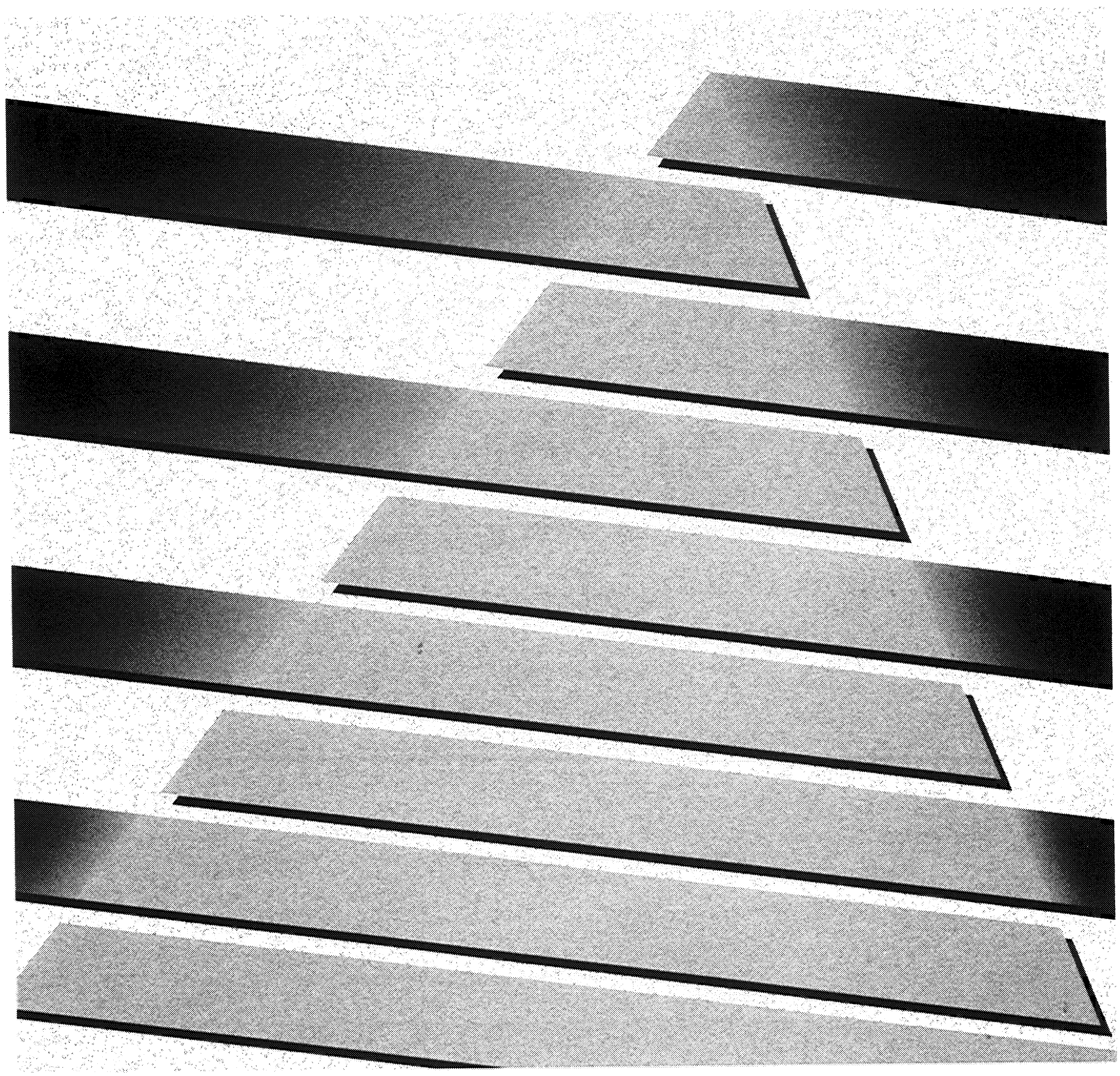
ALLEN-BRADLEY

Fast Millivolt Input Module
Cat. No. 1771-IFM Series B

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.

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.

Using This Manual

Purpose of Manual

This manual shows you how to use your Fast Millivolt 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 (PLC) to make efficient use of your input module. In particular, you must know how to program block transfer instructions.

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

Vocabulary

In this manual, we refer to:

- The individual fast millivolt 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 Fast Millivolt Input Module	Description of the module, including general and hardware features
2	Installing the Fast Millivolt Input Module	Module power requirements, keying, chassis location Wiring of field wiring arm
3	Module Programming	Reading data from your module Sample programs
4	Module Configuration	Hardware and software configuration Data format
5	Module Status and Input Data	Reading data from your module Module read block format
6	Calibration	How to calibrate your module
7	Troubleshooting	Diagnostics reported by the module

Chapter	Title	Topics Covered
Appendix A	Specifications	Your module specifications
Appendix B	Programming Examples	
Appendix C	Data Formats	Information on BCD, unsigned magnitude (12-bit) binary
Appendix D	Block Transfer with Mini-PLC-2 and Mini-PLC-2/20	

Warnings and Cautions

This manual contains warnings and cautions.

ATTENTION: A warning indicates where you may be injured if you use your equipment improperly.

ATTENTION: Cautions indicate where equipment may be damaged from misuse.

You should read and understand cautions and warnings before performing the procedures they precede.

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-IFM 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 and write block data. I/O image table use is an important factor in module placement and addressing selection. The module's data table use is listed in the following table.

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-IFM/B	8	8	12	21	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.
No = Restricted to complementary module placement

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
- 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 unless you are using 1 or 1/2 slot addressing. Avoid placing this module 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.

Table of Contents

Overview of the Fast Millivolt Input Module

Chapter 1

Chapter Objectives 1-1
Module Description 1-1
How Analog Modules Communicate with Programmable Controllers 1-2
Accuracy 1-3
Getting Started 1-3

Installing the Fast Millivolt Input Module

Chapter 2

Chapter Objectives 2-1
Before You Install Your Input Module 2-1
Electrostatic Damage 2-1
Power Requirements 2-2
Module Location in the I/O Chassis 2-2
Connecting Wiring 2-4
Indicator Lights 2-6
Chapter Summary 2-6

Module Programming

Chapter 3

Extended Input Range 3-4
Module Scan Time 3-5
Chapter Summary 3-5

Module Configuration

Chapter 4

Chapter Objectives 4-1
Configuring Your Input Module 4-1
Input Type 4-1
Digital Filtering 4-1
Real Time Sampling 4-3
Scaling 4-4
Fast Millivolt Input Module Default Settings 4-5
Chapter Summary 4-6

Module Status and Input Data

Chapter 5

Chapter Objectives 5-1
Reading Data From Your Module 5-1
Block Transfer with the Fast Millivolt Input Module 5-1
Chapter Summary 5-2

Module Calibration

Chapter 6
Chapter Objective 6-1
Tools and Equipment 6-1
How to Calibrate the Fast Millivolt Input Module 6-1

Troubleshooting

Chapter 7
Chapter Objectives 7-1
Diagnostics Reported by the Module 7-1

Specifications

Appendix A
Specifications A-1

Programming Examples

Appendix B
Sample Programs for the Analog Input Module B-1
PLC-2 Family Processors B-1
PLC-3 Family Processors B-3
PLC-5 Family Processors B-4

Data Formats

Appendix C
4-Digit Binary Coded Decimal (BCD) C-1
(12-bit) Binary C-2

Block Transfer with Mini-PLC-2 and Mini-PLC-2/20

Appendix D
Multiple GET Instructions - Mini-PLC-2 and PLC-2/20 Processors D-1
Setting the Block Length (Multiple GET Instructions only) D-4

Chapter 1

Overview of the Fast Millivolt Input Module

Chapter Objectives

This chapter gives you information on:

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

Module Description

The 1771-IFM fast millivolt 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. 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 1771-IFM module senses 8 differential analog inputs and converts them to a proportional four-digit BCD or fifteen-bit binary value. The module reads a 0 to 50mV input signal. It can read slightly overrange or slightly underrange voltages.

This module's features include:

- 8 differential inputs on one card
- Selectable real-time sampling
- Selectable scaling to engineering units
- Selectable digital filtering
- 0 to 50mV input range with extended linear range above 50mV and below 0mV

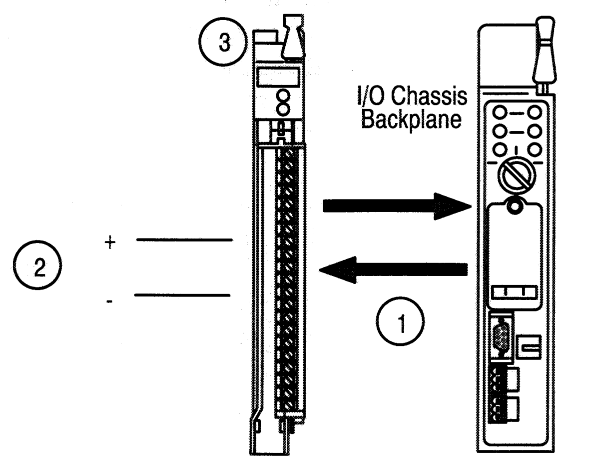
How Analog Modules Communicate with Programmable Controllers

The processor transfers data to and from the module using BTW (block transfer write) and BTR (block transfer read) 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 and calibration values to the module using 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 theses 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.

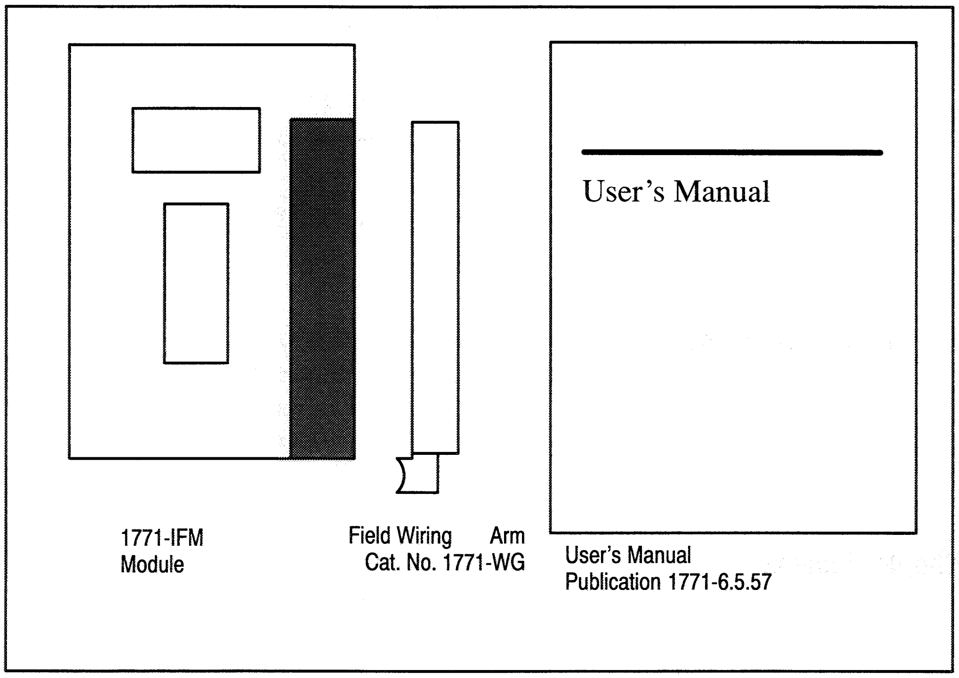
7. Your ladder program should allow write block transfers to the module only when enabled by the operator at power-up.

Accuracy

The accuracy of the fast millivolt input module is described in Appendix A.

Getting Started

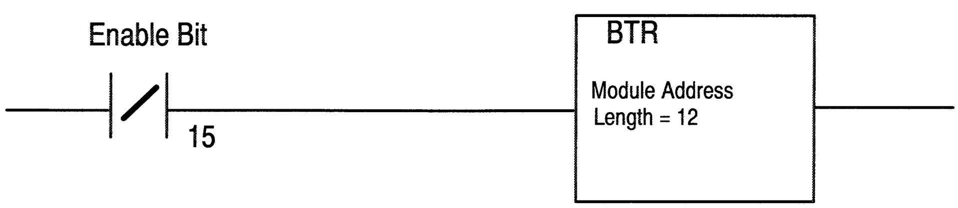
Your input module package contains the following items. Please check that each part is included and correct before proceeding.



Checking Operation of Your Input Module

You can check operation of your input module by programming a simple rung and checking data values. Proceed as follows.

1. Program the following rung into your ladder logic or user program.



2. Place the PLC in run mode and observe data values in words 5 through 12. Your module is operating correctly if it is returning data in these words. **NOTE:** The module is operating in default mode, which is binary data type and 0-4095 scaling.

ATTENTION: Do not use this data for control purposes until you have read and understood the first six chapters of this manual. You must understand how the data acts under all circumstances to prevent unintended machine operation which could result in injury to personnel or damage to your process.

---

3. Proceed with Chapters 2 through 7 of this manual.

Extended Input Range

In any application where the inputs use the extended linear range, the program must monitor the overrange and underrange bits; especially when 0 to 9999 (BCD) scaling is used. The output data can rollover past 9999 (at V_{in} = 50mV) to 0000 (at V_{in} > 50mV).

Chapter Summary

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

Chapter 2

Installing the Fast Millivolt 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
- installing the input module

Before You Install Your Input Module

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

Action required:	Refer to:
Calculate the power requirements of all modules in each chassis.	Power Requirements
Determine where to place the module in the I/O chassis.	Module Location in the I/O Chassis
Key the backplane connector in the I/O chassis.	Installing the analog module
Make connections to the wiring arm.	Connecting Wiring and Grounding

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 warning:

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

- Wear an approved wrist strap grounding device when handling the module.
- 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 chassis backplane from the chassis power supply. The module requires 750mA @ 5V DC.

Add this value 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 slot of the I/O chassis except for the extreme left slot. This slot is reserved for 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.

Installing the Analog Module

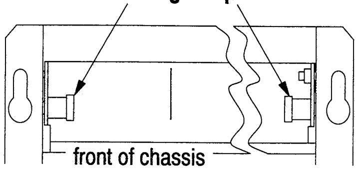
To install your module in an I/O chassis:

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

ATTENTION: Remove power from the 1771 I/O chassis backplane and disconnect the cable from the module 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 could cause module damage, degradation of performance, or injury.



12453-I

2. Lift the locking latch holding the module into the chassis. (On chassis equipped with a chassis locking bar, pull the locking-bar pins to release the locking bar and swing it up.)

3. Position the keying bands (Figure 2.1) in the backplane connectors to correspond to the key slots on the module. This prevents you from inserting the wrong module in this slot. This analog module uses:
   - between 20 and 22
   - between 28 and 30

ATTENTION: Observe the following precautions when inserting or removing keys:

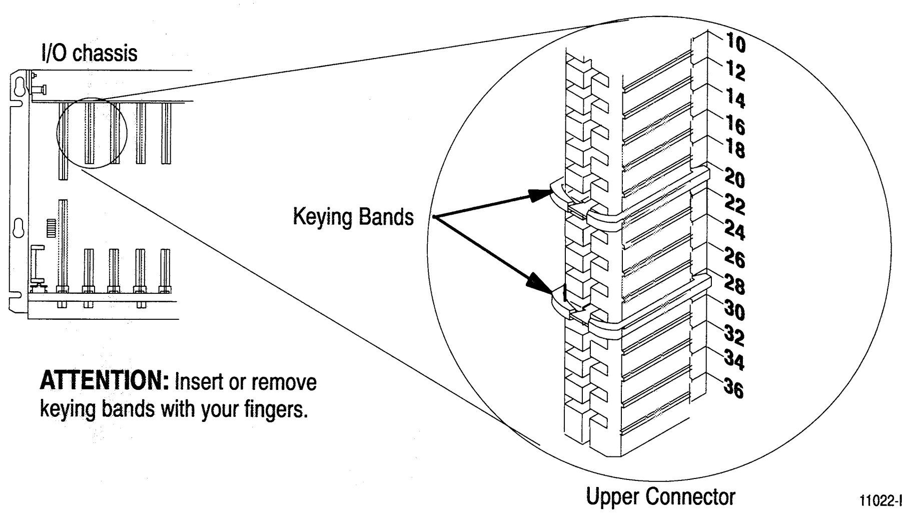
- insert or remove keys with your fingers
- make sure that key placement is correct

Incorrect keying or the use of a tool can result in damage to the backplane connector and possible system faults.

You can change the position of these bands if subsequent system design and rewiring makes insertion of a different type of module necessary. Use needlenose pliers to insert or remove keying bands.

Keying Band Location
Between 20 and 22
Between 28 and 30

Figure 2.1
Keying Positions



ATTENTION: Insert or remove keying bands with your fingers.

4. Place the module in the plastic tracks on the top and bottom of the slot that guides the module into position.

5. Do not force the module into its backplane connector. Apply firm even pressure on the module until it is firmly seated in the chassis. **Note:** The chassis locking bar will not close if all modules are not seated properly.

6. Snap the chassis locking bar (or locking latch on earlier chassis) over the top of the module to secure it. Make sure the locking pins on the locking bar are fully engaged.

7. Connect the wiring to the module as explained under "Connecting Wiring" below.

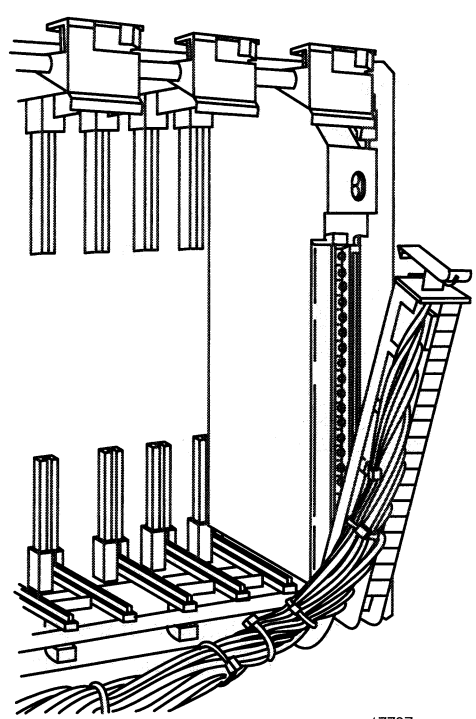
Connecting Wiring

Connect your I/O devices to the field wiring arm shipped with the module. Attach the field wiring arm to the pivot bar at the bottom of the I/O chassis (Figure 2.2). The field wiring arm pivots upward and connects with the module so you can install or remove the module without disconnecting the wires.

The wiring arm is specific to the input module. The 1771-IFM module uses the catalog number 1771-WG wiring arm.

Input connections for the 1771-IFM are shown in Figure 2.3.

Figure 2.2
Field Wiring Arm Installed



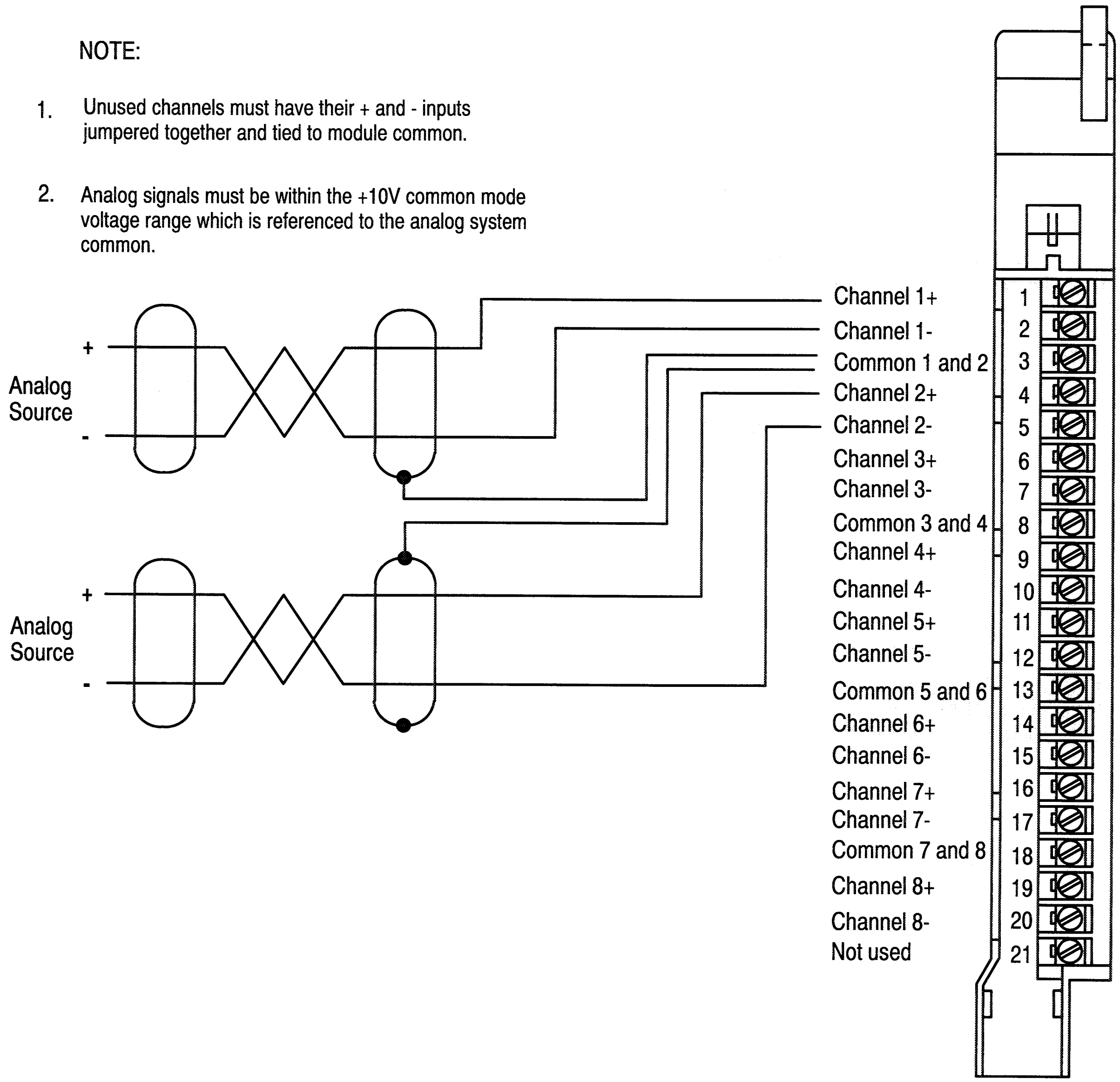
Recommended maximum cable length for voltage-mode input devices is 50 feet. This recommendation is based on considerations of signal degradation and electrical noise immunity in typical industrial environments.

Figure 2.3
Connection Diagram for the Fast Millivolt Input Module

NOTE:

1. Unused channels must have their + and - inputs jumpered together and tied to module common.

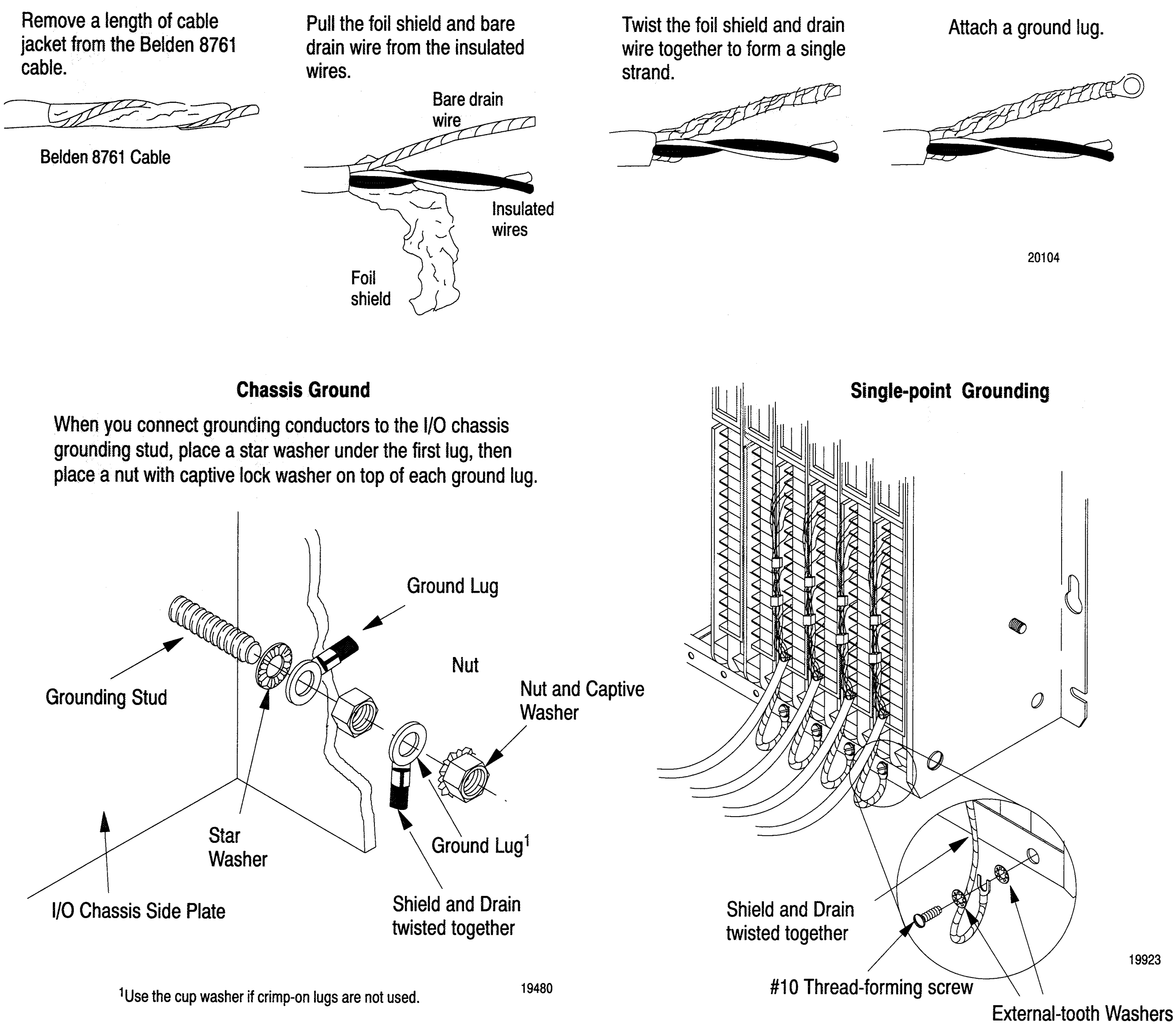
2. Analog signals must be within the +10V common mode voltage range which is referenced to the analog system common.



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 2.4). At the opposite end of the cable, tape exposed shield and drain wire with electrical tape to insulate it from electrical contact.

Figure 2.4
Cable Grounding

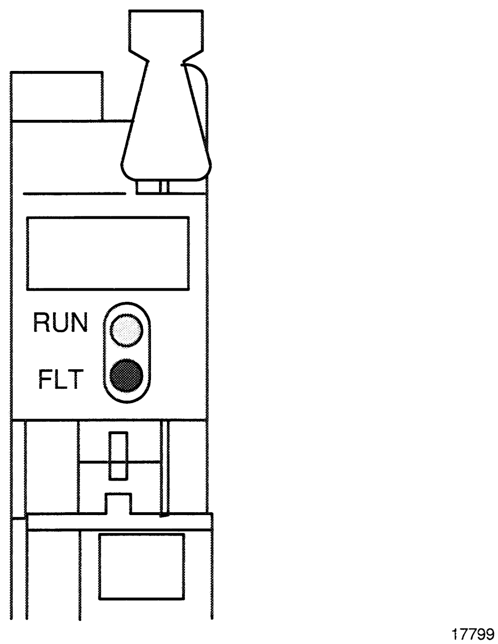


Refer to Wiring and Grounding Guidelines, publication 1770-4.1 for additional information.

Indicator Lights

The front panel of the input module contains a green RUN and a red FLT (fault) indicator (Figure 2.5). At power-up, the red indicator turns ON and an initial module self-check occurs. If there is no fault, the red indicator turns off. Then the green indicator will turn ON, and stay on. 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 8, Troubleshooting.

Figure 2.5
Diagnostic Indicators



Chapter Summary

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

The next step is to configure your module through the user program. Chapters 4 and 5 will show you how to program and configure your module for the type of operation you require.

Chapter 3

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 the 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.

Your fast millivolt input module will power-up with a default configuration. See the configuration default section to understand what this configuration looks like. Also, refer to Appendix B for example configuration blocks and instruction addresses to get started.

The following example programs illustrate the minimum programming required for communication to take place.

PLC-2 Program Example

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 D.

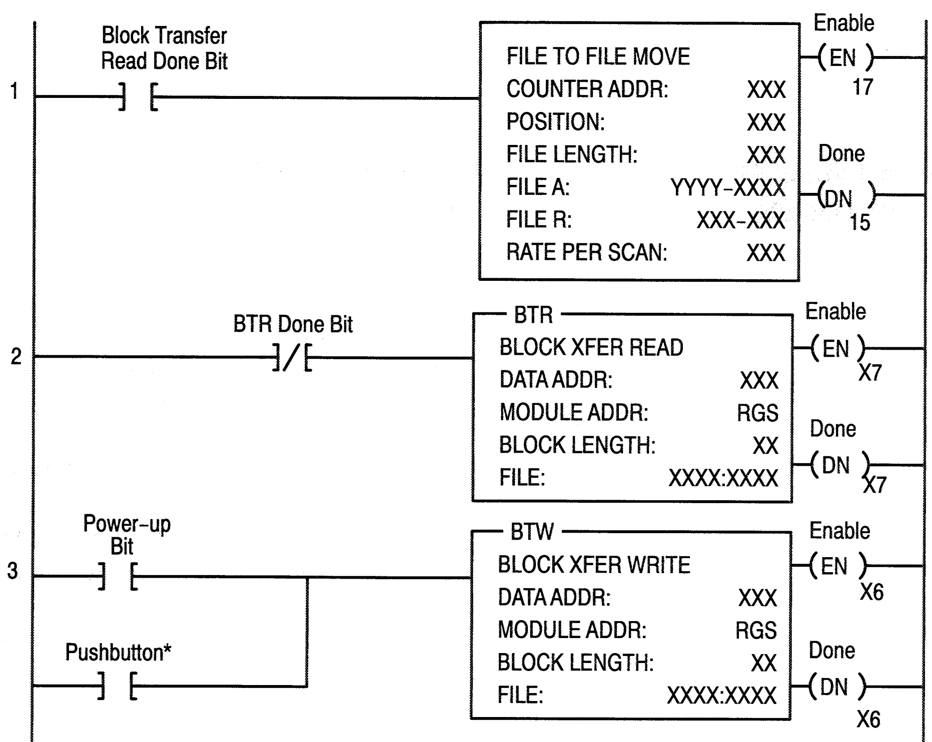
Figure 3.1
PLC-2 Family Sample Program Structure

Rung Descriptions

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.

1. 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.
2. 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 are the conditioning block transfer rungs. Include all the input conditioning shown in the example program.



* 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 Program Example

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 the module (when programming a block transfer write) or from the 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.

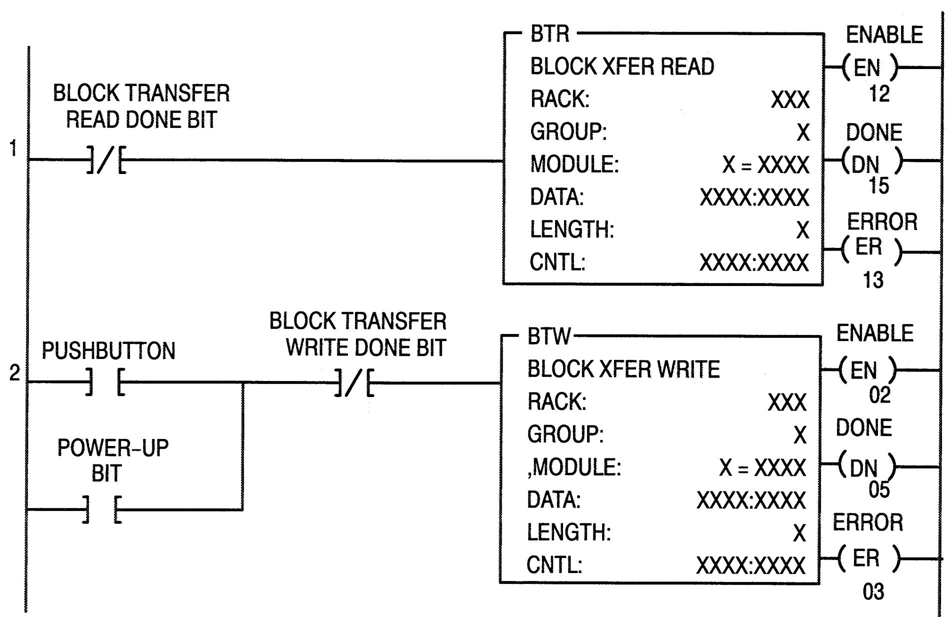
A sample program segment with block transfer instructions is shown in Figure 3.2, and described below.

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 write block transfer to configure the module, and then does consecutive read block transfers 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 enable 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 Program Example

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

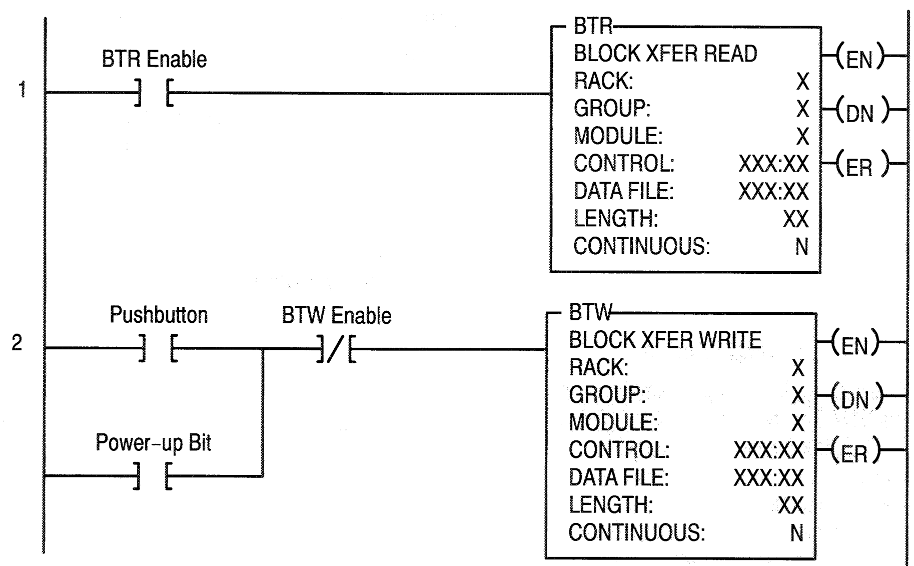
- You must use enable bits instead of done bits as the conditions on each rung.
- A separate control file must be selected for each of the BT instructions. Refer to Appendix B.

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 unless the first pass bit is added to the BTW input conditions.



Extended Input Range

In any application where the inputs use the extended linear range, the program must monitor the overrange and underrange bits; especially when 0 to 9999 (BCD) scaling is used. The output data can rollover past 9999 (at V_{in} = 50mV) to 0000 (at V_{in} > 50mV).

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 figure 4.4.

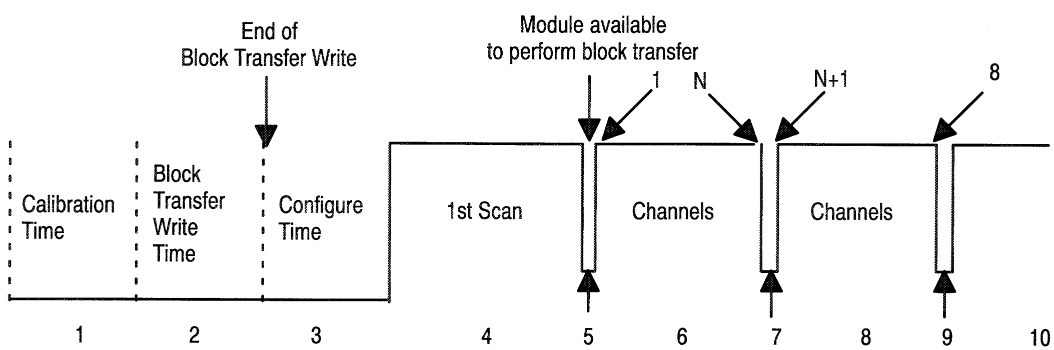
The following description references the sequence numbers in figure 4.4.

Following an initial self-calibration "1", a block transfer write "2", and a configuration set-up "3", the module will scan the inputs "4", and fill 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 BTR empties the buffer. Input channel scanning "8" is not interrupted by the BTR.

Following the BTR the input module inhibits block transfer communications with the programmable controller until it has scanned its inputs "9". The input module repeats the scan sequence "10", updating the input values until another block transfer request is received. **Therefore, BTRs can only be completed as frequently as the total scan time of the input module.**

Figure 3.4
Block Transfer Time



1771-IFM Scan time = 12.5ms for 8 differential inputs.
Add 2.5ms for digital filtering. Add 0.25ms for BCD data.
Scaling adds no time.

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.

Chapter 4

Module Configuration

Chapter Objectives

In this chapter you will read how to configure your module's hardware, 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.

The software configurable features available with the Fast Millivolt Input Module (1771-IFM) are:

- data format
- digital filtering
- real time sampling
- scaling to engineering units

Data format, digital filtering and real time sampling can be configured with a block transfer write of only 3 or 5 words in length.

Input Type

You must also indicate what format will be used to read data from your module (Table 4.A). Typically, BCD is selected with PLC-2 processors, and binary (also referred to as integer or decimal) is selected with PLC-3 and PLC-5 processors. See Figure 4.3 and Appendix C for details on Data Format.

Table 4.A
Selecting Format for Reading Data

Decimal Bit 10 (Octal Bit 12)	Data Format
0	Binary
1	BCD

Digital Filtering

The analog input 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.

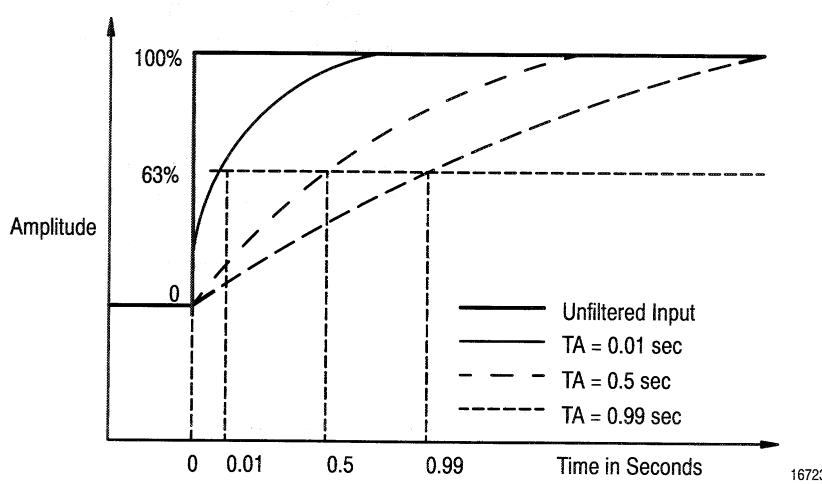
The digital filter equation is a classic first order lag equation (Figure 4.1). Using a step input change to illustrate the filter response (Figure 4.2), you can see that when the digital filter constant time elapses, 63.2% of the total response is reached. Each additional time constant achieves 63.2% 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
Δt = 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 .00 BCD to .99 BCD (.00 BCD = no filter; .99 BCD = maximum filter) are set in bits 00 through 07 of word 3 of the block transfer write data table. If an invalid digital filter value is entered (e.g., 1F), the invalid scaling/filtering data bit in the BTR status area will be set. If an invalid digital filter value is entered, the module will not perform digital filtering. If you choose to 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 from a fixed time period for use by the processor. It is available on all analog input modules.

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 (ΔT) 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 (msec) to 3.1 seconds in increments of 100msec. Refer to Table 4.B below for actual bit settings and figure 5.3 for bit locations. Note that the default mode of operation is implemented by placing all zeroes in bits 13 through 17. Note that binary representation of the RTS filter bit string is the RTS period X 100msec. For example, 900msec = 01001 = (9 X 100msec).

Table 4.B
Bit Settings for the Real Time Sample Mode

Decimal Bits	15	14	13	12	11	Sample Time Period
Octal Bits	17	16	15	14	13	
	0	0	0	0	0	No RTS - Default setting = 12.5ms updates
	0	0	0	0	1	100 ms
	0	0	0	1	0	200 ms
	0	0	0	1	1	300 ms
	0	0	1	0	0	400 ms
	0	0	1	0	1	500 ms
	0	0	1	1	0	600 ms
	0	0	1	1	1	700 ms
	0	1	0	0	0	800 ms
	0	1	0	0	1	900 ms
	0	1	0	1	0	1.0 sec
	0	1	1	1	1	1.5 sec
	1	0	1	0	0	2.0 sec
	1	1	0	0	1	2.5 sec
	1	1	1	1	0	3.0 sec
	1	1	1	1	1	3.1 sec

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 0 to 50mV range.

The format of scaled data is 4-digit BCD or 15-bit binary. The resolution at the module of scaled values is the same as for unscaled data: one part in 4095. 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.

Implementing the Scaling Feature

You implement the scaling feature by inserting minimum and maximum scaled values in the appropriate configuration words of the block transfer write data table. See Table 4.C and Figure 4.3.

Scaling Ranges

The maximum range of the scaling values is 9999 BCD or 32767 binary.

Typically, invalid scaling values are minimum values greater than maximum values, or values greater than 9999 in BCD or 32767 in binary. If invalid values are entered into the scaling words, the corresponding input in the BTR data will be zero and the invalid scaling/filtering data bit will be set. If the minimum and maximum scaling values are equal, the invalid scaling/filtering data bit will NOT be set, and the corresponding BTR data for that channel will be equal to the scaling values.

Table 4.C
Block Transfer Read and Write File Lengths for a Varying Number of Input Channels When Scaling is Chosen

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

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

Fast Millivolt Input Module Default Settings

If no block transfer write block, or a write block length of 3 or 5 words with all zeroes, is sent to the Analog Input Module (1771-IFM), the module will default to the following configuration:

- Binary data format
- no real time sampling (RTS)
- no filtering
- 0 to 4095 scaling

A block transfer write of length 0 will default to a length of 21, so scaling values must be entered into the BTW data table.

Figure 4.3
BTW Word Format for the Fast Millivolt Input Module (1771-IFM)

Decimal Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Octal Bit	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00
Word 1	Not used															
2	Not used															
3	Real time sampling					Data Format	Not Used		Digital filter							
4	Not used															
5	Not used															
6	Channel one, minimum scaling value															
7	Channel one, maximum scaling value															
8	Channel two, minimum scaling value															
9	Channel two, maximum scaling value															
10	Channel three, minimum scaling value															
:	:															
21	Channel eight, maximum scaling value															

Bit/Word Descriptions for the Configuration Block

Word	Decimal Bits (Octal Bits)	Definition
1 and 2		Not used
3	Bits 0 - 7	Digital filter reduces effect of noise on input. (00 to 99 BCD)
	Bit 12 (Bit 10)	Data format matches format of processor. Set for BCD (1), reset for Binary (0). See Table 4.A.
	Bits 13 - 17 (Bits 11 - 15)	Real time sampling (0.1 to 3.1 sec) See Table 4.B for time intervals.
4		Not used

Word	Decimal Bits (Octal Bits)	Definition
5		Not used
6 thru 21		Minimum and maximum scaling values for each channel. Enter in same format as data format.

Chapter Summary

In this chapter you learned how to configure your module's hardware, 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
- input module 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 the Fast Millivolt Input Module (1771-IFM) Block Transfer Read

Decimal Bit	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00	Remarks
Octal Bit	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00	
Word 1	Not used												Diagnostics				
2									8	7	6	5	4	3	2	1	Data underrange for channels 1 thru 8
3									8	7	6	5	4	3	2	1	Data overrange for channels 1 thru 8
4	Not used																
5	Channel 1 input																
6	Channel 2 input																
7	Channel 3 input																
:	:																
12	Channel 8 input																

Block Transfer with the Fast Millivolt Input Module

Your module communicates with your processor through bidirectional block transfers. A bidirectional block transfer is the sequential performance of both read and write operations.

BTR Format

The fast millivolt input module (1771-IFM) reports the status of eight channels to the processor.

Figure 5.2
BTR Format for the Fast Millivolt Input Module (1771-IFM)

Word	Decimal Bits (Octal Bits)	Description
Word 1	Bit 00	**Power up bit** is used by the module to tell the processor that it's alive but not yet configured. It is a key element in the application program.
	Bit 01	**Out of range bit** is sent to tell the processor that one or more channels are either over or under range.
	Bit 02	**Invalid scaling/filtering data bit** reports that the scaling or filtering data is somehow invalid.
	Bit 03	**Real Time Sampling** time-out bit
Word 2	Bits 00-07	Individual **underrange bits** for each channel.
Word 3	Bits 00-07	Individual **overrange bits** for each channel.
Word 4		**Not used**
Words 5 thru 12		**Input values.**

Chapter Summary

In this chapter you learned the meaning of the status information that the input modules send to the processor.

Chapter 6

Module Calibration

Chapter Objective

In this chapter we tell you how to calibrate your module.

Tools and Equipment

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

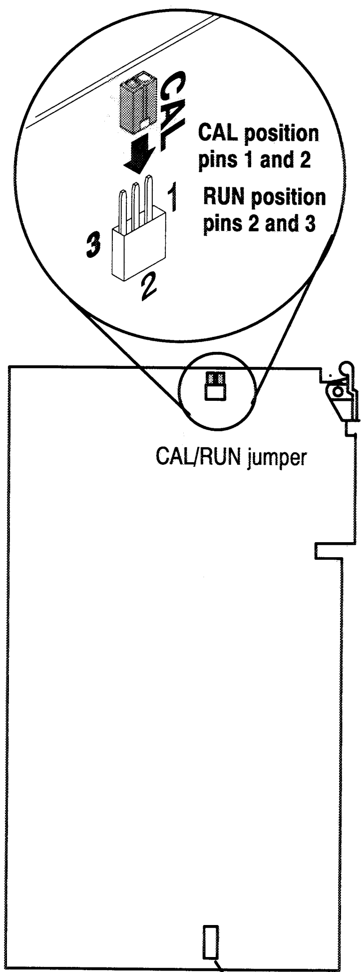
Tool or Equipment	Description	Model/Type	Available from:
Precision Voltage Source	0-5V DC, 0.1mV resolution minimum	Analog 3100, Data Precision 8200 or equivalent	
Digital Voltmeter	5-1/2 digit, 0.01% accuracy minimum	Keithley 191, Fluke 8300A or equivalent	
Alignment Tool	For potentiometer adjustment	PN 35F616	Newark Electronics 5000 N. Pulaski Road Chicago, IL
Potentiometer Sealant	For sealing pots after adjustment	Torque Seal	Organic Products P.O. Box 928 Irving, TX
Industrial Terminal and Interconnect Cable	Programming terminal for A-B family processors	Cat. No. 1770-T3 or Cat. No. 1784-T45, -T50, etc.	Allen-Bradley Company Highland Heights, OH
Backplane Extender Card	To extend output module out from chassis for adjustments	Cat. No. 1771-EZ	Allen-Bradley Company Highland Heights, OH

How to Calibrate the Fast Millivolt Input Module

The fast millivolt input module is **shipped already calibrated.** If necessary to recalibrate the module, you must calibrate it in an I/O chassis. The module must communicate with the processor and industrial terminal. Calibration consists of adjusting the internal offset voltage.

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 Internal Offset Voltage



1. Turn off the 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. Remove the module covers.
5. Move the CAL/RUN jumper to the CAL position (pins 1 and 2).
6. Plug the module into the extender card (cat. no. 1771-EZ).
7. Insert the extender card (with module attached) into the chassis, and firmly seat it into the connector.
8. Reposition the field wiring arm on the module. Press firmly to seat.
9. Set the precision voltage source to 49.970mV.
10. Turn on power to the processor and I/O chassis.

Important: The calibration voltage is selected as full scale, 50mV, minus 5/2 LSBs, 30 microvolts, in order to ensure that all channels read within ±1/2 LSB of each other. Verify the voltage at the input of the module.

11. With the module in the default configuration, observe the BTR file for all eight channels. Adjust the potentiometer on the bottom of the module until all eight channels flicker between FFDh and FFEh.

Important: All channel readings will not be the same. Some channels may stay on FFD while others may stay on FFE. Adjust the potentiometer until the majority of the channels flicker between FFD and FFE. When adjustment is complete, no channel should display any value other than FFD or FFE.

12. Turn off the power to the processor and I/O chassis.
13. Remove the module and extender card from the chassis and disconnect the module from the extender card.
14. Reposition the CAL/RUN jumper to the RUN position (pins 2 and 3).
15. Replace the covers and reinstall the IFM module in the chassis.

Chapter 7

Troubleshooting

Chapter Objectives

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 turns on the RED indicator as a lamp test, then checks for:

- correct RAM operation
- firmware errors

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

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 and 3** 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/filtering data bit (bit 02) will be set if invalid scaling data is entered into any of the minimum/maximum scaling value words. 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) time-out 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.

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

Table 7.A
Troubleshooting Chart for the Fast Millivolt Input Module

Indicator Status (color)	Description of Fault or System Status	Action to Take
Module run ON (green) Module fault OFF (red)	Normal Indication	None
	Incorrect data in final storage word locations in processor's data table, possible severed or disconnected input cable associated with the affected channels.	Repair or replace cable.
	Input module is conditioned for BCD instead of binary, or vice versa, or invalid scaling values chosen.	Condition module for desired format (BCD or binary), enter correct data and initiate another block transfer write.
	Block transfers are not being performed: processor is in program mode when module powers up.	Place processor in run mode. Recycle power. Replace faulty module if necessary.
	Ladder diagram program error. Read and write block transfer both enabled in PLC-2 processor. Wrong module address in block transfer instruction, or wrong BTR or BTW lengths chosen.	Debug program.
	Equipment failure (PC processor, adapter module, etc.).	Isolate system component that has failed and replace it.
Module run OFF (green) Module fault ON (red)	Hardware failure in module.	Return module for repair.
Neither indicator ON	No power.	Turn power OFF. Remove and reinsert module into chassis. Turn power ON.
	Fuse is bad.	Return module for repair.

Specifications

Inputs per module	8 differential low level
Module Location	1771 I/O chassis - 1 slot
Input voltage range	0 to 50mV
Resolution	12-bit binary
Accuracy	0.1% of range @ 25^{o}C
Linearity	$\pm$ 1 LSB
Repeatability	$\pm$ 1 LSB
Isolation Voltage	1500V (transient)
Input overvoltage protection	32V
Input impedance	1M ohms
Maximum input voltage	$\pm$ 10V
Common mode rejection	100db dc - 60Hz
Backplane current	0.75A @ +5V
BCD and unscaled binary output to processor	0000 to 4095_{10}
Engineering units sent to processor	9999 BCD with selectable scaling 32767 Binary
A/D converter	monotonic output with no missing codes resolution: 12-bit binary absolute accuracy: $\pm$0.1% of full scale quantizing error: $\pm$1/2 LSB temperature coefficient: $\pm$50ppm/^{o}C of full scale range for 0 to 60^{o}C ambient recalibration time: calibration should be checked at 6 month intervals to maintain specified accuracy Internal scan rate - 12.5 milliseconds for 8 differential inputs (no digital filtering) - add 2.5ms for filtering
Environmental conditions operational temperature: storage temperature: relative humidity:	 0 to 60^{0}C (32 to 140^{0}F) -40 to 85^{0}C (-40 to 185^{0}F) 5 to 95% (without condensation)
Conductors Wiring Category	14 gauge stranded (max.) 3/64 inch insulation (max.) Category 2[1]
Keying	between 20 and 22 between 28 and 30
Wiring Arm	Catalog Number 1771-WG
Field Wiring Arm Screw Torque	7-9 inch-pounds

[1] Refer to publication 1770-4.1, "Programmable Controller Wiring and Grounding Guidelines."

Programming Examples

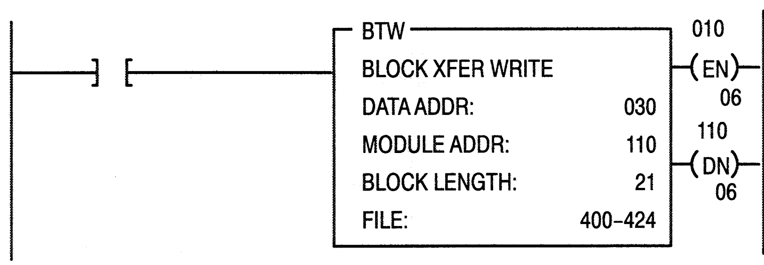
Sample Programs for the Analog Input Module

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. The examples shown are for the 1771-IFM.

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.

In RUN/PROG Mode

Action	Result
1. Press [SEARCH]8<data address>	Finds the block address transfer instruction
2. Press CANCEL COMMAND	Removes preceeding command
3. Press [DISPLAY]0 or 1	Displays the file in binary or BCD
4. Press [SEARCH]51 Cursor defaults to first entry in file when SEARCH 51 is pressed.	On line data change
5. Press [INSERT]	Writes data to file element

In PROG Mode

Action	Result
1. Press [SEARCH]8<data address>	Finds the block transfer instruction
2. Press CANCEL COMMAND	Removes preceeding command
3. Press [DISPLAY]0 or 1	Displays the file in binary or BCD
4. Press [DISPLAY]001 and enter data	Puts cursor on word 1
5. Press [INSERT]	

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 21 words if using 8 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

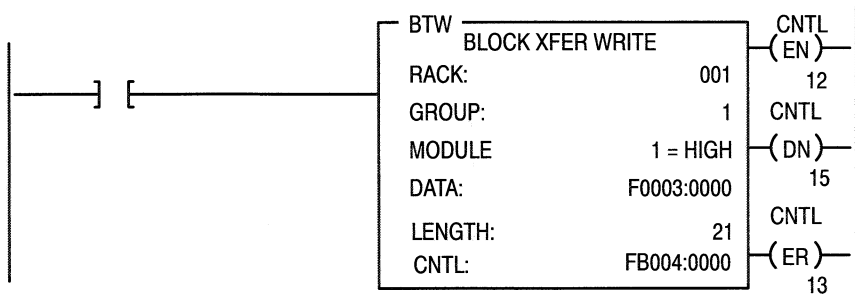
```
DATA ADDR: 030      BINARY DATA MONITOR          BLOCK LENGTH: 21
                    BLOCK XFER WRITE
                    MODULE ADDR: 110
                    FILE: 400-424
   POSITION                  FILE DATA

      001        00000000  00000000       00000000  00000000
      002        00000000  00000000       00000000  00000000
      003        00000000  00000000       00000000  00000000
      004        00000000  00000000       00000000  00000000
      005        00000000  00000000       00000000  00000000
      006        00000000  00000000       00000000  00000000
      007        00000000  00000000       00000000  00000000
      008        00000000  00000000       00000000  00000000
```

PLC-3 Family Processors

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, you can go on to step 3.

3. Enter the data corresponding to your bit selection in words 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].

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

START - W0003 : 0000

WORD #	0	1	2	3
00000	00000000 00000000	00000000 00000000	00000000 00000000	00000000 00000000
00004	00000000 00000000	00000000 00000000	00000000 00000000	00000000 00000000
00010	00000000 00000000	00000000 00000000	00000000 00000000	00000000 00000000
00014	00000000 00000000	00000000 00000000	00000000 00000000	00000000 00000000
00020	00000000 00000000	00000000 00000000	00000000 00000000	00000000 00000000

DATA MONITOR $ W03:0 - [ ]

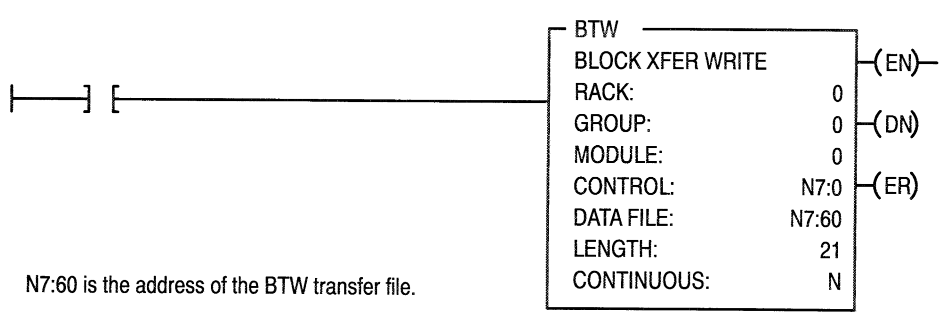
PROG : I/O OFF NO FORCES : NO EDITS : RUNG # [RM000000] : MEM PROT OFF

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.

1. Enter the following rung:



2. Press [F8],[F5] and enter N7:60 to display the configuration block.

The industrial terminal screen should like Figure B.3.

Figure B.3
Sample PLC-5 Data File

Address	0	1	2	3	4	5	6	7	8	9
N10:0	0	0	1024	0	0	0	4095	4096	8191	8192
N10:10	12287	12288	16383	16384	20479	20480	24575	24576	28671	28672
N10:20	32767	0	0	0	0	0	0	0	0	0
N10:30	2	100	0	0	0	4096	8193	12288	16384	20481
N10:40	24577	28672	0	0	0	0	0	0	0	0
N10:50	0	0	0	0	0	0	2	100	0	0
N10:60	0	4096	8193	12288	16384	20481	24577	28672		

3. Enter the data corresponding to your bit selections and add scaling values, if scaling is desired.

4. [ESC] returns you to the main menu.

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

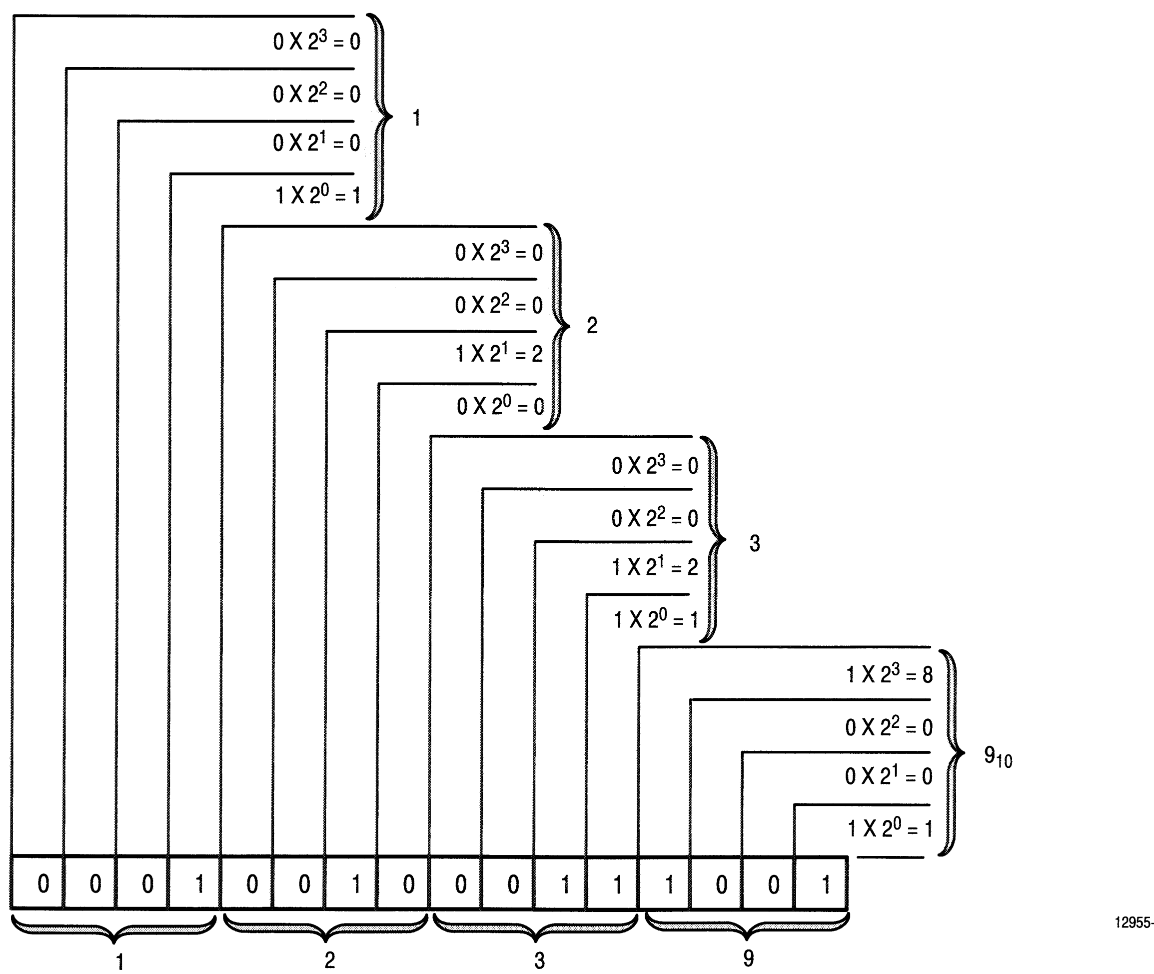


Table C.A
BCD Representation

Place Value 2^3 (8)	2^2 (4)	2^1 (2)	2^0 (1)	Decimal Equivalent
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

(12-bit) Binary

Binary should be used with the PLC-2 family of processors when performing computations in the processor.

The 12-bit binary format uses an arrangement of 12 binary digits to represent a decimal number ranging from 0 to 4095. The decimal equivalent of a 12-bit binary number is determined by multiplying the binary digit (0 or 1) by its corresponding place value and adding these numbers together.

Example: The following binary number is equal to decimal 3167.

$$\begin{array}{cccccccccccc} 2^{11} & 2^{10} & 2^9 & 2^8 & 2^7 & 2^6 & 2^5 & 2^4 & 2^3 & 2^2 & 2^1 & 2^0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1_2 \end{array} = 3167_{10}$$

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.

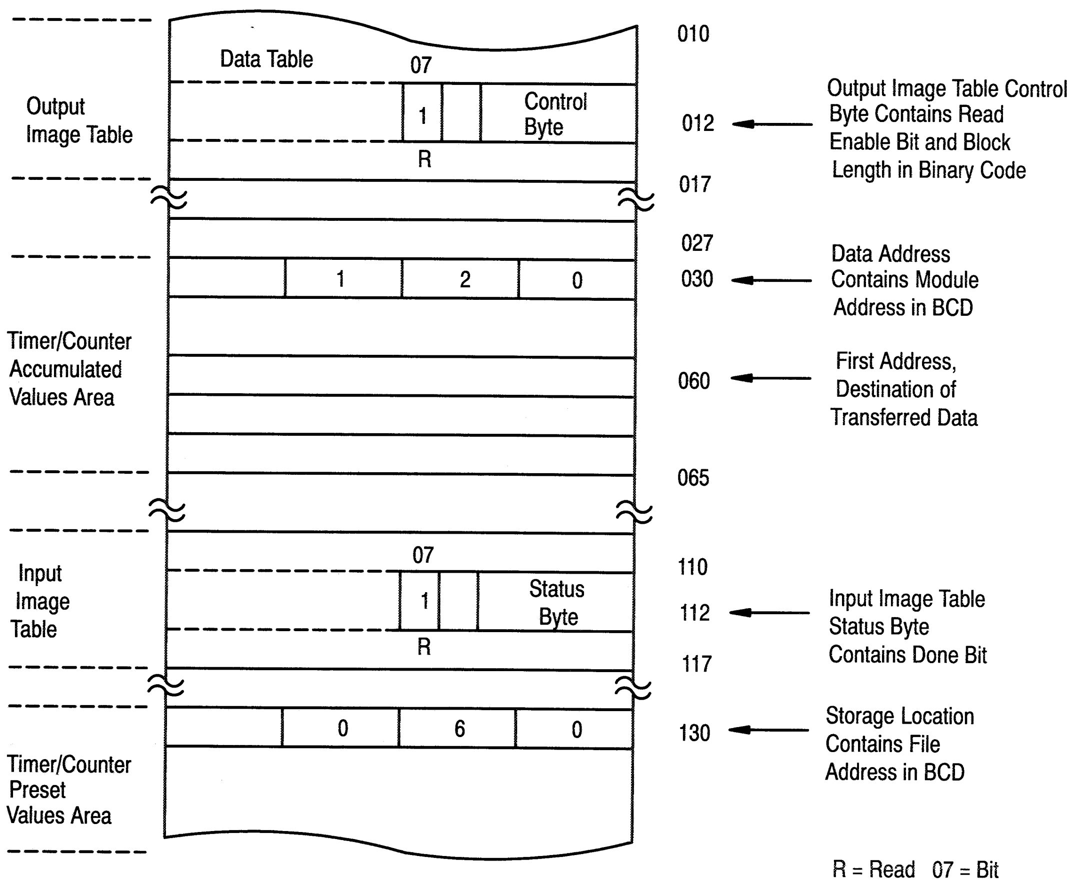
Rung 1: This rung is used to set four conditions.

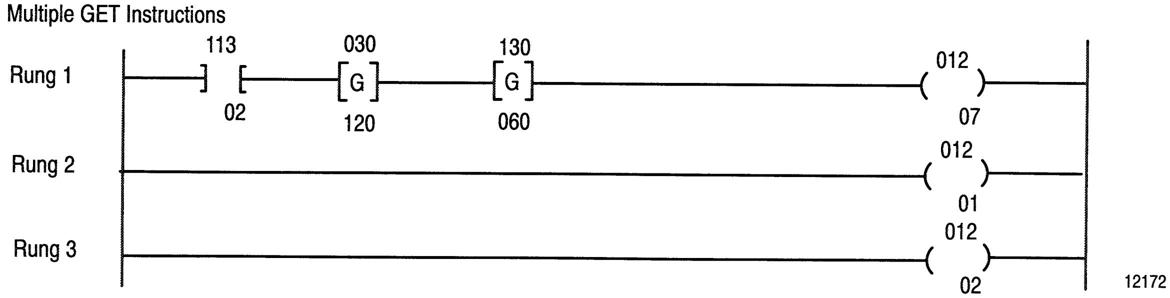
- **Examine On Instruction (113/02) -** This is an optional instruction. When used, block transfers will only be initiated when a certain action takes place. If you do not use this instruction, block transfers will be initiated every I/O scan.
- **First GET Instruction (030/120) -** identifies the module's 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_8 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)





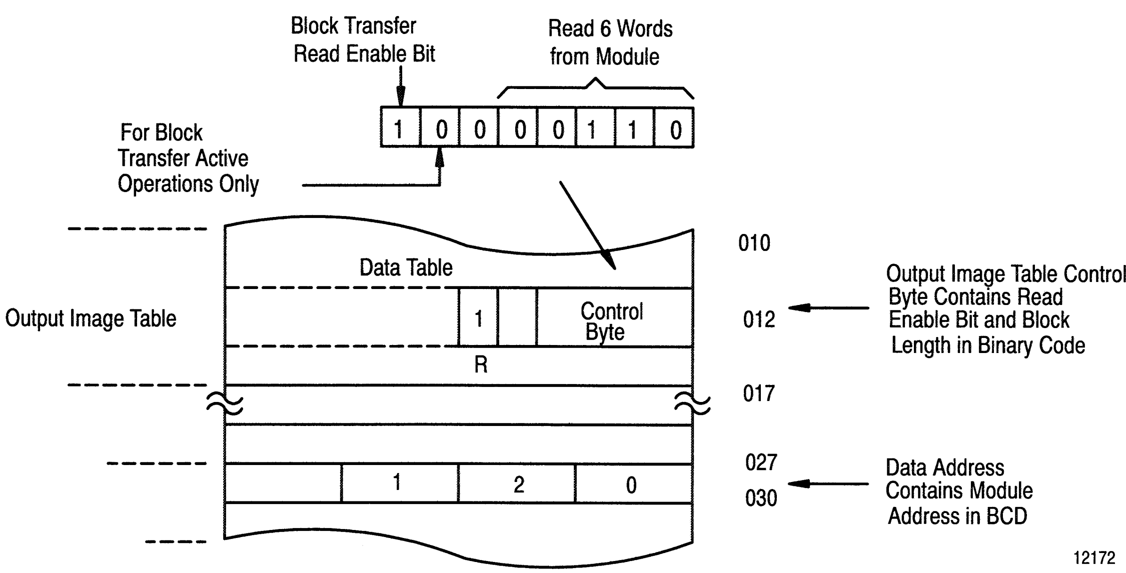
Setting the Block Length (Multiple GET Instructions only)

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)



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

Index

A
Accuracy, 2-3

B
Block transfer, 2-2
  block transfer read, 4-1
  block transfer write, 4-1
  format of BTR, 6-1
  with GET–GET instructions, D-1
  word assignments for BTR, 6-1
  word format, 5-6

C
Calibration
  how to, 7-1
  internal offset adjustment, 7-2
  test points and adjustments, 7-3
  tools required, 7-1
Compatibility, 1-2
  I/O table usage, 1-2
Configuration block, bit/word descriptions of, 5-6
Connecting wiring, 3-3
  grounding, 3-6

D
Data, reading from module, 6-1
Data table formats
  12–bit binary, C-2
  4–digit binary coded decimal, C-1
Data table usage, 1-3
Default, settings, 5-5
Diagnostics
  description, bits in status word, 8-1
  indicators, 8-1
  troubleshooting table, 8-3
Digital filtering, equation for determining, 5-2

E
Electrostatic discharge, 3-1

F
Features, configurable, 5-1
Field wiring arm, type, 3-3

I
Indicators
  fault, 3-7
  RUN indicator, 3-7
Input types, format, 5-1
Installation, initial, 3-1

K
Keying, position of keying bands, 3-2

L
Linear range, extended, 2-4, 4-5

M
Module description, 2-1
Module location, grouping, 3-2

O
Operational check, 2-3

P
Power requirements, 3-2
Program examples
  PLC–2, 4-1
  PLC–3, 4-3
  PLC–5, 4-4
  sample for PLC–2, B-1
  sample for PLC–3, B-3
  sample for PLC–5, B-4

R
Real time sampling
  mode of operation, 5-3
  table of bit settings, 5-4

S

Scaling
- explanation of, 5-4
- format, 5-4
- implementing, 5-4
- ranges, 5-5

Scan time, description, 4-5
Specifications, table of, A-1

V

Vocabulary, 1-1

A-B 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

Publication 1771-6.5.57 – July 1994
Supersedes Publication 1771-6.5.57 – November 1989

PN 955118–98