-channel crosstalk can cause invalid input readings and invalid underrange/overrange bits.

<sup>2</sup> Only 8 volts can be placed directly across the input when configured in the current mode.

<sup>3</sup> Refer to publication 1770-4.1, "Programmable Controller Wiring and Grounding Guidelines."

## Programming Examples

### Sample Programs for the Analog Input Module

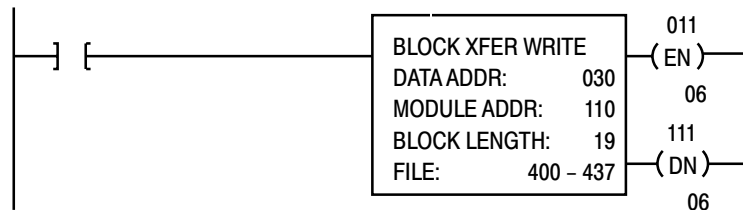
The following are sample programs for entering data in the configuration words of the write block transfer instruction when using the PLC-2, PLC-3 or PLC-5 family processors.

### PLC-2 Family Processors

To enter data in the configuration words, follow these steps:

**Example:**

Enter the following rung for a write block transfer:



400 is the address of the write block transfer data file. You want to examine configuration word 1.

Step	Action	Description
1.	Press [SEARCH]8<data address>	Finds the block transfer instruction
2.	Press CANCEL COMMAND	Removes preceding command
3.	Press [DISPLAY]0 or 1	Displays the file in binary or BCD
4.	Move cursor to data to be modified	
5.	Enter new data	
6.	Press [INSERT]	Writes data to file element

Use the above procedure to enter the required words of the write block transfer instruction. Be aware that the block length will depend on the number of channels selected and whether scaling is or is not performed; for example, the block may contain only 3 words if no scaling is performed but may contain 37 words if using 16 inputs with scaling. The PLC-2 family write block transfer data file should look like Figure B.1.



**Figure B.1**  
**Write Block Transfer Data Transfer for a PLC-2 Family Processor**

DATA ADDR: 030	BINARY DATA MONITOR BLOCK TRANSFER WRITE MODULE ADDR: 110 FILE: 400-444	BLOCK LENGTH: 37
POSITION	FILE DATA	
001	00000000	00000000
002	00000000	00000000
003	00000000	00000000
004	00000000	00000000
005	00000000	00000000
006	00000000	00000000
007	00000000	00000000
008	00000000	00000000
009	00000000	00000000
010	00000000	00000000
011	00000000	00000000
012	00000000	00000000
013	00000000	00000000
014	00000000	00000000
015	00000000	00000000
DATA	00000000	00000000

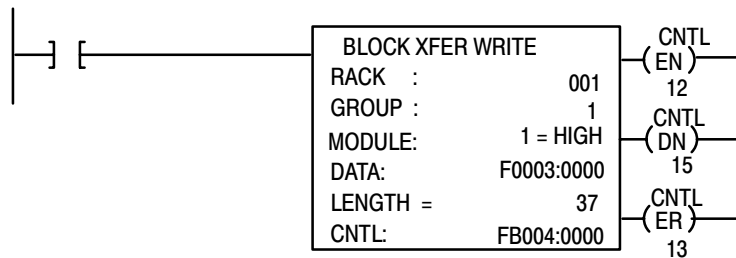
### PLC-3 Family Processor

Following is a sample procedure for entering data in the configuration words of the write block transfer instruction when using a PLC-3 processor.

To enter data in the configuration words, follow these steps:

**Example:**

Enter the following rung for a write block transfer:



F0003:0000 is the address of the write block transfer data file. You want to enter/examine word 1.

1. Press [SHIFT][MODE] to display your ladder diagram on the industrial terminal.
2. Press DD, 03:0[ENTER] to display the block transfer write file.

The industrial terminal screen should look like Figure B.2. Notice the highlighted block of zeroes. This highlighted block is the cursor. It should be in the same place as it appears in Figure B.2. If it is not, you can move it to the desired position with the cursor control keys. Once you have the highlighted cursor in the right place as shown above, you can go on to step 3.

**Figure B.2**  
**Write Block Transfer for a PLC-3 Processor**

```
START - W0003 : 0000

WORD
000000      00000000  00000000  00000000  00000000
000004      00000000  00000000  00000000  00000000
000010      00000000  00000000  00000000  00000000
000014      00000000  00000000  00000000  00000000
000020      00000000  00000000  00000000  00000000

DATA MONITOR          $ W0310- []

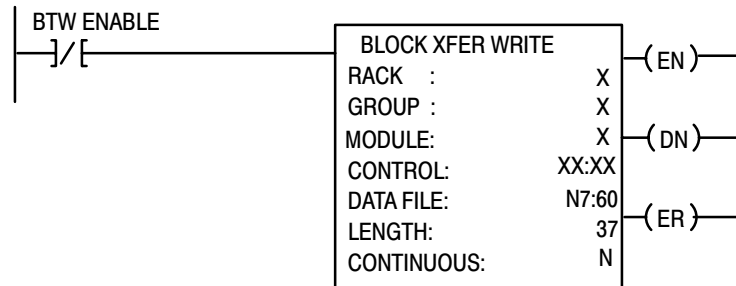
PROG : I/O OFF : NO FORCES : NO EDITS : RUNG # [RM000000 : MEM PORT OFF
```

3. Enter the data corresponding to your bit selection in word 0 through 4.
4. When you have entered your data, press [ENTER]. If you make a mistake, make sure the cursor is over the word you desire to change. Enter the correct data and press [ENTER].
5. Press [CANCEL COMMAND]. This returns you to the ladder diagram.

## PLC-5 Family Processors

The following is a sample procedure for entering data in the configuration words of the block transfer write instruction when using a PLC-5 processor and 6200 programming software.

1. Enter the following rung:



N7:60 is the address of the BTW transfer file

2. Press [F8] (data monitor),[F5] (change address) and enter N7:60 to display the configuration block.

The industrial terminal screen should look like Figure B.3.

**Figure B.3**  
**Sample PLC-5 Data File (Hexadecimal Data)**

ADDRESS	0	1	2	3	4	5	6	7	8	9
N7:60	5003	00FF	00FF	0040	0085	0040	0085	0040	0085	0040
N7:70	0085	0040	0085	0040	0085	0040	0085	0040	0085	0000
N7:80	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
N7:90	0000	0000	0000	0000	0000	0000	0000	0000		

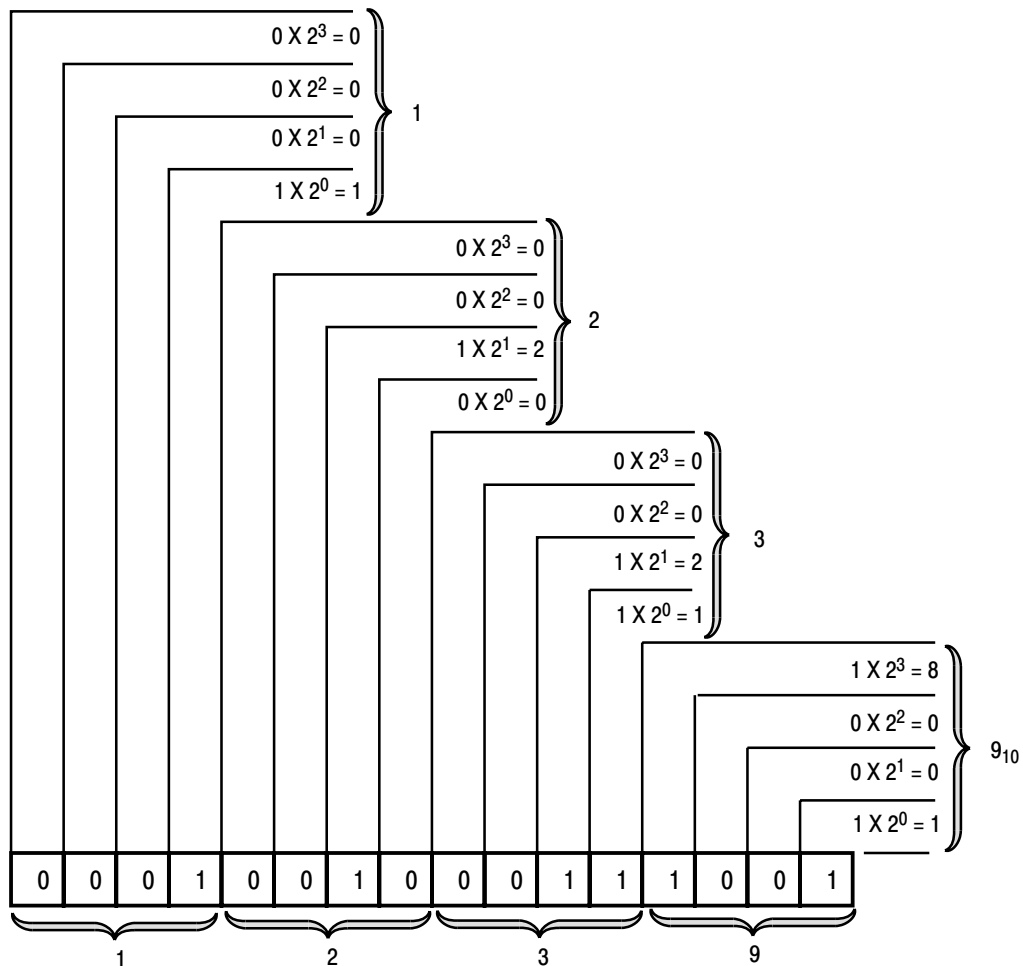
3. Enter the data corresponding to your bit selections and add scaling values, if scaling is desired.
4. [ESC] returns you to the ladder program.

## Data Table Formats

### 4-Digit Binary Coded Decimal (BCD)

The 4-digit BCD format uses an arrangement of 16 binary digits to represent a 4-digit decimal number from 0000 to 9999 (Figure C.1). The BCD format is used when the input values are to be displayed for operator viewing. Each group of four binary digits is used to represent a number from 0 to 9. The place values for each group of digits are  $2^0$ ,  $2^1$ ,  $2^2$  and  $2^3$  (Table C.A). The decimal equivalent for a group of four binary digits is determined by multiplying the binary digit by its corresponding place value and adding these numbers.

**Figure C.1**  
4-Digit Binary Coded Decimal



12955-1

**Table C.A**  
**BCD Representation**

2 <sup>3</sup> (8)	Place Value			Decimal Equivalent
	2 <sup>2</sup> (4)	2 <sup>1</sup> (2)	2 <sup>0</sup> (1)	
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9

### Signed-magnitude Binary

Signed–magnitude binary is a means of communicating numbers to your processor. It should be used with the PLC-2 family when performing computations in the processor. It cannot be used to manipulate binary 12-bit values or negative values.

**Example:** The following binary number is equal to decimal 22.

$$10110_2 = 22_{10}$$

The signed–magnitude method places an extra bit (sign bit) in the left–most position and lets this bit determine whether the number is positive or negative. The number is positive if the sign bit is 0 and negative if the sign bit is 1. Using the signed magnitude method:

$$0\ 10110 = +22$$

$$1\ 10110 = -22$$

## Two's Complement Binary

Two's complement binary is used with PLC-3 processors when performing mathematical calculations internal to the processor. To complement a number means to change it to a negative number. For example, the following binary number is equal to decimal 22.

$$10110_2 = 22_{10}$$

First, the two's complement method places an extra bit (sign bit) in the left-most position, and lets this bit determine whether the number is positive or negative. The number is positive if the sign bit is 0 and negative if the sign bit is 1. Using the complement method:

$$0\ 10110 = 22$$

To get the negative using the two's complement method, you must invert each bit from right to left after the first "1" is detected.

In the above example:

$$0\ 10110 = +22$$

Its two's complement would be:

$$1\ 01010 = -22$$

Note that in the above representation for +22, starting from the right, the first digit is a 0 so it is not inverted; the second digit is a 1 so it is not inverted. All digits after this one are inverted.

If a negative number is given in two's complement, its complement (a positive number) is found in the same way:

$$1\ 10010 = -14$$

$$0\ 01110 = +14$$

All bits from right to left are inverted after the first "1" is detected.

The two's complement of 0 is not found, since no first "1" is ever encountered in the number. The two's complement of 0 then is still 0.

## Block Transfer (Mini-PLC-2 and PLC-2/20 Processors)

### Multiple GET Instructions - Mini-PLC-2 and PLC-2/20 Processors

Programming multiple GET instructions is similar to block format instructions programmed for other PLC-2 family processors. The data table maps are identical, and the way information is addressed and stored in processor memory is the same. The only difference is in how you set up block transfer read instructions in your program.

For multiple GET instructions, individual rungs of ladder logic are used instead of a single rung with a block transfer instruction. A sample rung using multiple GET instructions is shown in Figure D.1 and described in the following paragraphs.

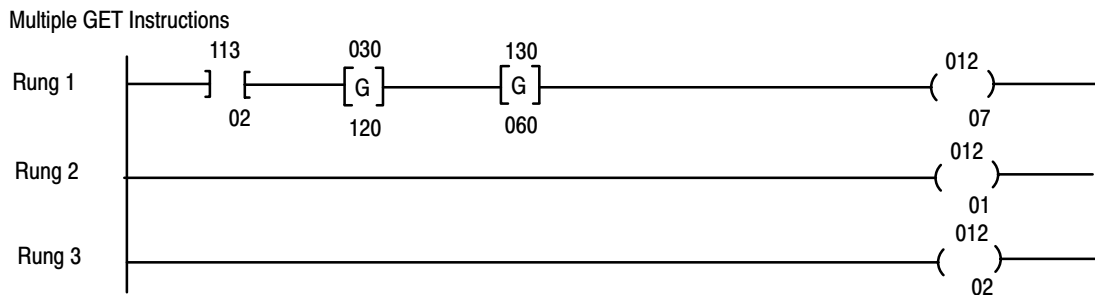
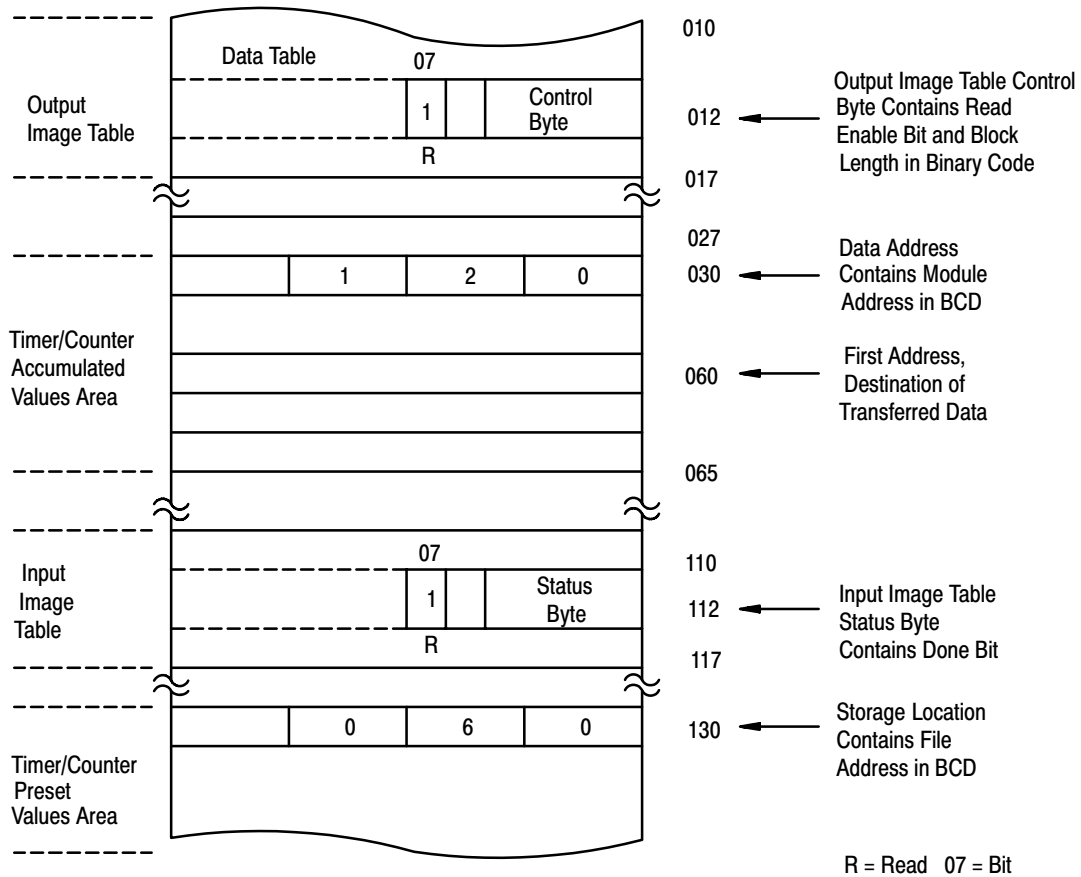
**Rung 1:** This rung is used to set four conditions.

- **Examine On Instruction (113/02)** - This is an optional instruction. When used, block transfers will only be initiated when a certain action takes place. If you do not use this instruction, block transfers will be initiated every I/O scan.
- **First GET Instruction (030/120)** - identifies the module's physical address (120) by rack, group and slot; and where in the accumulated area of the data table this data is to be stored (030).
- **Second GET Instruction (130/060)** - indicates the address of the first word of the file (060) that designates where the data will be transferred. The file address is stored in word 130, 100<sub>8</sub> above the data address.
- **Output Energize Instruction (012/07)** - enables the block transfer read operation. If all conditions of the rung are true, the block transfer read enable bit (07) is set in the output image data table control byte. The output image table control byte contains the read enable bit and the number of words to be transferred. The output energize instruction is defined as follows:
  - "0" indicates that it is an output instruction
  - "1" indicates the I/O rack address
  - "2" indicates the module group location within the rack
  - "07" indicates this is a block transfer read operation (if this were a block transfer write operation, "07" would be replaced by "06".)

**Rungs 2 and 3:** These output energize instructions (012/01 and 012/02) define the number of words to be transferred. This is accomplished by setting a binary bit pattern in the module's output image table control byte. The binary bit pattern used (bits 01 and 02 energized) is equivalent to 6 words or channels, and is expressed as 110 in binary notation.

**Rung Summary:** Once the block transfer read operation is complete, the processor automatically sets bit 07 in the input image table status byte and stores the block length of the data transferred.

**Figure D.1**  
**Multiple GET Instructions (Mini-PLC-2 and PLC-2/20 Processors Only)**





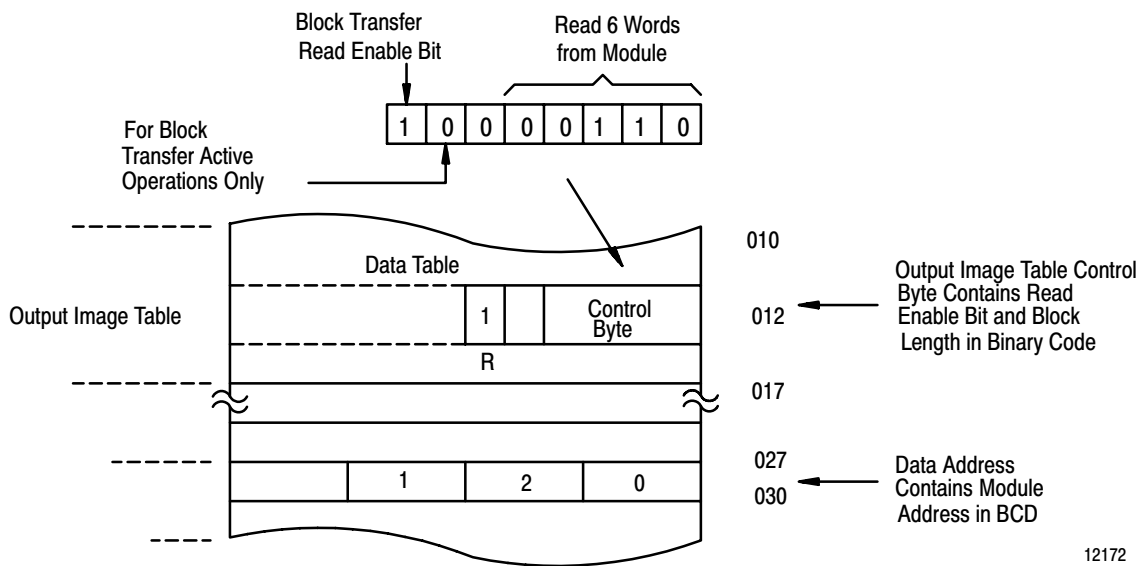
## Setting the Block Length (Multiple GET Instructions only)

The input module transfers a specific number of words in one block length. The number of words transferred is determined by the block length entered in the output image table control byte corresponding to the module's address.

The bits in the output image table control byte (bits 00 - 05) must be programmed to specify a binary value equal to the number of words to be transferred.

For example, Figure D.2 shows if your input module is set up to transfer 6 words, you would set bits 01 and 02 of the lower image table control byte. The binary equivalent of 6 words is 000110. You would also set bit 07 when programming the module for block transfer read operations. Bit 06 is used when block transfer write operations are required.

**Figure D.2**  
 Setting Block Length (Multiple GET Instructions only)



12172

Number of Words to Transfer	Binary Bit Pattern Lower Output Image Table Byte					
	05	04	03	02	01	00
Default	0	0	0	0	0	0
1	0	0	0	0	0	1
2	0	0	0	0	1	0
3	0	0	0	0	1	1
4	0	0	0	1	0	0
5	0	0	0	1	0	1
6	0	0	0	1	1	0
:						:
18	0	1	0	0	1	0
19	0	1	0	0	1	1

## Forms

This appendix contains forms useful in setting up your data table.

## Analog Block Transfer Read

Position	Decimal File Word Octal	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00	
		17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00	
1		Bits not used												Diagnostic bits		Power Up Bit		
2																		Data Underrange
3																		Data Overrange
4																		Data Polarity "0" = (+) "1" = (-)

Position	File Word	Channel Number	Value
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			

## Analog Block Transfer Write

Position	Decimal File Word Octal	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00		
		17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00		
		Channels 1 through 8 Range Selection															Power Up Bit		
		8	7	6	5	4	3	2	1										Channel Number
1																			
		Channels 9 through 16 Range Selection																	
		16	15	14	13	12	11	10	9										Channel Number
2																			
Position	Decimal File Word Octal	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00		
		17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00		
3																		Module Configuration	
4																		Minimum scaling value sign bits	
5																		Maximum scaling value sign bits	

Position	File Word	Channel Number	Min./Max Value	Position	File Word	Channel Number	Min./Max Value
6				22			
7				23			
8				24			
9				25			
10				26			
11				27			
12				28			
13				29			
14				30			
15				31			
16				32			
17				33			
18				34			
19				35			
20				36			
21				37			

## B

BCD, [4-1](#)  
BCD format, [1-3](#), [4-7](#)  
    digital filtering and scaling, [4-1](#)  
block transfer, [1-1](#), [1-3](#), [2-2](#), [7-1](#)  
    communication using, [1-2](#)  
    read and write file lengths, [4-7](#)  
    write, [1-2](#)  
block transfer read, [3-1](#), [3-2](#), [5-1](#), [7-2](#)  
    bit/word format, [5-2](#)  
    word assignments, [5-1](#)  
block transfer write, [3-2](#)  
    configuration block, [4-8](#)  
    filter settings, [4-4](#)  
    input range selection, [4-2](#)  
    programming, [3-1](#)

## C

calibration  
    adjusting the 10V reference, [6-2](#)  
    nulling the offset, [6-3](#)  
    tools and equipment, [6-1](#)  
calibration procedure, [6-1](#)  
compatibility, [P-2](#)  
configuration block  
    bit/word descriptions, [4-9](#)  
    block transfer write, [4-8](#)  
configuration plugs, setting  
    differential or single-ended voltage and current inputs, [2-10](#)  
    differential voltage and current inputs, adjacent channels, [2-11](#)  
    single-ended voltage and current inputs, adjacent channels, [2-11](#)  
configuration/calibration time, [3-5](#)  
connection diagram  
    16 single-ended inputs  
        2-wire transmitters, [2-4](#)  
        4-wire transmitters, [2-5](#)  
    8 differential inputs  
        2-wire transmitters, [2-6](#)  
        4-wire transmitters, [2-7](#)  
considerations, pre-installation, [2-1](#)

## D

data format, [4-3](#)  
    bit selection settings, [4-3](#)  
data formats  
    2's complement binary, [C-3](#)  
    4-digit binary coded decimal, [C-1](#)  
    signed-magnitude binary, [C-2](#)  
data table usage, [P-2](#)  
default configuration, block transfer write, [3-1](#), [4-8](#)  
diagnostic bits, [7-1](#)  
diagnostics, word 1, [7-2](#)

## F

factory setting, inputs, [2-3](#)  
fault indicator, [7-1](#)  
features, [1-1](#)  
field wiring arm, [2-1](#), [2-12](#)  
filtering, description, [4-4](#)  
format, data, [4-3](#)

## I

indicator lights, [2-12](#)  
indicators, [7-1](#)  
    diagnostic, [2-12](#), [7-1](#)  
    fault, [7-1](#)  
    RUN, [7-1](#)  
input range selection, [5-2](#)  
input ranges, program selectable, [1-2](#)  
input voltage/current ranges, [4-2](#)

## K

keying, [2-2](#)  
    band positions, [2-3](#)

## M

module configuraion, [2-1](#)

module location, in I/O chassis, [2-2](#)

## N

noise interference, [2-2](#), [2-3](#)

## P

potentiometers, [6-2](#)

power requirements, from backplane, [2-2](#)

programming, with multiple GETs, [D-1](#)

## R

range selection

bit settings, [4-2](#)

input, [5-2](#)

real time sampling, [4-5](#)

bit settings, [4-5](#)

related products, [P-2](#)

related publications, [P-3](#)

## S

sample program, [3-5](#)

PLC-2, [3-2](#)

PLC-3, [3-3](#)

PLC-5, [3-4](#)

scaling

description, [4-6](#)

implementation, [4-6](#)

minimum block transfer requirements,  
[4-7](#)

ranges, [4-6](#)

scan time, module, [3-5](#)

selection plugs, locations, [2-9](#)

specifications, [A-1](#)

## T

terminology used, for module, [P-1](#)

test points, [6-2](#)

troubleshooting, [7-1](#)

chart, [7-3](#)

## U

update time, [1-1](#)

## V

voltage-mode input devices, recommended

cable length, [2-3](#)



**ALLEN-BRADLEY**  
A ROCKWELL INTERNATIONAL COMPANY

Allen-Bradley has been helping its customers improve productivity and quality for 90 years. A-B designs, manufactures and supports a broad range of control and automation products worldwide. They include logic processors, power and motion control devices, man-machine interfaces and sensors. Allen-Bradley is a subsidiary of Rockwell International, one of the world's leading technology companies.



With major offices worldwide.

Algeria • Argentina • Australia • Austria • Bahrain • Belgium • Brazil • Bulgaria • Canada • Chile • China, PRC • Colombia • Costa Rica • Croatia • Cyprus • Czech Republic • Denmark • Ecuador • Egypt • El Salvador • Finland • France • Germany • Greece • Guatemala • Honduras • Hong Kong • Hungary • Iceland • India • Indonesia • Israel • Italy • Jamaica • Japan • Jordan • Korea • Kuwait • Lebanon • Malaysia • Mexico • New Zealand • Norway • Oman • Pakistan • Peru • Philippines • Poland • Portugal • Puerto Rico • Qatar • Romania • Russia-CIS • Saudi Arabia • Singapore • Slovakia • Slovenia • South Africa, Republic • Spain • Switzerland • Taiwan • Thailand • The Netherlands • Turkey • United Arab Emirates • United Kingdom • United States • Uruguay • Venezuela • Yugoslavia

World Headquarters, Allen-Bradley, 1201 South Second Street, Milwaukee, WI 53204 USA, Tel: (1) 414 382-2000 Fax: (1) 414 382-4444