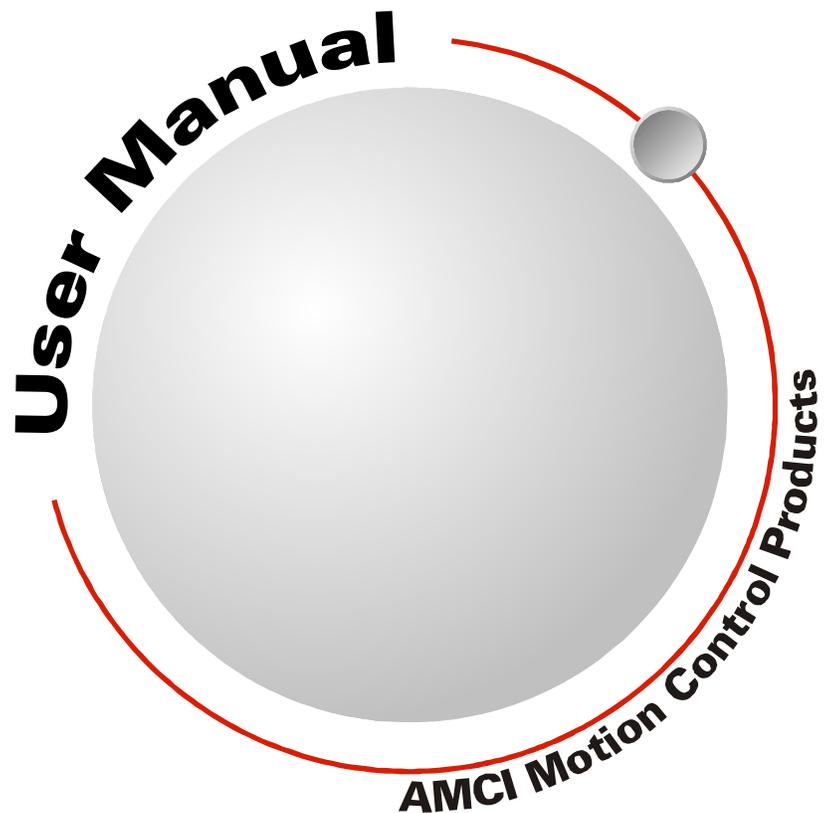


3102i

2 Axis Servo/Stepper Controller Module with Interpolated Moves



GENERAL INFORMATION

Important User Information

The products and application data described in this manual are useful in a wide variety of different applications. Therefore, the user and others responsible for applying these products described herein are responsible for determining the acceptability for each application. While efforts have been made to provide accurate information within this manual, AMCI assumes no responsibility for the application or the completeness of the information contained herein.

UNDER NO CIRCUMSTANCES WILL ADVANCED MICRO CONTROLS, INC. BE RESPONSIBLE OR LIABLE FOR ANY DAMAGES OR LOSSES, INCLUDING INDIRECT OR CONSEQUENTIAL DAMAGES OR LOSSES, ARISING FROM THE USE OF ANY INFORMATION CONTAINED WITHIN THIS MANUAL, OR THE USE OF ANY PRODUCTS OR SERVICES REFERENCED HEREIN.

No patent liability is assumed by AMCI, with respect to use of information, circuits, equipment, or software described in this manual.

The information contained within this manual is subject to change without notice.

This manual is copyright 2009 by Advanced Micro Controls Inc. You may reproduce this manual, in whole or in part, for your personal use, provided that this copyright notice is included. You may distribute copies of this complete manual in electronic format provided that they are unaltered from the version posted by Advanced Micro Controls Inc. on our official website: www.amci.com. You may incorporate portions of this documents in other literature for your own personal use provided that you include the notice "Portions of this document copyright 2009 by Advanced Micro Controls Inc." You may not alter the contents of this document or charge a fee for reproducing or distributing it.

Standard Warranty

ADVANCED MICRO CONTROLS, INC. warrants that all equipment manufactured by it will be free from defects, under normal use, in materials and workmanship for a period of [18] months. Within this warranty period, AMCI shall, at its option, repair or replace, free of charge, any equipment covered by this warranty which is returned, shipping charges prepaid, within eighteen months from date of invoice, and which upon examination proves to be defective in material or workmanship and not caused by accident, misuse, neglect, alteration, improper installation or improper testing.

The provisions of the "STANDARD WARRANTY" are the sole obligations of AMCI and excludes all other warranties expressed or implied. In no event shall AMCI be liable for incidental or consequential damages or for delay in performance of this warranty.

Returns Policy

All equipment being returned to AMCI for repair or replacement, regardless of warranty status, must have a Return Merchandise Authorization number issued by AMCI. Call (860) 585-1254 with the model number and serial number (if applicable) along with a description of the problem during regular business hours, Monday through Friday, 8AM - 5PM Eastern. An "RMA" number will be issued. Equipment must be shipped to AMCI with transportation charges prepaid. Title and risk of loss or damage remains with the customer until shipment is received by AMCI.

24 Hour Technical Support Number

24 Hour technical support is available on this product. If you have internet access, start at www.amci.com. Product documentation and FAQ's are available on the site that answer most common questions.

If you require additional technical support, call (860) 583-7271. Your call will be answered by the factory during regular business hours, Monday through Friday, 8AM - 5PM Eastern. During non-business hours an automated system will ask you to enter the telephone number you can be reached at. Please remember to include your area code. The system will page an engineer on call. Please have your product model number and a description of the problem ready before you call.

We Want Your Feedback

Manuals at AMCI are constantly evolving entities. Your questions and comments on this manual are both welcomed and necessary if this manual is to be improved. Please direct all comments to: Technical Documentation, AMCI, 20 Gear Drive, Terryville CT 06786, or fax us at (860) 584-1973. You can also e-mail your questions and comments to techsupport@amci.com

TABLE OF CONTENTS

General Information

| | |
|--|---|
| Important User Information | 2 |
| Standard Warranty | 2 |
| Returns Policy | 2 |
| 24 Hour Technical Support Number | 2 |
| We Want Your Feedback | 2 |

About this Manual

| | |
|--|---|
| Audience | 5 |
| Applicable Modules | 5 |
| Navigating this Manual | 5 |
| Manual Conventions | 6 |
| Trademarks and Other Legal Stuff | 6 |
| Revision Record | 6 |
| Old Revisions | 6 |
| Where To Go From Here | 7 |

Chapter 1: Introduction To The 3102i

| | |
|----------------------------|----|
| Module Overview | 9 |
| Front Panel LED's | 10 |
| Modes of Operation | 10 |
| Configuration Mode | 10 |
| Command Mode | 11 |
| Compatible Equipment | 11 |
| Servo Equipment | 11 |
| Stepper Equipment | 11 |
| Specifications | 12 |

Chapter 2: Move Profiles

| | |
|--|----|
| Units of Measure | 13 |
| Definition of Count Direction | 13 |
| Definition of Home Position | 14 |
| Definition of Starting Speed | 14 |
| Definition of Target Position | 14 |
| Definition of Acceleration Types | 15 |
| Linear Acceleration | 15 |
| Triangular S-Curve Acceleration .. | 15 |
| Trapezoidal S-Curve Acceleration | 15 |
| A Simple Move | 16 |
| Controlled and Immediate Stops | 17 |
| Backplane Control | 17 |
| Hardware Control | 17 |

Chp 2: Move Profiles (continued)

| | |
|-------------------------------|----|
| Profile Equations | 18 |
| Acceleration Equations | 18 |
| Total Time Equations | 19 |
| Available Move Types | 19 |
| Relative Move | 19 |
| Absolute Move | 19 |
| ±Jog Move | 20 |
| ±Registration Move | 21 |
| ±Blend Move | 22 |
| ±Encoder Follower Moves | 23 |

Chapter 3: Interpolated Move Profiles

| | |
|--------------------------------------|----|
| The X-Y Plane | 25 |
| Linear Interpolated Moves | 25 |
| Relative Linear Move | 25 |
| Absolute Linear Move | 26 |
| Required Parameters | 26 |
| Circular Interpolated Moves | 27 |
| Center Point Method | 27 |
| Radius Method | 28 |
| Via Point Method | 29 |
| Controlled and Immediate Stops | 29 |
| Backplane Control | 29 |
| Hardware Control | 30 |
| Welcome to the Integer World | 30 |

Chapter 4: Homing the 3102i

| | |
|-------------------------------------|----|
| Definition of Home Position | 31 |
| Position Preset | 31 |
| ±Find Home Commands | 31 |
| Homing Inputs | 31 |
| Physical Inputs | 31 |
| Backplane Inputs | 31 |
| Homing Configurations | 32 |
| Homing Profiles | 32 |
| Home Input Only Profile | 32 |
| Profile with Proximity Input | 33 |
| Profile with Overtravel Limit | 34 |

Chapter 5: INSTALLING THE 3102i

Location 35

Safe Handling Guidelines 35

 Prevent Electrostatic Damage 35

 Prevent Debris From Entering the Module 35

 Remove Power Before Servicing 35

Mounting 36

 Minimum Spacing 36

 Panel Mounting 36

 DIN Rail Mounting 37

System Assembly 37

Terminal Block Pinout 38

MS-40 Connector Kit 38

IM-3R Interface Module and Cable 39

General Wiring Guidelines 39

Driver Wiring 40

 Differential Wiring 40

 Open Collector Wiring 40

General Purpose Output Wiring 41

Discrete Input Wiring 42

Encoder Wiring 42

 Differential Encoder Wiring 42

 Single Ended Wiring 43

Diagnostic Feedback Wiring 44

Chapter 6: RSLogix 500 Configuration

Add the 3102i to Your Project 45

Finding the 3102i in Your Project 46

Data Location 46

PLC I/O Errors 47

Chapter 7: Configuration Mode Data Format

Output Data Format 49

 Configuration Bits MSW 50

 Configuration Bits LSW 51

 Starting Speed 51

 Homing Timeout 52

Input Data Format 52

Global Status Bits 53

Chp 7: Configuration Mode Data Format (continued)

Invalid Configurations 53

 Configuration Bits MSW 53

 Configuration Bits LSW 54

 Starting Speed Parameter 54

 Homing Timeout Parameter 54

 Reserved Words 54

Chapter 8: Command Mode Data Format

Multi-Word Parameters 55

Output Data Format 55

 Command Bits MSW 56

 Command Bits LSW 57

Command Bits Must Transition 57

Single Axis Command Blocks 58

 Absolute Move 58

 Relative Move 58

 Hold Move 59

 Resume Move 59

 Immediate Stop 60

 +Find Home (CW) 61

 -Find Home (CCW) 61

 +Jog Move (CW) 62

 +Registration Move (CW) 62

 +Encoder Follower Move 62

 -Jog Move (CCW) 63

 -Registration Move (CCW) 63

 -Encoder Follower Move (CCW) 63

 Preset Position 64

 Preset Encoder Position 64

 Reset Errors 64

 Run +Blend Move Profile (CW) 65

 Run -Blend Move Profile (CCW) 65

 Set Min. Registration Move Distance 65

Programming Blend Move Profiles 66

 Initial Profile Programming Block 67

 Secondary Profile Programming Block 68

TABLE OF CONTENTS

Chp 8: Command Mode Data Format (continued)

| | |
|--|----|
| Interpolated Move Command Blocks | 69 |
| Linear Interpolated Move | 69 |
| Via Point Circular | |
| Interpolated Move | 70 |
| Center Point Circular | |
| Interpolated Move | 71 |
| Radius Circular | |
| Interpolated Move | 72 |
| Writing Interpolated Move | |
| Commands to the 3102i | 73 |
| Input Data Format | 74 |
| Status Bits MSW | 74 |
| Global Status Bits | 75 |
| Status Bits LSW | 76 |
| Current Position | 77 |
| Encoder Position/Diagnostic | |
| Feedback Value | 77 |
| Capture Data | 77 |

Chapter 9: Tabulated Command & Error Codes

| | |
|-----------------------------------|----|
| Status Bits | 79 |
| Axis Stopped | 79 |
| Move Complete Bit | 79 |
| Home Invalid Bit | 79 |
| Invalid Profile Bit | 79 |
| Position Invalid Bit | 80 |
| Input Error Bit | 80 |
| Command Error Bit | 81 |
| Configuration Error Bit | 81 |
| Command Error Conditions | |
| By Command | 81 |
| ±Jog Move Command | 81 |
| ±Blend Move Command | 82 |
| Registration Move Command | 82 |
| Encoder Follower Operations | 82 |

Notes

ABOUT THIS MANUAL

Read this chapter to learn how to navigate through this manual and familiarize yourself with the conventions used in it. The last section of this chapter highlights the manual's remaining chapters and their target audience.

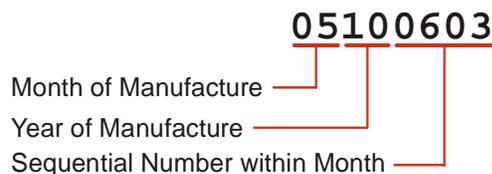
Audience

This manual explains the installation and operation of the 3102i Two Axis Servo/Stepper Controller Module from AMCI for the Rockwell Automation MicroLogix platform. It is written for the engineer responsible for incorporating the 3102i into a design as well as the engineer or technician responsible for its actual installation. If there are any unanswered questions after reading this manual, call the factory. An applications engineer will be available to assist you.

Applicable Modules

This manual is written to incorporate additions made to the 3102i firmware after the initial release of the module. These additions support programming a Circular Interpolated Move using the Via Point Method. Therefore, this manual is applicable to all 3102i modules, but a small number of modules do not have the Via Point functionality. Check the serial number of your module to determine if the module has the Via Point functionality.

The serial number on all 3102i modules has a date code embedded in the serial number. The format of AMCI serial numbers is shown below.



All 3102i modules that were manufactured prior to May, 2010 do not have the Via Point functionality. All 3102i modules manufactured after May, 2010 have the Via Point functionality. Additionally, modules that were manufactured in May, 2010 have the Via Point functionality if their sequential number is 0603 or greater.

Navigating this Manual

This manual is designed to be used in both printed and on-line forms. Every chapter is an even number of pages, making it easy to print individual chapters on a duplex, or double sided, printer. The manual's on-line form is a PDF document, which requires Adobe Acrobat Reader version 6.0+ to open it. Please note that the PDF document was created with version 9 of Adobe Acrobat. When you open this file with Acrobat Reader versions 6 through 8, you may see a warning message stating that the file was created with a later version of Acrobat. This warning can be safely ignored as this file has been tested with these versions of Acrobat Reader.

Bookmarks of all the chapter names, section headings, and sub-headings are in the PDF file to help you navigate through it. The bookmarks should have appeared when you opened the file. If they didn't, press the F5 key on Windows platforms to bring them up.

Throughout this manual you will also find *blue text that functions as a hyperlink* in HTML documents. Clicking on the text will immediately jump you to the referenced section of the manual. If you are reading a printed manual, most links include page numbers.

The PDF file is password protected to prevent changes to the document. You are allowed to select and copy sections for use in other documents and, if you own Adobe Acrobat version 6.0 or later, you are allowed to add notes and annotations.

Manual Conventions

Three icons are used to highlight important information in the manual:



NOTES highlight important concepts, decisions you must make, or the implications of those decisions.



CAUTIONS tell you when equipment may be damaged if the procedure is not followed properly.



WARNINGS tell you when people may be hurt or equipment may be damaged if the procedure is not followed properly.

The following table shows the text formatting conventions:

| Format | Description |
|------------------------|---|
| Normal Font | Font used throughout this manual. |
| <i>Emphasis Font</i> | Font used the first time a new term is introduced. |
| <i>Cross Reference</i> | When viewing the PDF version of the manual, clicking on the cross reference text jumps you to referenced section. |

Trademarks and Other Legal Stuff

The AMCI logo is a trademark, and “AMCI” is a registered trademark of Advanced Micro Controls Inc. “MicroLogix” is a trademark of Rockwell Automation. “Adobe” and “Acrobat” are registered trademarks of Adobe Systems Incorporated.

All other trademarks contained herein are the property of their respective holders.

Revision Record

This manual, 940-01031, is the second release of this manual. It was released on January 24th, 2010. This revision corrects typographical errors and improves descriptions. There are no new 3102i features introduced with this revision.

Old Revisions

- 940-01031: Added information on the Via Point Method. Released May 26th, 2010
- 940-01030: Initial Release, January 8, 2010.

ABOUT THIS MANUAL

Where To Go From Here

This manual contains information that is of interest to everyone from engineers to operators. The table below gives a brief description of each chapter's contents to help you find the information you need to do your job.

| CHP Num. | Chapter Title | Chapter Description |
|----------|--|---|
| 1 | <i>INTRODUCTION TO THE 3102i</i> | Intended for anyone new to the 3102i, this chapter gives a basic overview of the unit and lists complementary equipment available from AMCI. |
| 2 | <i>MOVE PROFILES</i> | Information on how to define moves with the 3102i and the various move types available with the module. |
| 3 | <i>HOMING THE 3102i</i> | For some applications, it is easier to define moves in terms of actual positions on the machine. Before you can run these moves you must define a home position on the machine. This chapter gives detailed information on how the 3102i can help you define this position. |
| 4 | <i>INSTALLING THE 3102i</i> | Information on physically installing the 3102i in your system, including PLC installation and I/O wiring. |
| 5 | <i>RSLogix 500 CONFIGURATION</i> | This chapter explains how to add a 3102i to an RSLogix 500 project. |
| 6 | <i>CONFIGURATION MODE DATA FORMAT</i> | This chapter explains how to configure the 3102i for use in your application. |
| 7 | <i>COMMAND MODE DATA FORMAT</i> | This chapter explains the formats of the commands you can issue to the 3102i. |
| 8 | <i>TABULATED COMMAND & ERROR CODES</i> | This chapter lists the bits used by the 3102i to signal error conditions and the reasons why the bits may be set. |

Notes

INTRODUCTION TO THE 3102i

Module Overview

The 3102i is the second servo/stepper controller for the MicroLogix 1100, 1200, and 1400 platforms. The module offers two independent motion control axes that function in an open-loop configuration. Each axis can be used to drive a stepper or a servo with step/direction input capability. These capabilities are the same as the 3102i servo/stepper controller, also available from AMCI.

The 3102i can also synchronize the two axes, giving you the ability to control linear and circular motions in an XY plane.

The 3102i requires sixteen input words and sixteen output words from the PLC. Power draw from the PLC does not affect the current maximum expansion modules supported by the MicroLogix controllers. This allows a MicroLogix system to drive up to ten axes with a single PLC.

Each axis of the 3102i offers a full 32 bit (± 2 billion+) motor position register, move lengths of up to 2^{31} (± 1 billion+) counts, programmable S-curve acceleration types, five discrete inputs for various control functions and a differential encoder input.

The figure to the right shows a 3102i with its front door open. The 3102i requires a forty pin, high density connector for all of its I/O connections. The I/O connections for axis 1 are on the right column of the connector and connections to axis 2 are on the left.

The 3102i has the following I/O connections for each axis:

- ▶ Differential outputs to your servo or stepper driver. Maximum output frequency is 1 MHz.
- ▶ Single ended outputs to servo or stepper driver. Use these outputs if your driver does not have differential inputs. Maximum output frequency is 100 KHz. You cannot use the differential and single-ended output at the same time.
- ▶ Differential quadrature encoder inputs. An encoder can be used by the 3102i for open loop position feedback or in *Encoder Follower Mode*. In Encoder Follower Mode, the 3102i will generate pulses on the outputs in response to changes on the encoder inputs. This allows you to use the 3102i in electronic gearing applications. The 3102i also allows you to use the Z input on the encoder to define the *Home Position* of the machine. The Home Position is a reference position on the machine that you must define before running one type of move available with the 3102i.
- ▶ Home Input. This single ended input is typically used when defining the home position on the machine.
- ▶ CW Limit and CCW Limit Inputs. These single ended inputs are used to define the maximum clockwise and counter-clockwise positions on the machine. If one of these inputs becomes active while traveling in that direction, the 3102i will immediately stop the move.
- ▶ Emergency Stop Inputs. If one or both of the CW and CCW Limits are not required, then the input can be configured as an Emergency Stop input. The 3102i will immediately stop the move if an Emergency Stop input becomes active.
- ▶ Capture Input. This single ended input can be used to capture the motor position or encoder position during a move. This is useful in applications where you must capture the position value and the event is too short to be captured by the PLC.
- ▶ External Input. A single ended input that can be used to bring moves to a controlled stop.
- ▶ General Purpose Output. This single ended output is controlled through a bit from the PLC.



Figure 1.1 3102i Module

Module Overview (continued)

Front Panel LED's

The 3102i has five LED's to indicate the status of the module and each of its axes. The POWER LED shows the status of the module while the OK and ERROR LED's show you the status of the axis. The table below lists the information reported by the LED's.

| LED | State | Description |
|--------------|----------|---|
| Power | On | Module has power, has passed its internal diagnostics, and is communicating with the PLC. |
| | Off | Any of the above are not true |
| Axis 1 OK | On | Axis is OK with no motion in progress |
| | Blinking | Motion is occurring on the axis |
| | Off | Axis is disabled (Axis 1 Error will also be off.) |
| Axis 1 Error | On | PLC communication has been interrupted. |
| | Blinking | A Configuration, Command, Input, Homing, or Invalid Profile Error exists on the axis. |
| | Off | No errors, or the axis is disabled (Axis 1 OK will also be off in this case.) |
| Axis 2 OK | On | Axis is OK with no motion in progress |
| | Blinking | Motion is occurring on the axis |
| | Off | Axis is disabled (Axis 2 Error will also be off.) |
| Axis 2 Error | On | PLC communication has been interrupted. |
| | Blinking | A Configuration, Command, Input, Homing, or Invalid Profile Error exists on the axis. |
| | Off | No errors, or the axis is disabled (Axis 2 OK will also be off in this case.) |

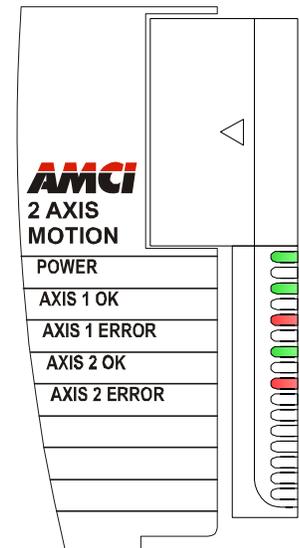


Figure 1.2 Status LED's

Table 1.1 Front Panel LED Description

Modes of Operation

Each axis of the 3102i has two modes of operation, *Configuration Mode* and *Command Mode*.

Configuration Mode

This mode give you that ability to configure the axis for your application without having to set any programming switches and change the axis configuration at any time. The axis must be configured after every power up. The following parameters are set in Configuration Mode:

- The enabled or disabled state of each of the five inputs
- The configuration of the CW Limit and CCW Limit as overtravel limits or Emergency Stop inputs
- The enabled or disabled state of the Backplane Home Proximity bit
- If a quadrature encoder will be used or not
- If Diagnostic Feedback will be used to test the 3102i axis
- The Output Pulse type (CW/CCW or Step & Direction)
- If a homing operation will be to the Home Input or the encoder marker pulse
- The Starting Speed of most moves available with the 3102i. This parameter allows you to start a move above the motor's low frequency resonance point and is also useful in micro-stepping applications.
- The Homing Timeout value. This parameter allows you to automatically stop a homing operation if it takes longer to occur than expected.



Modes of Operation (continued)

Command Mode

This mode give you the ability to set the machine's home position, program, carry out, and control servo or stepper motor moves, detect and reset errors as they occur, and run diagnostics on the I/O if needed.

- Allows you to set the home position in the machine
- Allows you to preset the motor position register
- Allows you to preset the encoder position register
- Allows you to run six different single-axis move types
 - Relative Move
 - Absolute Move
 - Jog
 - Registration Move
 - Blend Move
 - Encoder Follower

See chapter 2, **MOVE PROFILES**, starting on pg 15, for a complete description of these moves.

- Allows you to run Linear and Circular Interpolated Moves.
See chapter 2, **MOVE PROFILES**, starting on pg 15, for a complete description of these moves.
- Allows you to hold and resume relative and absolute moves
- Allows you to bring a move to an immediate stop via a backplane bit or hard-wired input
- Allows you to reset command errors so a new command can be issued
- Allows you to feed the driver outputs into the encoder inputs and perform diagnostics.

Compatible Equipment

Servo Equipment

As of the release of this manual, the 3102i has been tested with the Rockwell Automation Ultra 1500 and Ultra 3000 servo drives. Due to its differential and single ended outputs, the 3102i should be compatible with drives from any manufacturer. If you have any questions concerning compatibility, check our website for an up to date list of known compatible equipment or contact AMCI Sales or Technical Support for information.

Stepper Equipment

AMCI has a broad range of stepper drivers and motors that are compatible with the 3102i.

| Product Family | Description |
|---------------------------|---|
| SD7540 Stepper Driver | A small, (2.2" x 3" x 1") DC powered micro-step driver with a programmable output current of up to 4 amps. |
| 'SD170xx' Stepper Drivers | A complete line of 120Vac powered drivers that offer 170Vdc motor bus and programmable motor currents up to 6 amps. |
| SD31045 Stepper Driver | A 240Vac powered driver that offers a 310Vdc motor bus and programmable motor current up to 4.5 amps. |
| 'SM' Stepper Motors | A line of NEMA size 23, 34, and 42 motors with holding torques in the range of 130 to 1,870 oz-in. They are all available with optional optical encoders. |
| 'SMD' Stepper Motors | Our NEMA size 23 motors with the SD7540 driver mounted on the back, these motors offer high performance for space constrained applications. |

Table 1.2 AMCI Compatible Equipment

Our stepper product family is constantly growing. Please go to our website at <http://www.amci.com> for information on these products and others that have been added since the release of this manual.

With the differential and single-ended outputs available on the module, the 3102i should be compatible with most third party stepper drives available today. Feel free to contact AMCI Sales or Technical Support for additional information on any third party stepper equipment you may be using in your project.



Specifications

Compatible Platforms

MicroLogix 1100, 1200, and 1400

Number of Axes

Two axes. Can be operated independently or synchronized for linear and circular interpolated moves.

Motor Control Outputs

Differential and single-ended output available.

Programmable CW/CCW or Step & Direction formats

Discrete Inputs

Five per Axis

Home, Capture, CW Limit, CCW Limit, and External

CCW Limit and CCW Limit inputs can be programmed to act as Emergency Stop inputs.

Discrete Outputs

One per Axis

Output state controlled by bit from PLC.

Encoder Inputs

One Encoder per Axis

+5Vdc differential ABZ inputs

X4 decoding always used by 3102i

Maximum pulse train frequency: 250KHz
(1MHz maximum counting frequency)

Encoder can be used for position feedback, homing to Z pulse, and with Encoder Follower Mode.

Maximum Number of Modules

1100: 4 per processor

1200: 3 per processor

1400: 5 per processor

Maximum number assumes that the 3102i's are the only expansion modules used in the system.

Number of I/O Words per Module

16 Input Words

16 Output Words

Output Wiring

40-pin MIL-C-83503 Header, Keyed

AMCI IM-3R breakout box or Rockwell Automation 1746-N3 Connector Kit can be used for field wiring.

Differential Stepper Outputs

+5Vdc differential (3.5Vdc to 4Vdc peak typical)

Maximum Output Current: 20mA

Maximum Output Frequency: 1MHz

Single-Ended Stepper Outputs

Open Collector Sinking

Voltage Range: 12Vdc to 24Vdc

Maximum Output Current: 30mA

Maximum Output Frequency: 100KHz

An external supply is required for operation.

General Purpose Output

Same electrical specifications as the Single-Ended Stepper Outputs

Input Specifications

Open Collector Sinking

All inputs share a common return

ON Voltage Range: 8Vdc to 24Vdc

OFF Voltage Range: 0Vdc to 2Vdc

Maximum Input Current: 15mA @ 24Vdc

An external supply is required for operation.

Throughput Time

100 microseconds

Inputs must remain in a steady state for 100 microseconds after a transition for the transition to be recognized.

Programming Storage

RAM Memory.

3102i must be configured on power up.

Backplane Current Draw

110mA from 5Vdc bus

85mA from 24Vdc bus

Environmental Conditions

Operating Temperature: -20 to 65°C
(-4 to 149°F)

Relative Humidity: 5 to 95%
(w/o condensation)

Operating Altitude: Up to 2000m (6560 ft)

Storage Temperature: -40 to 85°C
(-40 to 185°C)

CHAPTER 2

MOVE PROFILES

When a move command is sent to the 3102i, the module calculates the entire profile before starting the move or issuing an error message. This chapter explains how the single axis move profiles are calculated and the different moves available.

Units of Measure

Distance: Every distance is measured in steps. Your driver and motor combination will give you a specific number of steps needed to complete one rotation of the motor shaft. It is up to you to determine how many steps are required to travel the required distance in your application.

Speed: All speeds are measured in steps/second. Since the number of steps needed to complete one shaft rotation is determined by your driver and motor combination, it is up to you to determine how many steps per second is required to rotate the motor shaft at your desired speed.

Acceleration: The formulas in the rest of this chapter use a unit of measure of steps/second/second (steps/second²) for acceleration and deceleration. However, when programming the 3102i, all acceleration and deceleration values must be programmed in the unit of measure of steps/millisecond/second.

- To convert from steps/second² to steps/millisecond/second, divide the value by 1000. This must be done when converting from a value used in the equations to a value programmed into the 3102i.
- To convert from steps/millisecond/second to steps/second², multiply the value by 1000. This must be done when converting from the value programmed into the 3102i to the value used in the equations.

Definition of Count Direction

Each axis of the 3102i can be configured to output motor pulses in one of two formats, CW/CCW or Step & Direction. The two formats are shown in the figure below. In this figure, a high signal shows the output's active state, while a low signal shows its inactive state. For the differential outputs, an active state is when the +output has a positive voltage with respect to the -output. For the single ended sinking outputs, the active state is when the output is pulling the signal to ground.

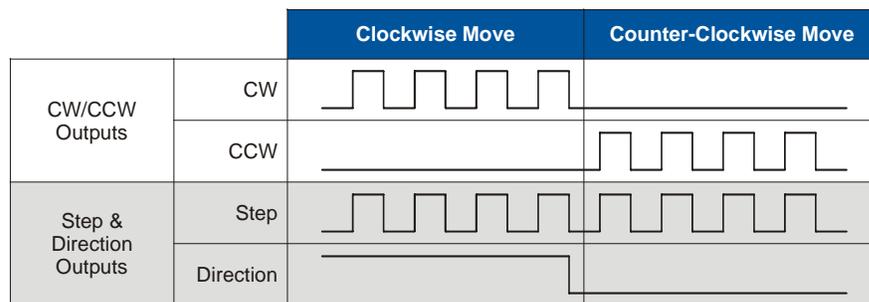


Figure 2.1 Directional Outputs



For single axis, non-interpolated moves, clockwise moves will always increase the motor position data reported back to the PLC. Some of the moves, such as the Jog Move, have a positive and negative command. A positive command, such as the +Jog Move command, will output pulses for a clockwise move. For interpolated moves, the direction of motion is defined in the X-Y plane defined by the two axes. **INTERPOLATED MOVE PROFILES** are introduced in chapter 3, starting on page 27.



Definition of Home Position

The Home Position is any position on your machine that you can sense and stop at. One way of defining the Home Position is by presetting the motor position register with the Preset Position command that is available with the 3102i. The other is by using one of the module's \pm Find Home commands. If you use one of the \pm Find Home commands, the motor position and encoder registers will automatically be set to zero once the home position is reached. Defining a Home Position is completely optional. Some applications, such as those that use a servo or stepper for speed control, don't require position data at all.

Definition of Starting Speed

The Starting Speed is the speed that most moves will begin and end at. This value is set while configuring the axis and it has a valid range of 1 to 1,000,000 steps/second. This value is typically used to start the move above the motor's low frequency resonances and in micro-stepping applications to limit the amount of time needed for acceleration and deceleration. AMCI does not specify a default value in this manual because it is very dependent on motor size and attached load.

Definition of Target Position

The Target Position is the position that you want the move to end at. There are two ways to define the Target Position, either with relative coordinates or with absolute coordinates.

Relative coordinates define the Target Position as an offset from the present position of the motor. Most 3102i moves use relative coordinates.

- The range of values for the Target Position when it is treated as a relative offset is $\pm 1,073,741,823$ counts. Positive offsets will result in clockwise moves, while negative offsets result in counter-clockwise moves.
- The Current Motor Position register that is reported back to the PLC is a full 32 bit register. ($\pm 2,147,483,647$ counts) The only way to move beyond $\pm 1,073,741,823$ counts is with multiple relative moves.

Absolute coordinates treat the Target Position as an actual position on the machine. Note that you must set the Home Position on the machine before you can run an Absolute Move. (See [Definition of Home Position](#) on the previous page.)

- The range of values for the Target Position when it is treated as an actual position on the machine is $\pm 1,073,741,823$ counts. The move will be clockwise if the Target Position is greater than the Current Position and negative if the Target Position is less than the Current Position.
- The Current Motor Position register that is reported back to the PLC is a full 32 bit register. ($\pm 2,147,483,647$ counts) However, you cannot move beyond $\pm 1,073,741,823$ counts with an Absolute Move.

Definition of Acceleration Types

With the exception of homing operations, all of the move commands allow you to define the acceleration type used during the move. The 3102i supports three types of accelerations and decelerations.

Linear Acceleration

The axis accelerates (or decelerates) at a constant rate until the programmed speed is reached. This offers the fastest acceleration, but consideration must be given to insure the smoothest transition from rest to the acceleration phase of the move. The smoothest transition occurs when the configured Starting Speed is equal to the square root of the programmed Linear Acceleration. Note that other values will work correctly, but you may notice a quick change in velocity at the beginning of the acceleration phase.

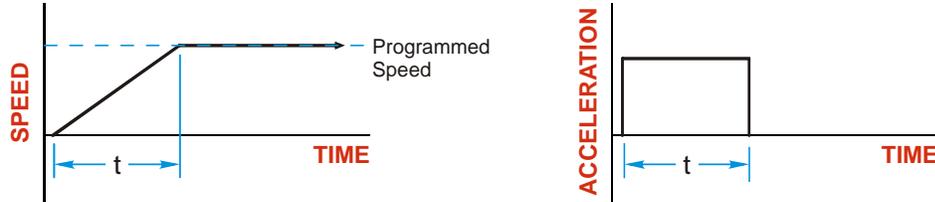


Figure 2.2 Linear Acceleration

Triangular S-Curve Acceleration

The axis accelerates (or decelerates) at a constantly changing rate that is slowest at the beginning and end of the acceleration phase of the move. The Triangular S-Curve type offers the smoothest acceleration, but it takes twice as long as a Linear Acceleration to achieve the same velocity.



Figure 2.3 Triangular S-Curve Acceleration

Trapezoidal S-Curve Acceleration

The Trapezoidal S-Curve acceleration is a good compromise between the speed of Linear acceleration and the smoothness of Triangular S-Curve acceleration. Like the Triangular S-Curve, this acceleration type begins and ends the acceleration phase smoothly, but the middle half of the acceleration phase is linear. Because of this, the Trapezoidal S-Curve acceleration only requires 33% more time to achieve the same velocity as a Linear Acceleration, compared to the 100% more time of a Triangular S-Curve acceleration.

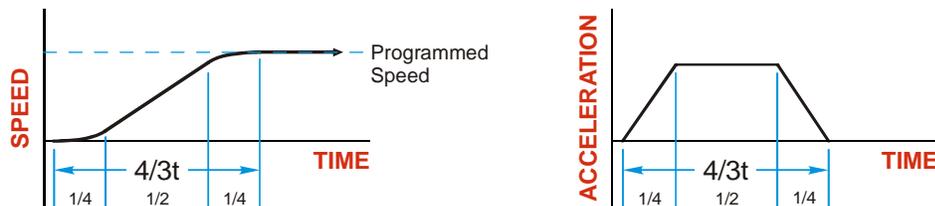


Figure 2.4 Trapezoidal S-Curve Acceleration



A Simple Move

As shown in the figure below, a move from A (Current Position) to B (Target Position) consists of several parts.

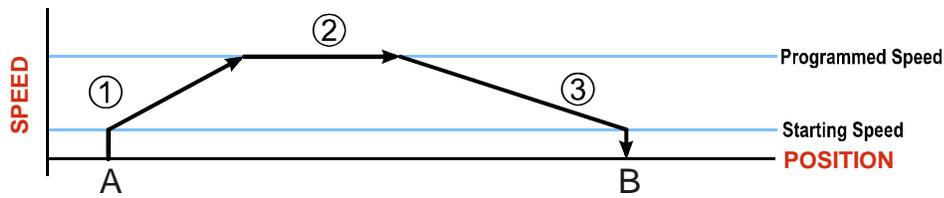


Figure 2.5 A Trapezoidal Profile

- 1) The move begins at point A, where the motor jumps from rest to the configured *Starting Speed*. The motor then accelerates at the programmed *Acceleration Value* until the speed of the motor reaches the *Programmed Speed*. Both the *Acceleration Value* and the *Programmed Speed* are programmed when the move command is sent to the 3102i.
- 2) The motor continues to run at the *Programmed Speed* until it reaches the point where it must decelerate before reaching point B.
- 3) The motor decelerates at the *Deceleration Value*, which is also programmed by the move command, until the speed reaches the *Starting Speed*, which occurs at the *Target Position (B)*. The motor stops at this point. Note that the acceleration and deceleration values can be different in the move.

Figure 2.5 above shows a Trapezoidal Profile. A Trapezoidal Profile occurs when the *Programmed Speed* is reached during the move. This occurs when the number of steps needed to accelerate and decelerate are less than the total number of steps in the move.

Figure 2.6 below shows a Triangular Profile. A Triangular Profile occurs when the number of steps needed to accelerate to the *Programmed Speed* and decelerate from the *Programmed Speed* are greater than the total number of steps in the move. In this case, the profile will accelerate as far as it can before decelerating and the *Programmed Speed* is never reached.

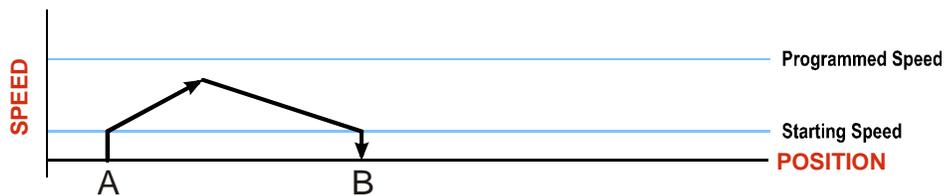


Figure 2.6 A Triangular Profile

Controlled and Immediate Stops

Once a move is started, there are several ways to stop the move before it comes to an end. These stops are broken down into two types:

- ▶ **Controlled Stop:** The axis immediately begins decelerating at the move's programmed deceleration value until it reaches the configured Starting Speed. The axis stops at this point. The motor position value is still considered valid after a Controlled Stop and the machine does not need to be homed before Absolute Moves can be run again.
- ▶ **Immediate Stop:** The axis immediately stops outputting pulses regardless of the speed the motor is running at. Because it is possible for the inertia of the load attached to the motor to pull the motor beyond the stopping point, the motor position value is considered invalid after an Immediate Stop and the machine must be homed before Absolute Moves can be run again.

Backplane Control

Hold Move Command: This command can be used with some moves to bring the axis to a Controlled Stop. Not all moves are affected by this command. The section [Available Move Types](#), starting on page 21, describes each move type in detail, including if the move is affected by this command.

Immediate Stop Command: When this command is issued from the PLC, the axis will come to an Immediate Stop. The move cannot be restarted and the machine must be homed before Absolute Moves can be run again.

Hardware Control

External Input: The External Input can be used with some moves to bring the axis to a Controlled Stop. Not all moves are affected by this input. The section [Available Move Types](#), starting on page 21, describes each move type in detail, including if the move is affected by this input.

CW Limit and CCW Limit Inputs: In most cases, activating these inputs during a move will bring the axis to an Immediate Stop. The exceptions are the \pm Find Home commands, the \pm Jog Move commands, and the \pm Registration Move commands. The \pm Find Home commands are explained in chapter 3, [HOMING THE 3102i](#), which starts on page 33. The \pm Jog Move commands are fully explained on page 22, and the \pm Registration Move commands are fully explained on page 23.

Emergency Stop Input: It is possible to configure the CW Limit Input and/or the CCW Limit Input as an Emergency Stop Input. When an Emergency Stop Input is activated, the axis will come to an Immediate Stop, regardless of the direction of travel.

Profile Equations

This section was added because some of our customers must program very precise profiles. Understanding this section is not necessary before programming the 3102i and it can be considered optional.

The equations in this section allow you to calculate the number of steps and time needed to accelerate and decelerate as well as the time required for the complete move. You will also be able to determine if your move will generate a Trapezoidal or Triangular Profile.

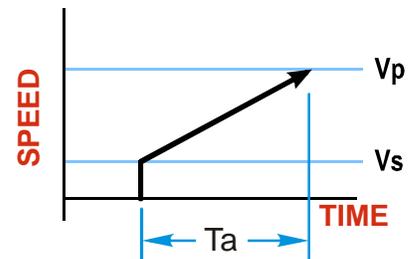
The equations in the rest of this chapter use a unit of measure of steps/second/second (steps/second²) for acceleration and deceleration. However, when programming the 3102i, all acceleration and deceleration values must be programmed in the unit of measure of steps/millisecond/second.

- To convert from steps/second² to steps/millisecond/second, divide the value by 1000. This must be done when converting from a value used in the equations to a value programmed into the 3102i.
- To convert from steps/millisecond/second to steps/second², multiply the value by 1000. This must be done when converting from the value programmed into the 3102i to the value used in the equations.

Acceleration Equations

The following variables are used in these equations:

- **a** = Acceleration/deceleration value. Must be in the units of steps/second²
- **T_A or T_D** = Time needed to complete the acceleration or deceleration phase of the move
- **D_A or D_D** = Number of Steps needed to complete the acceleration or deceleration phase of the move
- **V_S** = Configured Starting Speed of the move
- **V_P** = Programmed Speed of the move



| Acceleration Type | T _A or T _D (Time to Accelerate or Decelerate) | D _A or D _D (Distance to Accelerate or Decelerate) | a (Acceleration value based on T _A) |
|---------------------|--|--|--|
| Linear | $T_A = (V_P - V_S)/a$ | $D_A = T_A * (V_P + V_S)/2$ | $a = (V_P - V_S)/T_A$ |
| Triangular S-Curve | $T_A = 2((V_P - V_S)/a)$ | $D_A = T_A * (V_P + V_S)/2^{\dagger}$ | $a = (V_P - V_S)/T_A^{\dagger}$ |
| Trapezoidal S-Curve | $T_A = 4/3((V_P - V_S)/a)$ | $D_A = T_A * (V_P + V_S)/2^{\ddagger}$ | $a = (V_P - V_S)/T_A^{\ddagger}$ |

[†]For these equations, $T_A = 2((V_P - V_S)/a)$

[‡]For these equations, $T_A = 4/3((V_P - V_S)/a)$

Table 2.1 Acceleration Equations

If the sum of the D_A and D_D values for of the move is *less than* the total number of steps in the move, your move will have a Trapezoidal profile.

If the sum of the D_A and D_D values for of the move is *equal to* the total number of steps in the move, your move will have a Triangular profile and your move will reach the Programmed Speed before it begins to decelerate.

If the sum of the D_A and D_D values for of the move is *greater than* the total number of steps in the move, your move will have a Triangular profile and your move *will not* reach the Programmed Speed before it begins to decelerate. You can determine your maximum running speed by substituting your T_A equation into your D_A equation and solving for V_P. The value of D_A that you use will depend on the ratio of the acceleration and deceleration values. If both values are the same, use a value of D_A/2 in your equation. Once you have determined your maximum running speed, you can determine your T_A and T_D values.

Profile Equations (continued)

Total Time Equations

For Trapezoidal Profiles you must first determine the number of counts that you are running at the Programmed Speed. This value, (D_P below), is equal to your D_A and D_D values subtracted from your total travel. You can then calculate your total profile time, (T_T below), from the second equation.

$$D_P = (\text{Total Number of Steps}) - (D_A + D_D)$$

$$T_T = T_A + T_D + D_P/V_P$$

For Triangular Profiles, the total time of travel is simply:

$$T_T = T_A + T_D$$

Available Move Types

This section describes the six single-axis move types available with the 3102i. It does not describe the Linear or Circular Interpolated Move Types that coordinate motion between the two axes. For information on these move types, refer to chapter 3, *INTERPOLATED MOVE PROFILES*, starting on page 27.

Relative Move

Relative Moves move an offset number of steps (n) from the Current Position (A). A trapezoidal profile is shown to the right, but Relative Moves can also generate triangular profiles. The current position can be any value and the Home Position does not have to be set before a Relative Move can be run.

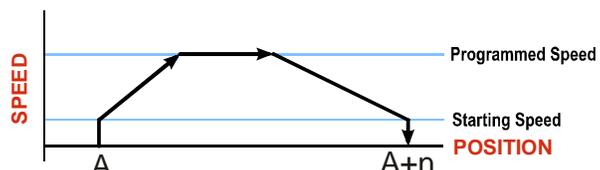


Figure 2.7 Relative Move

The command's Target Position registers hold the move's offset. The offset can be in the range of $\pm 1,073,741,823$ counts. Positive offsets will result in clockwise moves, while negative offsets result in counter-clockwise moves.

Relative Moves can be brought to a Controlled Stop by using the Hold Move Command from the backplane or by triggering the External Input. When either of these two inputs are used, the axis will immediately decelerate at the programmed rate and stop. When stopped successfully, the 3102i will set a *Hold State* bit in the input data table. The Relative Move can be restarted with the Resume Move command from the backplane or the move can be aborted. The Resume Move command allows you to change the move's Programmed Speed, Acceleration Value and Type, and the Deceleration Value and Type. The Target Position cannot be changed with the Resume Move Command.

If the External Input is active when a Relative Move is started, the 3102i will output one step in the specified direction before entering the Hold State.

Absolute Move

Absolute Moves move from the Current Position (A) to a given position (B). (The 3102i calculates the number of steps needed to move to the given position and moves that number of steps.) A trapezoidal profile is shown to the right, but Absolute Moves can also generate triangular profiles.

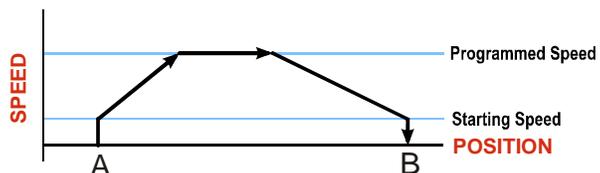


Figure 2.8 Absolute Move

NOTE The *Home Position* on the machine must be set before running an Absolute Move.

Available Move Types (continued)**Absolute Move (continued)**

One way of setting the Home Position is by presetting the motor position register from the backplane. The other method is by using the \pm Find Home commands. Once one of the \pm Find Home commands is issued, the 3102i will independently drive the axis in the programmed direction until it finds the Home Position. The Home Position is defined by sensors attached to the 3102i. A complete description of how to use the Find Home commands is given in chapter 3, *HOMING THE 3102i*, which starts on page 33.

- ▶ The Target Position registers of an Absolute Move hold the move's given position. This value can be in the range of $\pm 1,073,741,823$ counts. The move will be clockwise if the Target Position is greater than the Current Position and counter-clockwise if the Target Position is less than the Current Position.

Absolute Moves can be brought to a Controlled Stop by using the Hold Move Command from the backplane or by triggering the External Input. When either of these two inputs are used, the axis will immediately decelerate at the programmed rate and stop. When stopped successfully, the 3102i will set a *Hold State* bit in the input data table. The Absolute Move can be restarted with the Resume Move command from the backplane or the move can be aborted. The Resume Move command allows you to change the move's Programmed Speed, Acceleration Value and Type, and the Deceleration Value and Type. The Target Position cannot be changed with the Resume Move command.

If the External Input is active when an Absolute Move is started, the 3102i will output one step in the specified direction before entering the Hold State.

 \pm Jog Move

Jog Moves move in the programmed direction as long as the command is active. Two commands are available, the +Jog Move will output CW steps while the -Jog Move will output CCW steps. These commands are often used to give the operator manual control over the axis.

NOTE  The Target Position value must be zero during a Jog Move. If it is not zero, the move will actually be a *\pm Registration Move*, and the final stopping position may be unexpected.

Jog Moves can be brought to a Controlled Stop by turning off the Jog Move command bit from the backplane or activating the External Input. The axis will decelerate at the programmed rate to the configured Starting Speed and stop. The 3102i will output one step in the specified direction if the External Input is active when a Jog Move is started.

The CW Limit and CCW Limit inputs behave differently for \pm Jog Moves than all other move types except for \pm Registration Moves. Like all moves, activating a limit that is the same as the direction of travel, for example activating the CW Limit during a +Jog Move, will bring the move to an Immediate Stop. Unlike the other moves, activating a limit that is opposite to the direction of travel, for example activating the CCW Limit during a +Jog Move, has no effect. This allows you to jog off of the activated limit switch.

If the initial Programmed Speed is greater than the configured Starting Speed when the command is issued, then the move's Programmed Speed, Acceleration Value and Type, and Deceleration Value and Type can be changed while the move is running. The axis will accelerate or decelerate to the new Programmed Speed when it is changed. Note that the acceleration and deceleration parameters are not changed unless the Programmed Speed is also changed.

If the initial Programmed Speed is set to a value less the configured Starting Speed when the command is issued, the 3102i will perform a *Constant Speed Jog*. The move will run at the Programmed Speed only. The move's Programmed Speed, Acceleration Value and Type, and Deceleration Value and Type cannot be changed while the move is running. Attempting to do so will cause an error.

If the Programmed Speed is set to zero when the command is issued, the 3102i will perform a *One Shot Jog* and output one pulse in the specified direction. The pulse is output at the configured Starting Speed.

Available Move Types (continued)

±Registration Move

Similar to a Jog Move, a Registration Move will travel in the programmed direction as long as the command is active. +Registration Moves result in CW output pulses, -Registration Moves result in CCW output pulses. When the command terminates under Controlled Stop conditions, the 3102i will output a programmed number of steps as part of bringing the move to a stop. Controlled Stop conditions are resetting the command bit from the backplane or activating the External Input. Note that all position values programmed with a Registration Move are relative values, not absolute machine positions.

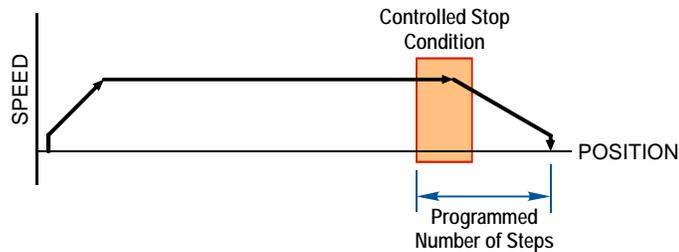


Figure 2.9 Registration Move

If the Programmed Number of Steps are less than the number of steps needed to bring the axis to a stop based on the Programmed Speed and Deceleration values set with the command, the 3102i will issue an error message instead of starting the move. An error message will also be issued if the External Input is active when a Registration Move is initiated.

Like the ±Jog Moves, activating a limit that is opposite to the direction of travel, for example activating the CCW Limit during a +Registration Move, has no effect.

An additional feature of the 3102i is the ability to program the module to ignore the Controlled Stop conditions until a minimum number of steps have occurred. This value is programmed through the Minimum Registration Move Distance parameter, which is set through a separate command. This value is stored until the module is re-configured. Figure 2.10 shows how the Minimum Registration Move Distance parameter affects when the Stop Condition is applied to the move. As shown in the second diagram, Controlled Stop conditions are level triggered, not edge triggered. If a Controlled Stop Condition occurs before the Minimum Registration Move Distance is reached and it stays active, the move will begin its controlled stop once the Minimum Registration Move Distance is reached.

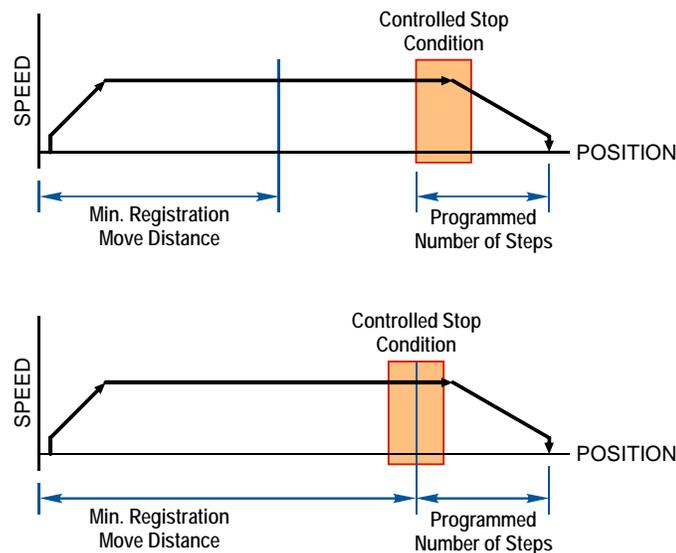


Figure 2.10 Min. Registration Move Distance

Available Move Types (continued)

±Registration Move (continued)

The Programmed Speed can be changed while the move is occurring under the following conditions:

- The value cannot be changed once a Controlled Stop condition occurs
- The value can only be decreased.
- When changing the Programmed Speed, the Acceleration and Deceleration values must be the same as they were when the original command was sent to the module.

All other parameter values are fixed when the command is first issued.

±Blend Move

This command allows you to create more complicated move profiles consisting of two to sixteen pre-programmed segments. These segments are programmed using positive relative values. A +Blend Move results in CW output pulses, a –Blend Move result in CCW output pulses. The figure below shows a three segment Blend Move that is run twice. It is first run in the CW direction, (left to right), the second is in the CCW direction, (right to left).

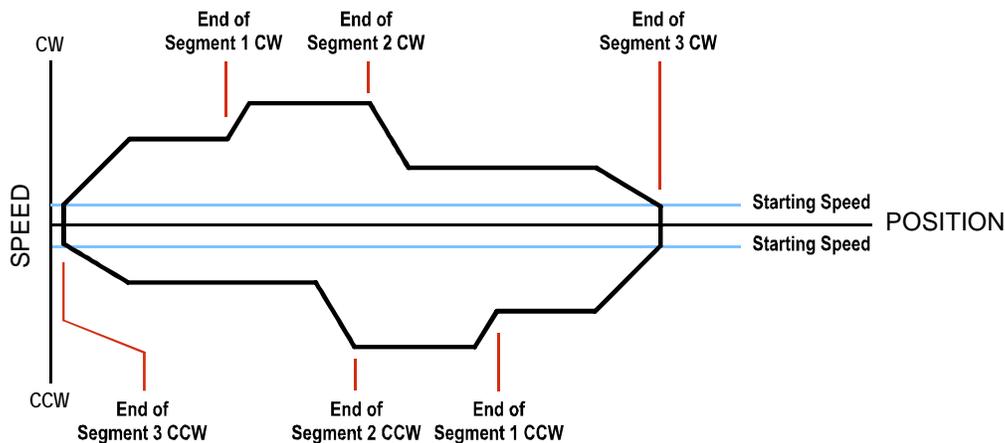


Figure 2.11 Blend Move Profile

Each segment is programmed with a programming block sent from the PLC using the Output Image Table words assigned to the axis. The first programming block specifies the total number of segments in the move as well as the acceleration types used for each segment and the final deceleration value. You need one additional programming block for each segment that programs the Segment Length, the Programmed Speed for the segment, and the Acceleration/Deceleration value used to reach the Programmed Speed for the segment.

The minimum amount of changing information needed to define a new move segment is the Programmed Speed. The Segment Length and the Acceleration/Deceleration rate do not have to change from one segment to the next. An Invalid Profile Error will be generated if the Programmed Speed parameter is not different in two consecutive blend move segments. An Invalid Profile Error will also be issued if the segment length does not contain enough steps to reach the Programmed Speed based on the programmed Acceleration Type and Value.

Blend move segment programming is done before the move is begun, with all of the segments of the blend move profile stored in the internal memory of the 3102i. This data will remain in the module's memory until power is removed from the module, the configuration data is re-programmed, or a new blend move profile is programmed for the axis. Once stored, multiple Blend Move commands can be run on the data stored in memory. You do not have to re-program the Blend Move segment before every move.

When a ±Blend Move command is issued, the first segment starts at the configured Starting Speed and accelerates to the specified Programmed Speed. The starting speed for the next segment is equal to the Programmed Speed of the current segment. The final segment will decelerate from its Programmed Speed to the Starting Speed and then stop. It is not possible to program a direction reversal in the Blend Move profile.

Available Move Types (continued)

±Blend Move (continued)

Blend Moves cannot be brought into a Hold State. The External Input is ignored during a Blend Move and issuing a Hold command will only result in the *Command Error* bit being set. The move profile will continue until it is complete, at which time the error bit will be reset.

±Encoder Follower Moves

An encoder is typically used in an open-loop system for position verification. The encoder is mounted to the back of the controlled motor and the encoder position is read after a move to verify that the motor moved as expected.

When an axis of the 3102i is used in Encoder Follower mode, the encoder is not mounted on the controlled motor. Instead, the encoder is typically mounted on a second motor, but it can be mounted anywhere, including on something as simple as a hand crank. While in this mode, the axis does not accept move commands over the backplane. Instead, the axis will output motor control pulses in response to pulses on the encoder inputs.

Two commands are available. The +Encoder Follower Move will output clockwise pulses when the encoder count is increasing and the -Encoder Follower Move will output counter-clockwise pulses when the encoder count is increasing.

The 3102i increases the encoder count register when the encoder's A signal is leading the B signal.

The External Input is ignored when the axis is in Encoder Follower mode.

This mode is also known as Electronic Gearing, because the motor behaves as if it is mechanically geared to the encoder. The 3102i has Multiplier and Divisor parameters that allow you to adjust the ratio between the encoder and the motor. The Multiplier and Divisor parameter each have a range of 1 to 32,768. It is possible to enter a ratio greater than one when programming these values. The 3102i will output multiple steps per encoder count.

As with all encoder functions on the 3102i, X4 decoding is used when changing the encoder count register. This fact must be taken into account when calculating the appropriate Multiplier and Divisor values.

Note that the maximum encoder input frequency is 250KHz, and the maximum motor output frequency is 1 MHz.

It is possible for your application to mathematically exceed this 1 MHz maximum output frequency when the Multiplier/Divisor ratio is greater than one. For example, assume your Multiplier/Divisor ratio is 3, and your encoder input frequency is 125 KHz. Ideally, your motor output frequency would be:

$$125\text{KHz} \times 4\{\text{decoding}\} \times 3\{\text{ratio}\} = 1.5\text{MHz}$$

As this exceeds the 1 MHz limit, the 3102i will output pulses at its 1MHz rate until motor position catches up with the encoder position.

The Linear Acceleration type is the only one available when using Encoder Follower Moves. Unlike other moves, the Acceleration and Deceleration values can be set to zero with this move type. If they are zero, the axis will immediately match the encoder speed when a move begins. If this stalls the motor, set the Acceleration and Deceleration parameters to their maximum value of 2000 steps/millisecond/second. If a move still causes the motor to stall, then decrease the Acceleration and Deceleration values until the motor runs correctly.

NOTE

- 1) The 3102i will always output the correct number of steps when operating in this mode. However, the use of the Acceleration and Deceleration parameters, especially low values, can cause the motor's motion to lag behind the encoder's motion.
- 2) If the Acceleration and Deceleration parameters are set to zero, electrical noise on the encoder inputs may cause the motor to oscillate back and forth by ±1 step when there is no encoder motion.



Notes

This chapter introduces the three interpolated move types available with the 3102i. These moves coordinate the outputs of the two channels, allowing you to perform synchronized moves.

The X-Y Plane

Its easiest way to explain the interpolated move types is in terms of motion in the X-Y plane defined by the two axes. This plane is shown in figure 3.1.

- A move that results in increasing counts on the axis will cause CW pulses on the outputs of the axis. Likewise, a move that results in decreasing counts on the axis will cause CCW pulses on the outputs of the axis.
- The ± 2 billion+ counts at the ends of each axis represent the minimum and maximum values of the Current Position register for the axis, which is a signed 32 bit value. *These values are not hard limits.* If you are performing a relative move in a CW direction that exceeds 2,147,483,647 counts, the Current Position value will roll over to its maximum negative value.
- The colored square with limits of $\pm 1,073,741,823$ counts represents the limits of Absolute coordinates. All defined positions must be within these limits when programming an absolute move, but, in the case of circular interpolated moves, the move can travel beyond these limits while it is running.

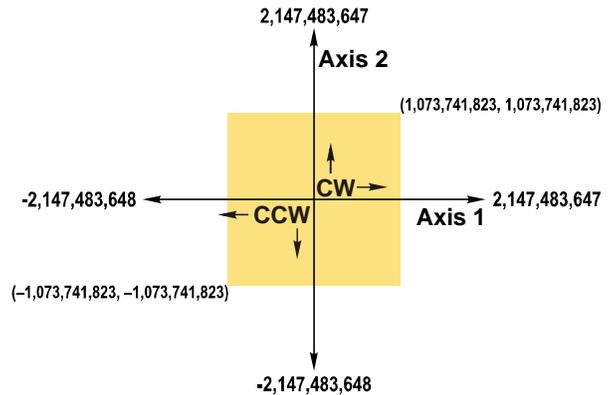


Figure 3.1 X-Y Plane

Linear Interpolated Moves

Conceptually, a Linear Interpolated Move is performed by travelling the shortest distance between two points on the X-Y plane defined by the two axes of the 3102i.

The Start Point is the current position defined by the two axes. The End Point can be programmed with Relative or Absolute coordinates. In figure 3.2 on the right, relative coordinates are shown in parentheses.

Relative Linear Move

When programming a move with Relative coordinates, you program the number of steps, or offset, you want each axis to travel.

- When programming the End Point with Relative coordinates, the maximum offset is $\pm 1,073,741,823$ counts. Positive offsets will result in clockwise moves on the axis, while negative offsets result in counter-clockwise moves.
- The Current Motor Position register that is reported back to the PLC is a full 32 bit register. ($\pm 2,147,483,647$ counts) The only way to move beyond $\pm 1,073,741,823$ counts is with multiple relative moves.

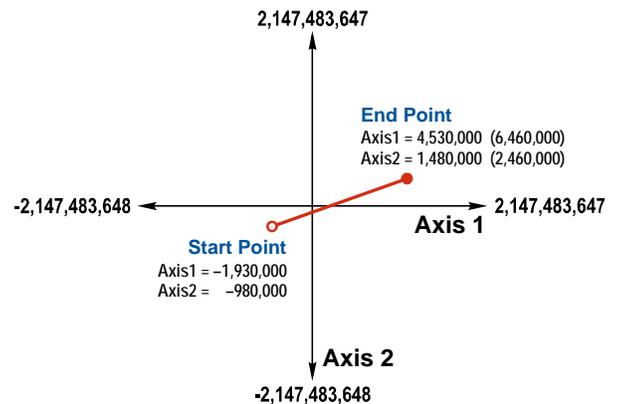


Figure 3.2 Linear Interpolated Move

Linear Interpolated Moves (continued)

Absolute Linear Move

Absolute coordinates treat the End Point as an actual position on the machine. Note that you must set the Home Position for both axes of the machine before you can run an Absolute Linear Move. (See *Definition of Home Position* on page 16.)

- The range of values for the End Point when it is treated as an actual position on the machine is $\pm 1,073,741,823$ counts. The move will be clockwise if the End Position coordinate is greater than the Current Position of the axis and negative if the End Position coordinate is less than the Current Position of the axis.
- The Start Point can be any position in the X-Y plane. It does not have to be within the $\pm 1,073,741,823$ count range.
- The Current Motor Position register that is reported back to the PLC is a full 32 bit register. ($\pm 2,147,483,647$ counts) However, you cannot move beyond $\pm 1,073,741,823$ counts with an Absolute Linear Move.

Required Parameters

Five parameters are required to define a Linear Interpolated Move:

- End Point X (Axis1) coordinate (Absolute or Relative)
- End Point Y (Axis2) coordinate (Absolute or Relative)
Must use same system as the X coordinate
- Interpolated Target Speed
- Interpolated Acceleration
- Interpolated Deceleration

Target Speed, Acceleration, and Deceleration define the vector for the path as a whole, not the individual axes, so they are programmed only once. As shown in figure 3.3 to the right, the 3102i calculates the required component vector values for each axis before beginning the move.

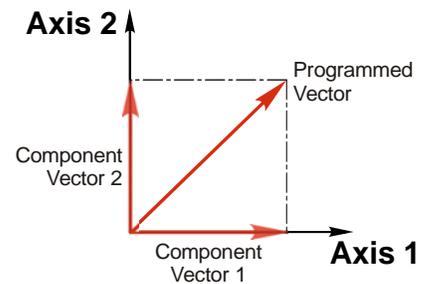


Figure 3.3 Programmed Vectors

Circular Interpolated Moves

Conceptually, a Circular Interpolated Move is performed by traveling between two points in the X-Y plane along an arc of a circle defined within the plane. As with Linear Interpolated Moves, the Start Point of every move is the current position of the two axes. The other points can be programmed with Absolute or Relative coordinates.

For non-interpolated, single axis moves, the terms “clockwise” and “counter-clockwise” refer to whether or not the Motor Position value reported back to the PLC will increase or decrease as a result of the move. (Clockwise moves increase the Motor Position Value.) For Circular Interpolated Moves, the terms “clockwise” and “counter-clockwise” refer to the direction of travel when looking down at the X-Y plane. For example, figure 3.4 below shows a counter-clockwise move.

There are three methods to specify a Circular Interpolated Move.

Center Point Method

The Center Point method is shown in figure 3.4. In addition to the End Point, this method defines the Center Point of the circle. The 3102i verifies that the length from the Center Point to the Start Point is equal to the length from the Center Point to the End Point before running the profile. *Both lengths are radii of the circle and must be equal.*

Even though these three points completely define the circle, they cannot define the direction of travel along the circle to move from the Start Point to the End Point. Because of this, the 3102i has two commands for use with the Center Point method. One command causes CW motion along the circle path while the other causes CCW motion. Note that CW and CCW motion in this case refers to the direction of travel in the X-Y plane as shown in the figure above. It does not refer to the state of the outputs during the move. Depending on the size and location of the circular path, each axis may output both CW and CCW pulses during the move.

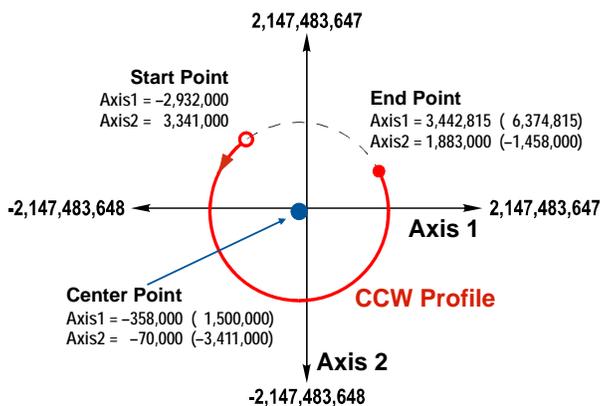


Figure 3.4 Center Point Circular Move

The Center Point and End Point can be specified with Relative or Absolute coordinates and the two points must use the same coordinate system.

- When programming the End Point and Center Point with Relative coordinates, the maximum offsets are $\pm 1,073,741,823$ counts from the Start Point.
- When programming the End Point and Center Point with Absolute coordinates, the positions must be in the range of $\pm 1,073,741,823$ counts. The Start Point can be any position in the X-Y plane. It does not have to be within the $\pm 1,073,741,823$ count range.
- The 3602 calculates the radius value and it cannot exceed 2,147,483,647 counts.

NOTE This is the only method that allows you to set the End Point equal to the Start Point and travel along the entire circular path in the X-Y plane. All other methods only allow you to travel over an arc of the defined circle.

Circular Interpolated Moves (continued)

Radius Method

The Radius method is shown in figure 3.5. In addition to the End Point, this method defines radius of the circular path. These three pieces of information actually define two circles in the plane, so an additional piece of information is used to define the move path. This piece of additional information is the sign of the radius value.

- ▶ If the radius is positive, the move will travel the shortest arc between the two points. If the radius is negative, the moves will travel the longest arc between the two points.

In order to determine the direction of travel, clockwise or counter-clockwise, the 3102i has two commands for use with the Radius method. One command causes CW motion along the circle path while the other causes CCW motion. Note that CW and CCW motion in this case refers to the direction of travel in the X-Y plane as shown in the figure 3.5. It does not refer to the state of the outputs during the move.

The bottom half of figure 3.5 shows the four available moves based on the sign of the radius value and the move direction.

- ▶ **Path A:** CW move, Negative radius
- ▶ **Path B:** CW move, Positive radius
- ▶ **Path C:** CCW move, Positive radius
- ▶ **Path D:** CCW move, Negative radius

The End Point can be specified with Relative or Absolute coordinates.

- ▶ When programming the End Point with Relative coordinates, the maximum X and Y offsets are $\pm 1,073,741,823$ counts from the Start Point.
- ▶ When programming the End Point with Absolute coordinates, the X and Y positions must be in the range of $\pm 1,073,741,823$ counts. The Start Point can be any position in the X-Y plane. It does not have to be within the $\pm 1,073,741,823$ count range.
- ▶ The End Point cannot have the same coordinates as the Start Point. Therefore, a move programmed by this method cannot travel along the entire circular path in the X-Y plane. If you need your move to travel the entire circular path, then you must program the move using the *Center Point Method* as described on page 29.
- ▶ The maximum value for the Radius parameter is $\pm 2,147,483,647$ counts.

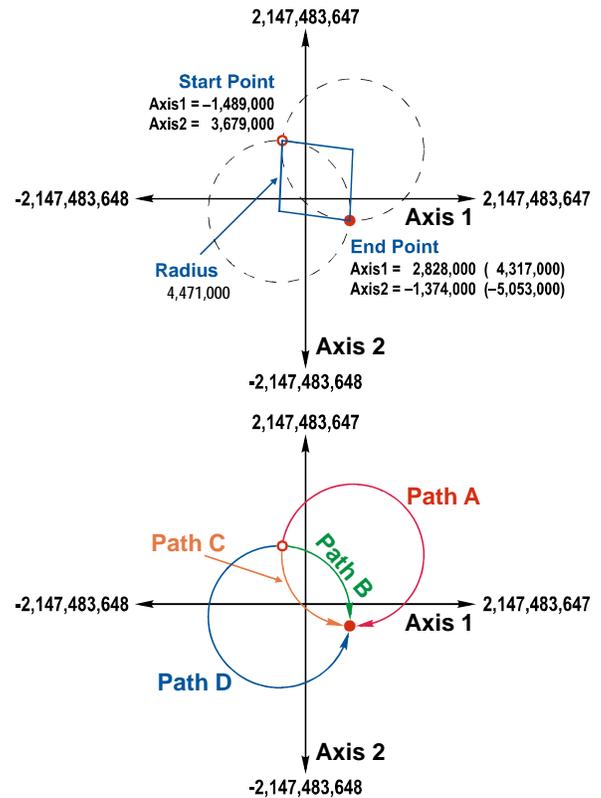


Figure 3.5 Radius Circular Move

Circular Interpolated Moves (continued)

Via Point Method



This method is only available on units made after May of 2010. Please refer to *Applicable Modules* found on page 7 for information on determining when your 3102i was manufactured.

The Via Point Method defines a third point on the circular path that the move will pass through while travelling from the Start Point to the End Point. This method is illustrated in figure 3.6.

This method has only one command associated with it because the 3102i can determine the direction of travel for the move with the three given points.

The Via Point and End Point can be specified with Absolute or Relative coordinates and the two points must use the same coordinate system.

- When programming the End Point and Via Point with Relative coordinates, the maximum X and Y offsets are $\pm 1,073,741,823$ counts from the Start Point.
- When programming the End Point and Via Point with Absolute coordinates, the X and Y positions must be in the range of $\pm 1,073,741,823$ counts. The Start Point can be any position in the X-Y plane. It does not have to be within the $\pm 1,073,741,823$ count range.
- The maximum value for the Radius parameter is 2,147,483,647 counts.
- Note that the Start Point, End Point, and Via Point must all have different coordinates and they cannot be collinear. Therefore, a move programmed by this method cannot travel along the entire circular path in the X-Y plane. If you need your move to travel the entire circular path, then you must program the move using the *Center Point Method* as described on page 29.

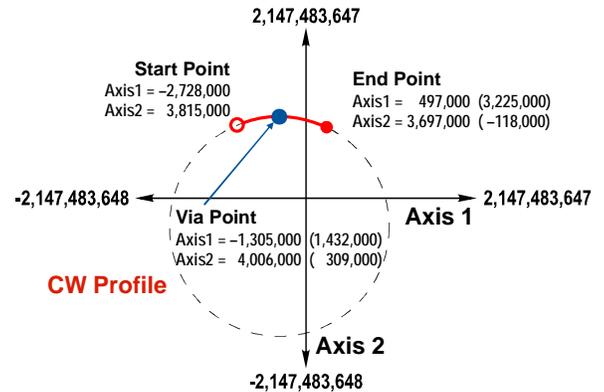


Figure 3.6 Via Point Circular Move

Controlled and Immediate Stops

Like most of the single axis moves described in chapter 2, Interpolated Moves can be brought to a Controlled Stop or Immediate Stop.

- **Controlled Stop:** The move immediately begins decelerating at its programmed deceleration value until it reaches the configured Starting Speed. The move stops at this point. The motor position values are still considered valid after a Controlled Stop and the machine does not need to be homed before Absolute Moves can be run again.
- **Immediate Stop:** The move immediately stops outputting pulses regardless of the speed the motors are running at. Because it is possible for the inertia of the load attached to the motors to pull them beyond their stopping points, the motor position values are considered invalid after an Immediate Stop and the machine must be homed before Absolute Moves can be run again.

Backplane Control

Hold Move Command: This command can be used to bring the interpolated move to a Controlled Stop. This command should be issued on axis 1.

Immediate Stop Command: When this command is issued from the PLC, the Interpolated Move will come to an Immediate Stop. The move cannot be restarted and the machine must be homed before Absolute Moves can be run again. This command should be issued on axis 1.

Controlled and Immediate Stops (continued)

Hardware Control

External Input: The External Input can be used to bring an Interpolated Move to a Controlled Stop. There is an External Input on both axes. Either input will bring the Interpolated Move to a Controlled Stop.

CW Limit and CCW Limit Inputs: Activating any of these four inputs during an Interpolated Move will bring the move to an Immediate Stop.

Emergency Stop Input: It is possible to configure the CW Limit Input and/or the CCW Limit Input as an Emergency Stop Input. When an Emergency Stop Input is activated, the Interpolated Move will come to an Immediate Stop.

Welcome to the Integer World

When you enter coordinates for Interpolated Moves, you enter integer values. The 3102i then calculates the path it needs to take before it actually starts the move. For Linear Interpolated Moves, the 3102i will always be able to calculate a move profile as long as the end point is within its valid range. This is not the case for Circular Interpolated Moves.

Regardless of the method you use to define a Circular Interpolated Move, the 3102i always calculates the center point of the circle as well as the radius from the center point to the start point and the radius from the center point to the end point. If these radii are not equal, then the 3102i generates an error message to the PLC instead of performing the move.

More likely than not, the two radii will be real values with fractional components. These two values are rounded off by the 3102i to integers before being compared. The 3102i will round off these values as you expect, all values with a fractional component greater than or equal to 0.5 are rounded up, all values with a fractional component less than 0.5 are rounded down.

Figure 3.7 is an example of this problem. The radius from the Center Point and Start Point is approximately 1,828.6, which is rounded to the integer 1,829 by the 3102i. The radius from the Center Point and End Point is approximately 1,828.2, which is rounded to the integer 1,828. Because these radii differ by one count, the 3102i will refuse to run the move.

You have two options to correct this problem. You can adjust the Center Point location or you can adjust the End Point location. Which solution you choose depends on which point location is more important for you to keep. In the example, the End Point was adjusted by increasing the axis 1 position by one count. This is the simplest solution because adjusting the Center Point which requires you to recalculate the Center Point to Start Point radius because it will also change.

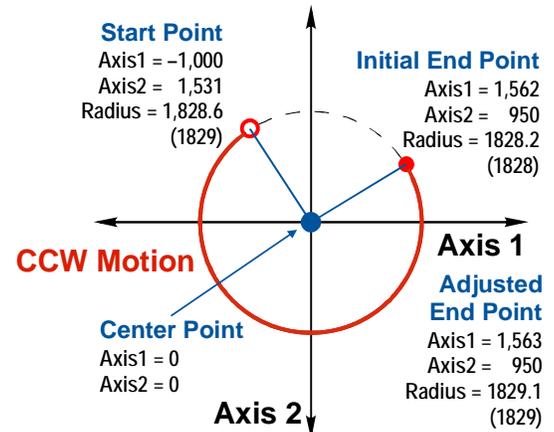


Figure 3.7 Round Off Error Example

This chapter explains the various ways of homing an axis on the 3102i. Inputs used to home the module are introduced and diagrams that show how the module responds to a homing command are given.

Definition of Home Position

The Home Position is any position on your machine that you can sense and stop at. Once at the Home Position, the motor position register on the 3102i must be set to an appropriate value. If you use the module's \pm Find Home commands, the motor position register will automatically be set to zero once the home position is reached. The Encoder Position register will also be reset to zero if the quadrature encoder is enabled for the axis.



Defining a Home Position is completely optional. Some applications, such as those that use a servo or stepper for speed control, don't require position data at all.

With the exception of Absolute Moves, the 3102i can still perform all of its move commands if the Home Position is not defined. Both axes must be homed before you can run Absolute Interpolated Moves.

Position Preset

One of the ways to define the Home Position is to issue the Preset Position command to the 3102i axis. Before doing this, you will need a way of sensing position outside the 3102i module. One possibility is by using an AMCI 1141 or 1142 Resolver Interface module for the MicroLogix PLC to sense the absolute position of the machine. Another possibility is an AMCI 7161 or 7162 SSI Interface module for the MicroLogix system and an SSI sensor. In either case, the position data must be brought into the PLC, the correct preset value calculated, and this value written to the 3102i axis with the Position Preset command.

\pm Find Home Commands

The other choice is to use the module's Find Home commands to order the 3102i to find the Home Position based on sensors brought into the unit. The +Find Home command begins searching by outputting CW pulses to the motor's driver and ends when the home sensor triggers while the 3102i is outputting CW pulses at a low rate. The -Find Home command operates in the same way but starts and ends with CCW pulses.

Homing Inputs

Five inputs can be used when homing the module. These inputs are either physical inputs attached to the module or bits in the PLC output data words.

Physical Inputs

- **Home Input:** This input is used in one of two ways: 1) This input is used to define the actual home position of the machine. 2) The input is used as a home proximity input when using the encoder marker pulse to home the machine.
- **Encoder Marker (Z) Pulse:** If you configure the 3102i to use an encoder, you have the option of using the encoder's marker pulse to home the 3102i.
- **CW Limit Switch Input:** This input is used to prevent overtravel in the clockwise direction.
- **CCW Limit Switch Input:** This input is used to prevent overtravel in the counter-clockwise direction.

Backplane Inputs

- **Home Proximity Bit:** The 3102i can be configured to ignore changes on the physical homing input until the Home Proximity Bit makes a 0→1 transition. The 3102i will home on the next inactive-to-active change on the physical input once this transition occurs. You must program your PLC to control the state of this bit.

Homing Configurations

The 3102i axis must be correctly configured before one of the homing commands will be accepted. One of the following must be part of the axis configuration before you can run the homing commands.

- 1) Home Input Enabled with Home Operation To Home Input
- 2) Home Input and Backplane Home Proximity Operation Enabled with Home Operation To Home Input
- 3) Home Input and Encoder Enabled with Home Operation to Marker Pulse. (In this configuration, the Home Input acts as a hardware home proximity input.)
- 4) Encoder and Backplane Home Proximity Operation Enabled with Home Operation To Marker Pulse

NOTE ⚠ A homing operation can occur without configuring the axis to use the CW Limit or CCW Limit inputs. If you choose to configure the axis in this way, the axis has no way to automatically prevent overtravel during a homing operation. You must prevent overtravel by some external means, or ensure that the homing command is issued in the direction that will result in reaching the homing input directly.

In addition to the above settings, the Homing Timeout parameter will also be set. This parameter sets a maximum number of seconds the 3102i axis will attempt to find the Home Position before it fails with an error. The range is 0 to 300 seconds, with a value of zero disabling the timeout feature.

Homing Profiles

NOTE ⚠ The +Find Home command is used in all of these examples. The -Find Home command will generate the same profiles in the opposite direction.

Home Input Only Profile

Figure 4.1 below shows the move profile generated by a +Find Home command when you use the Home Input without the Backplane Home Proximity bit.

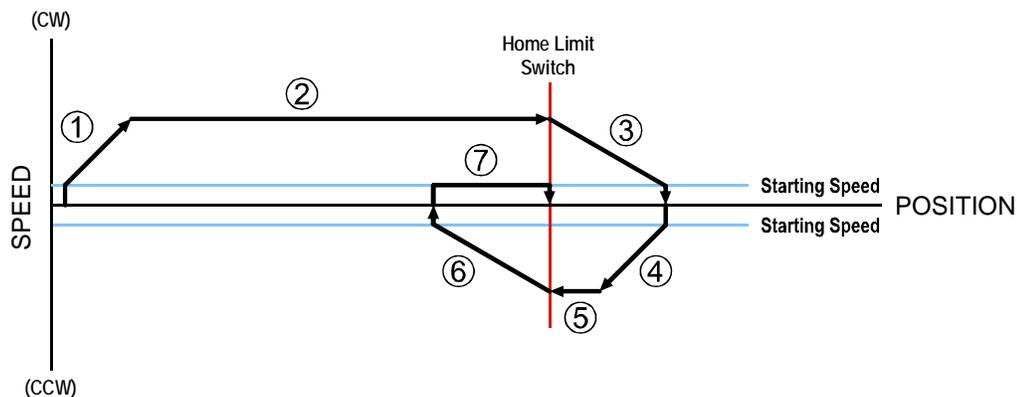


Figure 4.1 Home Input Profile

- 1) Acceleration from the configured Starting Speed to the Programmed Speed
- 2) Run at the Programmed Speed until the Home Input activates
- 3) Deceleration to the Starting Speed and stop, followed by a two second delay.
- 4) Acceleration to the Programmed Speed opposite to the requested direction.
- 5) Run opposite the requested direction until the Home Input transitions from Active to Inactive
- 6) Deceleration to the Starting Speed and stop, followed by a two second delay.
- 7) Return to the Home Input at the configured Starting Speed. Stop when the Home Input transitions from inactive to active.

NOTE ⚠ If the Home Input is active when the command is issued, the move profile begins at step 5 above.

Homing Profiles (continued)

Profile with Proximity Input

Figure 4.2 below shows the move profile generated by a +Find Home command when you use:

- Home Input with Backplane Home Proximity bit
- Marker Pulse home with Home Input as proximity sensor
- Marker Pulse home with Backplane Home Proximity bit

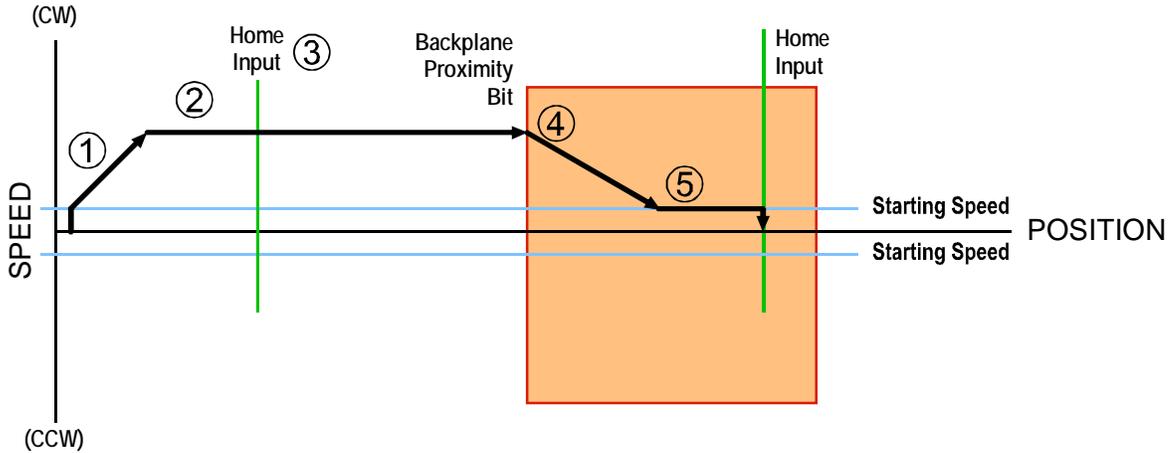


Figure 4.2 Homing with Proximity

- 1) Acceleration from the configured Starting Speed to the Programmed Speed
- 2) Run at the Programmed Speed
- 3) Ignores homing input, (Home Input or Marker Pulse) because proximity input has not made an inactive to active transition.
- 4) Deceleration towards the Starting Speed when the proximity input (Backplane bit or Home Input) transitions from its inactive to active state. The axis will stop as soon as the Home Input becomes active.
- 5) The Starting Speed is the minimum speed the profile will run at. If the axis decelerates to the Starting Speed before reaching the Home Input, it will continue at this speed.

NOTE

Figure 4.2 shows the Backplane Home Proximity Bit staying active until the 3602 reaches its home position. This is valid, but does not have to occur. As stated in step 4, the 3602 starts to hunt for the home position as soon and the Backplane Home Proximity Bit makes a 0→1 transition

Homing Profiles (continued)

Profile with Overtravel Limit

Figure 4.3 below shows the move profile generated by a +Find Home command when you use:

- CW Overtravel Limit
- Home Input without Backplane Home Proximity Bit

The profile is generated when you encounter an overtravel limit in the direction of travel. (In this example, hitting the CW limit while traveling in the CW direction.) Hitting the overtravel limit associated with travel in the opposite direction is an Immediate Stop condition. The axis will stop all motion and issue a *Home Invalid* error to the PLC.

The 3102i will stop the axis with an error if both overtravel limits are activated while the 3102i is trying to find the home position.

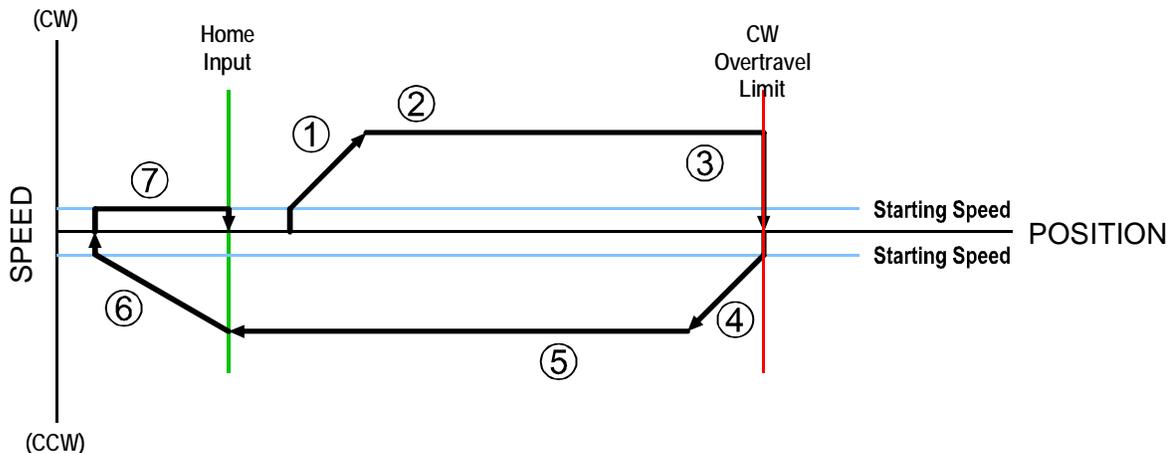


Figure 4.3 Profile with Overtravel Limit

- 1) Acceleration from the configured Starting Speed to the Programmed Speed
- 2) Run at the Programmed Speed
- 3) Hit CW Limit and immediately stop, followed by a two second delay.
- 4) Acceleration to the Programmed Speed opposite to the requested direction.
- 5) Run opposite the requested direction until the Home Input transitions from Active to Inactive
- 6) Deceleration to the Starting Speed and stop, followed by a two second delay.
- 7) Return to the Home Input at the configured Starting Speed. Stop when the Home Input transitions from inactive to active.



If the overtravel limit is active when the Find Home Command is active, the profile will begin at step 4.

INSTALLING THE 3102i

This chapter contains information on properly handling the 3102i as well as mounting the module in a safe environment. This chapter also covers how to wire your field I/O to the 3102i.

Location

1762 I/O, including the 3102i, is suitable for use in an industrial environment that meet the following criteria:

- ▶ Only non-conductive pollutants normally exist in the environment, but an occasional temporary conductivity caused by condensation is expected.
- ▶ Transient voltages are controlled and do not exceed the impulse voltage capability of the product's insulation.

Note that these criteria apply to the system as a whole, not just to the 3102i.

These criteria are equivalent to the *Pollution Degree 2* and *Over Voltage Category II* designations of the International Electrotechnical Commission (IEC).

Safe Handling Guidelines

Prevent Electrostatic Damage



Electrostatic discharge can damage the 3102i if you touch the bus connector pins. Follow these guidelines when handling the module.

- 1) Touch a grounded object to discharge static potential before handling the module.
- 2) Work in a static-safe environment whenever possible.
- 3) Wear an approved wrist-strap grounding device.
- 4) Do not touch the pins of the bus connector or I/O connector.
- 5) Do not disassemble the module
- 6) Store the module in its anti-static bag and shipping box when it is not in use.

Prevent Debris From Entering the Module



During panel or DIN rail mounting of all devices, be sure that all debris (metal chips, wire strands, tapping liquids, etc.) is prevented from falling into the module. Debris may cause damage to the module or unintended machine operation with possible personal injury.

Remove Power Before Servicing



The 1762 I/O bus is not hot swap capable, so remove power before removing or installing any module. If you remove or install a module with power applied, an electric arc may occur. In addition to causing excessive wear on the contact of the connectors, this electrical arc can cause personal injury or property damage by:

- 1) sending a faulty signal to your system's field devices, causing unintended machine operation
- 2) causing an explosion in a hazardous environment
- 3) causing permanent damage to the module that may result in its immediate failure or shorten its service life.

Mounting

Minimum Spacing

As shown in figure 5.1, you must maintain a minimum spacing of 2 inches (50.8 millimeters) from enclosure walls, wireways, adjacent equipment, etc. for adequate system ventilation.

Also note that all 1762 expansion I/O must be mounted in the orientation shown in the figure. Mounting the system in any other orientation will decrease the efficiency of the ventilation slots on the top and bottom of each module which may lead to system overheating and malfunction.

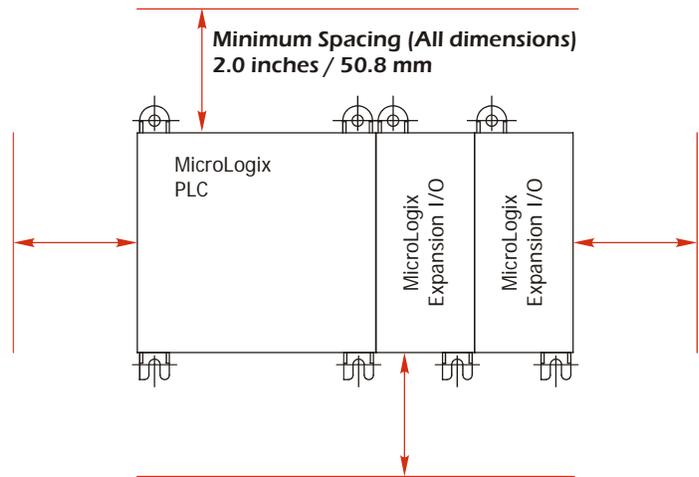


Figure 5.1 Ventilation Spacing

Panel Mounting

Panel mounting is the required mounting method for environments where the system will be exposed to high shock and/or vibration values. Use figure 4.2 to design a template to mount your system. Note that dimensions for each MicroLogix PLC are specific to each model number so you must refer to the PLC documentation to determine those dimensions.

Two mounting screws are required for every module. The preferred hardware is M4 or #8 panhead screws. M3.5 or #6 panhead screws can also be used, but you may have to install a washer to ensure good ground contact.

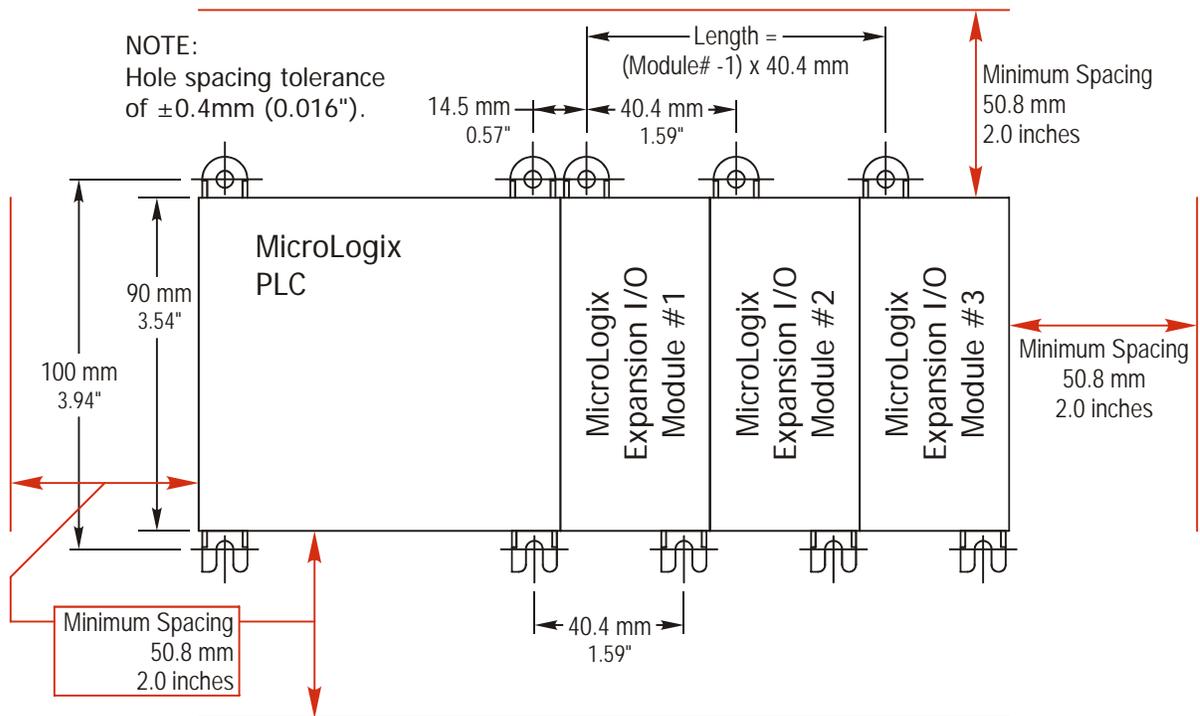


Figure 5.2 Panel Mounting Dimensions

Mounting (continued)

DIN Rail Mounting

The module can be mounted on following DIN Rails

- EN 05 022 - 35 x 7.5 (35 x 7.5 mm)
- EN 05 022 - 35 x 15 (35 x 15 mm)

Close the DIN rail latch on the module before mounting the module on the rail. Press the DIN rail mounting area of the module against the rail. The latch will open as you install the module and will snap into place when the module is properly seated on the DIN rail.

If the system will be exposed to limited shocks of vibrations then use Allen-Bradley DIN rail end anchors to secure the system as shown in figure 5.3. The Allen-Bradley part numbers for these anchors are 1492-EA35 and 1492-EAH35.

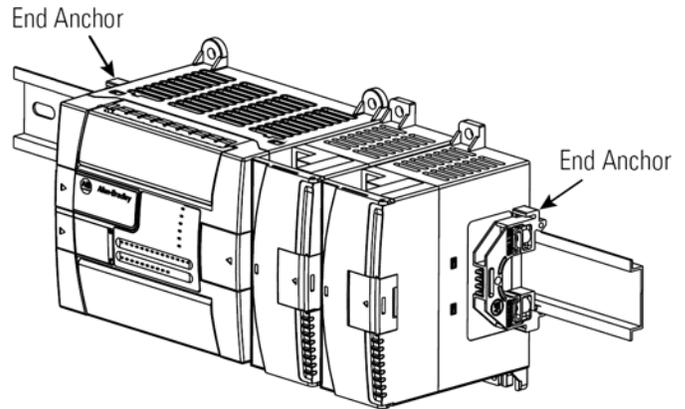


Figure 5.3 DIN Rail Mounting

NOTE ⚠ If your system will be exposed to large amounts of shock or vibration, you should panel mount your system instead of mounting it to a DIN rail. Instructions for panel mounting are given on the previous page.

System Assembly

The communications bus is carried between the modules through the ribbon cable attached to every module. Once the system is properly mounted, you can connect the I/O modules together and to the PLC as shown in figure 5.4.

- NOTE** ⚠
- 1) Use the pull loop on the connector to disconnect modules. Do not pull on the ribbon cable as this may damage the module.
 - 2) The bus connector covers must be re-installed after the bus connections are made.

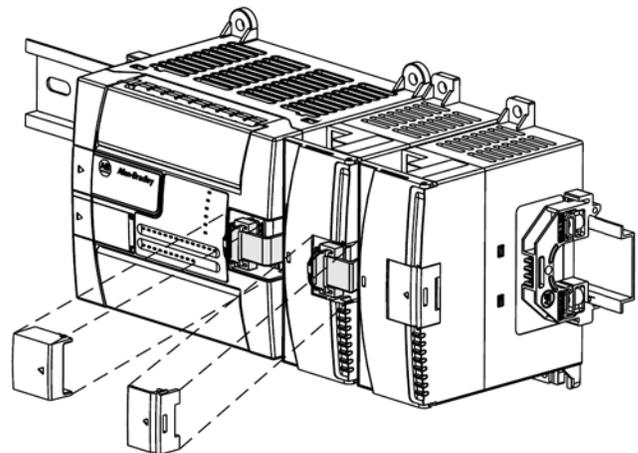
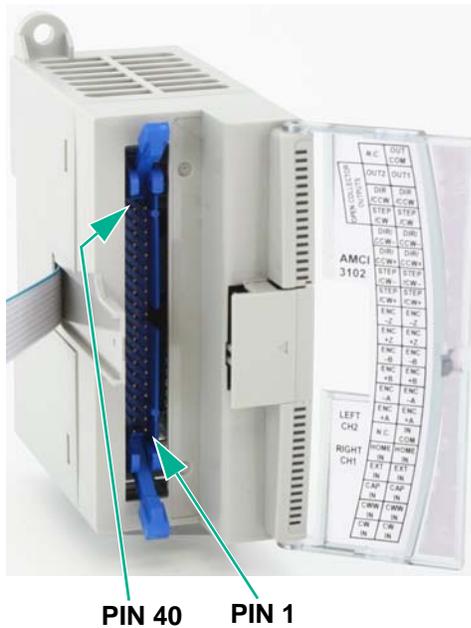


Figure 5.4 Bus Connectors

Terminal Block Pinout

The following table lists the pinout of the 40 pin high density connector on the 3102i.



| Pin # | Function | Pin # | Function |
|-------|---------------------|-------|---------------------|
| 40 | No Connection | 39 | Output Common |
| 38 | Output 2 | 37 | Output 1 |
| 36 | CH2 Dir/CCW_OC | 35 | CH1 Dir/CCW_OC |
| 34 | CH2 Step/CW_OC | 33 | CH1 Step/CW_OC |
| 32 | CH2 -Dir/CCW | 31 | CH1 -Dir/CCW |
| 30 | CH2 +Dir/CCW | 29 | CH1 +Dir/CCW |
| 28 | CH2 -Step/CW | 27 | CH1 -Step/CW |
| 26 | CH2 +Step/CW | 25 | CH1 +Step/CW |
| 24 | CH2 -Z Encoder | 23 | CH1 -Z Encoder |
| 22 | CH2 +Z Encoder | 21 | CH1 +Z Encoder |
| 20 | CH2 -B Encoder | 19 | CH1 -B Encoder |
| 18 | CH2 +B Encoder | 17 | CH1 +B Encoder |
| 16 | CH2 -A Encoder | 15 | CH1 -A Encoder |
| 14 | CH2 +A Encoder | 13 | CH1 +A Encoder |
| 12 | No Connection | 11 | Input Common |
| 10 | CH2 Home Input | 9 | CH1 Home Input |
| 8 | CH2 External Input | 7 | CH1 External Input |
| 6 | CH2 Capture Input | 5 | CH1 Capture Input |
| 4* | CH2 CCW Limit Input | 3* | CH1 CCW Limit Input |
| 2* | CH2 CW Limit Input | 1* | CH1 CW Limit Input |

Table 5.1 3102i Connector Pinout

* By default, these inputs are CW and CCW Limit switch (overtravel) inputs. Any, and even all, of them can be configured to be Emergency Stop inputs.

MS-40 Connector Kit

The MS-40 connector kit contains a 40 pin mating connector for the 3102i along with forty individual crimp pins. This kit allows you to make your own custom cable. It is equivalent to the Allen-Bradley part number 1746-N3.

IM-3R Interface Module and Cable

AMCI offers a breakout box to simplify field wiring to the 3102i. The IM Interface Module is designed to be DIN rail mounted and can be attached to either of the rails that the MicroLogix system can be mounted on. The kit includes a three foot round ribbon cable to connect to the 3102i module. Dimensions for the IM module are given below.

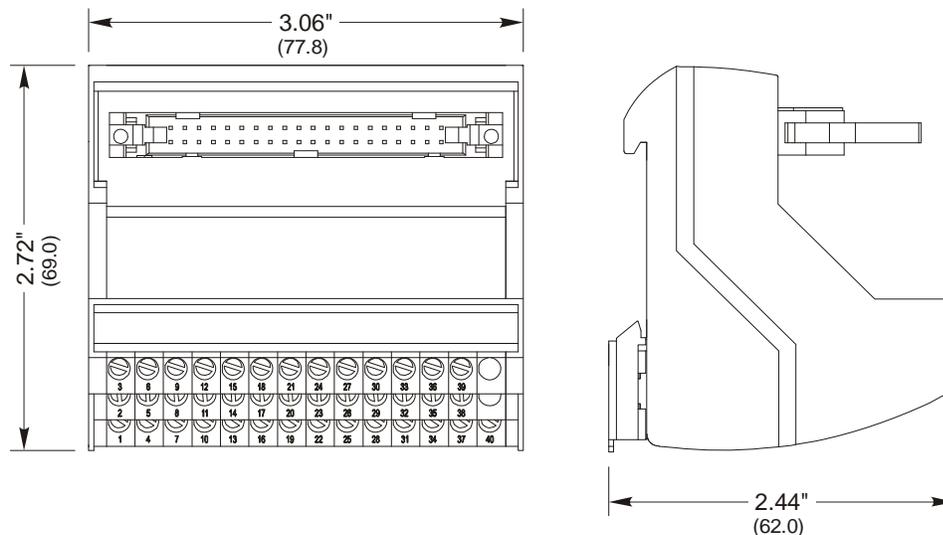


Figure 5.5 IM Interface Module Dimensions

General Wiring Guidelines

Allen-Bradley has a reference document, *Industrial Automation Wiring and Grounding Guidelines* that should be followed when installing the 3102i. The document's catalog number is 1770-4.1 and it is available online from the www.ab.com website.

When using the guidelines, all I/O wiring for the 3102i should be treated as Category 2 wiring (Signal and Communication). If you are using the IM-3R Interface Module from AMCI, the rounded ribbon cable that connects the IM to the 3102i should be treated as Category 3 wiring (Intra-enclosure).

Driver Wiring

Connections to your driver can be made through the module's differential outputs or the single-ended open collector sinking outputs. You cannot use both output types at the same time.

Differential Wiring

Differential outputs are low voltage, low current outputs that require shielded, twisted pair cable to carry the signals. A suggested cable is the Belden 9729, which supports a 1 MHz pulse frequency to over 1000 feet of cable length, but any quality, shielded, twisted pair cable should suffice. A simple wiring diagram is shown below. Note that the cable shields should be grounded at the module end only.

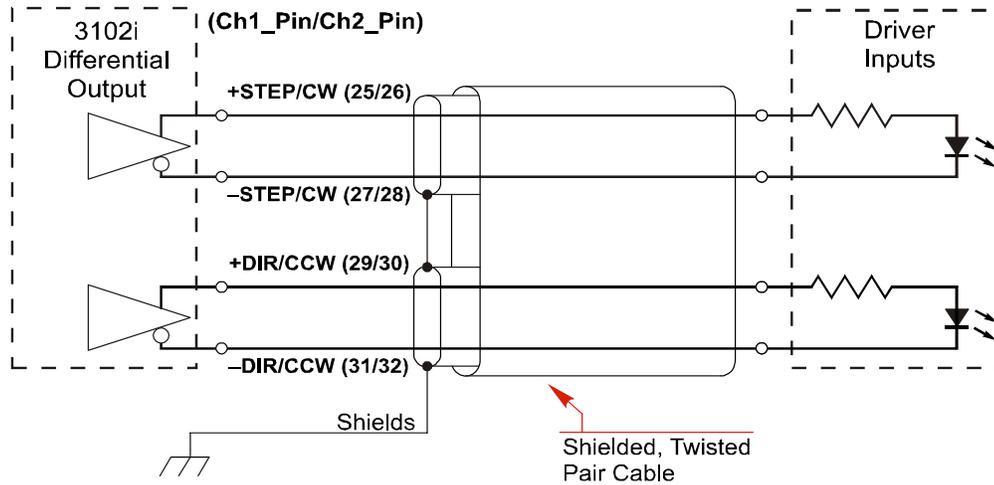


Figure 5.6 Differential Driver Wiring

Open Collector Wiring

All open collector outputs in the 3102i share a single common pin that must be tied to the ground connection of the power supply powering the outputs. Therefore, all outputs are open collector sinking. Figure 5.7 below shows how to wire the outputs to open collector sourcing inputs on your driver.

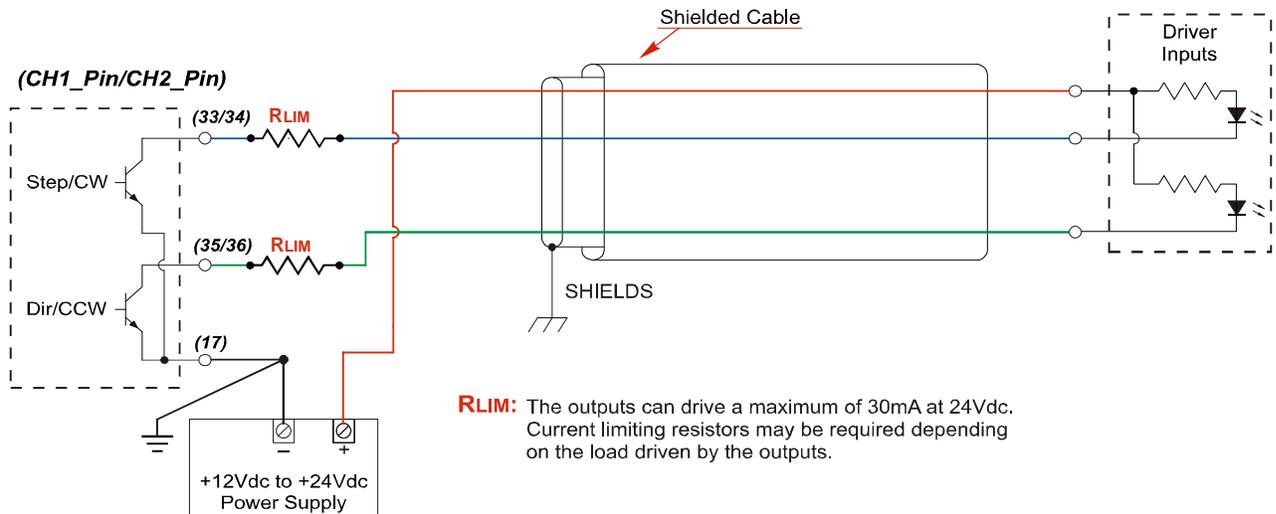


Figure 5.7 Open Collector Driver Wiring (Sourcing Inputs)

NOTE RLIM may be needed to limit current flow. The 3102i drivers can conduct a maximum of 30 milliamperes. The exact value of RLIM is based on the power supply voltage and the current requirements of the Driver Inputs.

Driver Wiring (continued)

Open Collector Wiring (continued)

It is possible to wire the 3102i outputs to open collector sinking inputs on a driver by using pull up resistors.

NOTE This configuration will effectively reverse the logic states of the outputs. A Driver Input will be normally on and turn off when the 3102i output is in a logic “1” state.

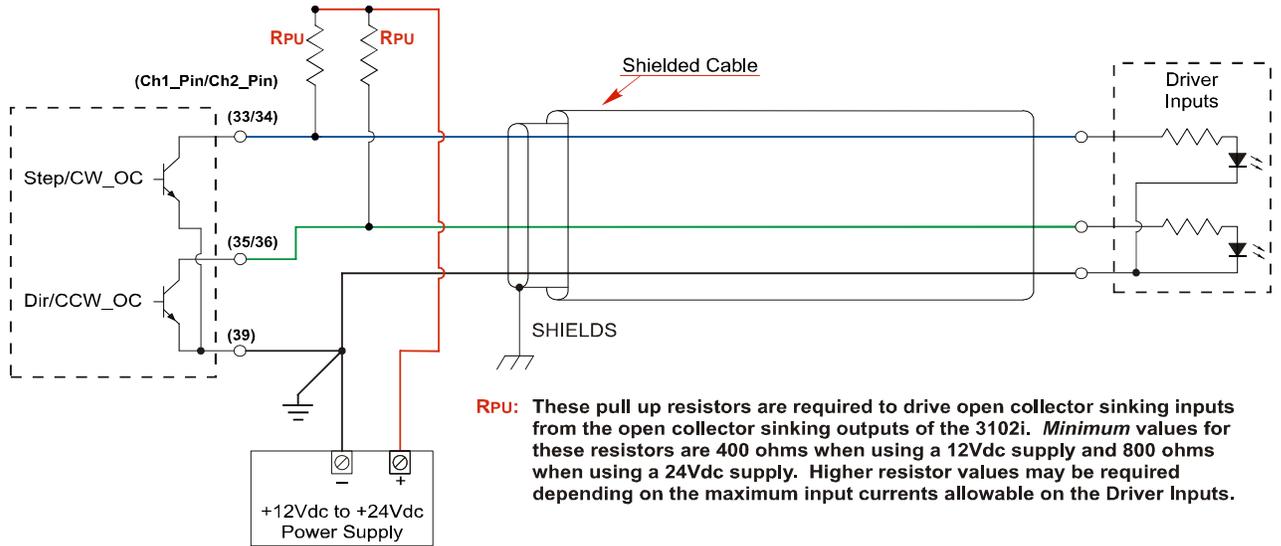


Figure 5.8 Open Collector Driver Wiring (Sinking Inputs)

CAUTION Carefully select the value of the pull up resistors. (RPU in the figure.) The 3102i outputs can conduct a maximum of 30 milliamps without damage, so the minimum resistor value is 400 ohms / 0.5 watt when using a 12 Vdc supply and 800 ohms / 1.0 watt when using a 24 Vdc supply. *You must increase the size of these resistors if the inputs to your Driver cannot conduct 30 milliamps without damage.*

General Purpose Output Wiring

The two General Purpose Outputs are electrically identical to the open collector Step/CW_OC and Dir/CCW_OC outputs available on each axis. The General Purpose Outputs shares a common ground pin, (pin 39) with these outputs. Wiring is identical to the diagrams given in figures 5.7 and 5.8.

NOTE The General Purpose Output can, but does not have to, use the power supply that powers the open collector Driver Outputs of the 3102i. If you use two different supplies, the low side of the supplies must be tied together at pin 39 of the 3102i. For this reason, the two supplies should be mounted close together and share the same Earth Ground bus to avoid ground loop issues.

Discrete Input Wiring

The discrete inputs are all open collector sinking type with a single common. They accept 8Vdc to 24Vdc signals. Figure 5.9 below shows how to wire an input to a typical three wire sourcing sensor.

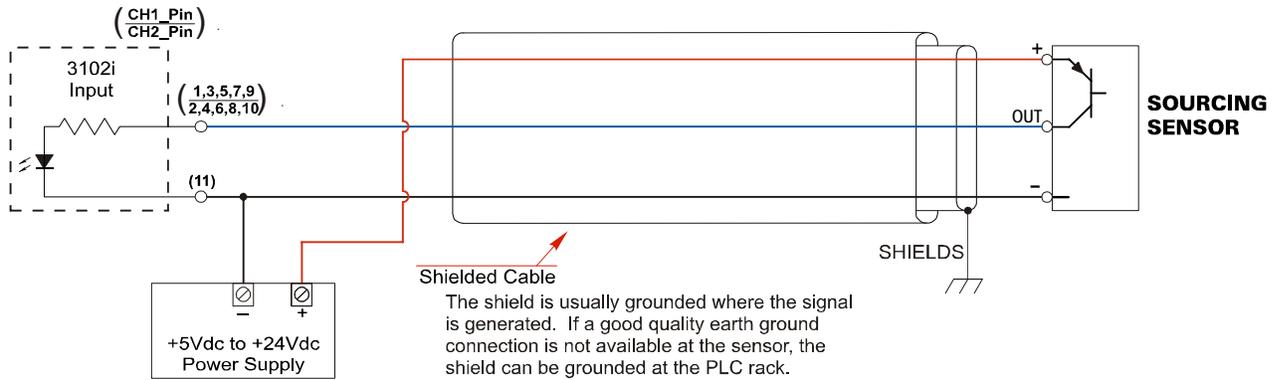


Figure 5.9 Discrete Input Wiring

Encoder Wiring

Differential Encoder Wiring

The encoder inputs on the 3102i accept 5Vdc differential signals from your incremental encoder. AMCI offers optical encoders as an option when ordering any of our dual-shaft stepper motors. Figure 5.10 below shows a typical encoder installation using the color codes of our optical encoders. Note that there is no single color coding standard in the encoder market so you must refer to your encoder's documentation if you are not using an AMCI optical encoder.

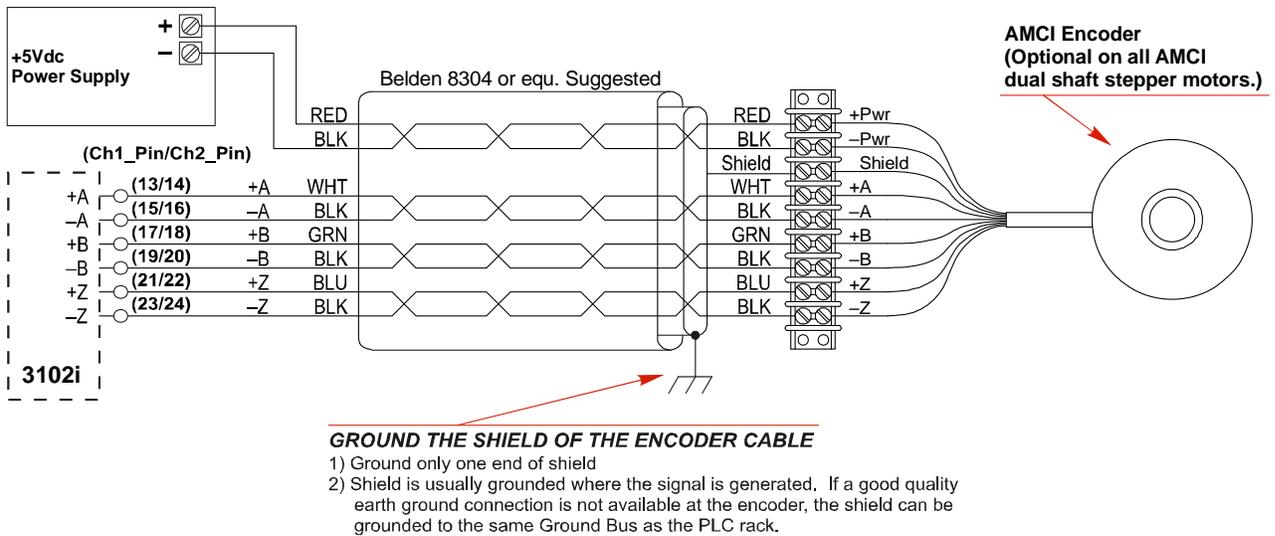


Figure 5.10 Differential Encoder Wiring

Encoder Wiring (continued)

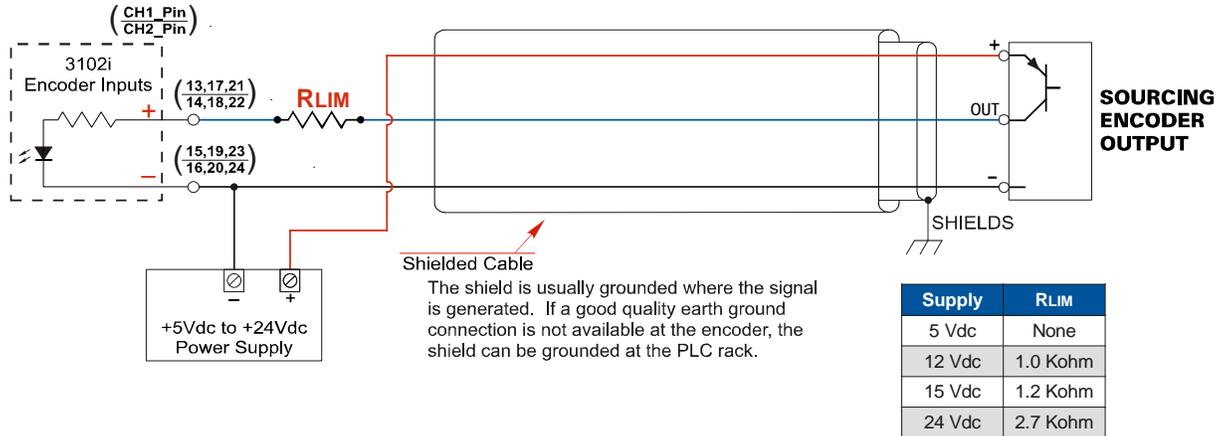
Single Ended Wiring

Figure 5.11 below shows how to wire the encoder inputs to both a single ended sourcing and single ended sinking encoder outputs.

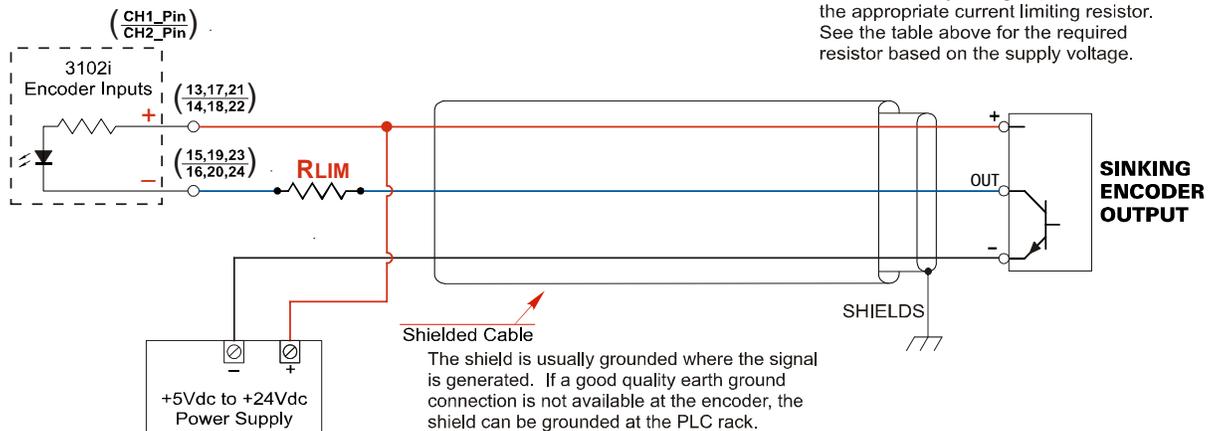


The encoder inputs on the 3102i are rated for 5Vdc only. You must use a current limiting resistor on each input if the outputs of your encoder are greater than 5Vdc. Appropriate current limiting resistors are shown in the figure below.

3102i Encoder Input Connection to Sourcing Encoder Output



3102i Encoder Input Connection to Sinking Encoder Output



RLIM: The inputs are designed to accept 5Vdc but can use any voltage up to 24Vdc with the appropriate current limiting resistor. See the table above for the required resistor based on the supply voltage.

Figure 5.11 Single Ended Encoder Wiring

Diagnostic Feedback Wiring

The 3102i can be placed in a Diagnostic Feedback Mode that allows you to check the functionality of the differential motor control outputs and the encoder A and B inputs. You must wire the outputs to the inputs as shown in the table below before you can use this mode.

| Axis # | Wire This Pin... | | ...To This Pin | |
|--------|------------------|-------|----------------|----------|
| | Function | Pin # | Pin # | Function |
| 1 | +Step | 25 | 13 | +A |
| 1 | -Step | 27 | 15 | -A |
| 1 | +Direction | 29 | 17 | +B |
| 1 | -Direction | 31 | 19 | -B |
| 2 | +Step | 26 | 14 | +A |
| 2 | -Step | 28 | 16 | -A |
| 2 | +Direction | 30 | 18 | +B |
| 2 | -Direction | 32 | 20 | -B |

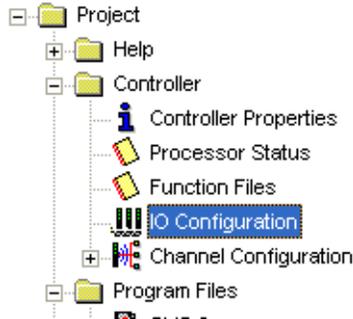
Table 5.2 Diagnostic Feedback Wiring

RSLogix 500 CONFIGURATION

The 3102i requires 16 input words and 16 output words in order to communicate with a MicroLogix controller. This chapter outlines how to add a 3102i module to you project and configure the I/O.

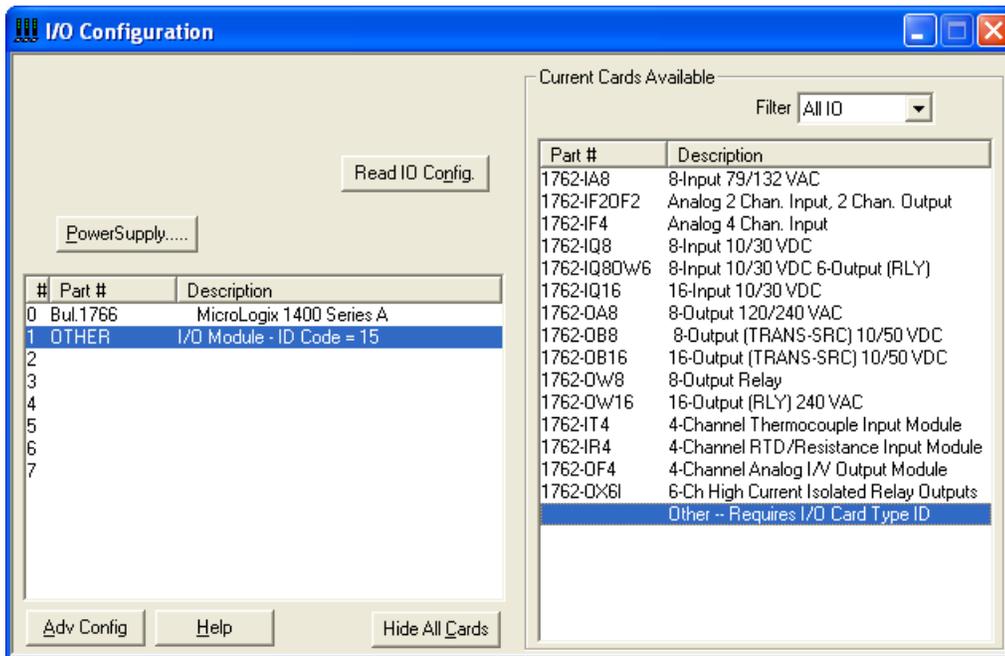
Add the 3102i to Your Project

- 1) Open or create the RSLogix 500 project in which you wish to use the 3102i.
- 2) Double click on the *I/O Configuration* leaf of the project tree...



The I/O Configuration window will open.

- 3) Select the slot where the 3102i will be installed.



In the *Current Cards Available* pane on the right, double click on *Other-Requires I/O Card Type ID*.

Add the 3102i to Your Project (continued)

- 4) Enter the module data in the I/O slot configuration window that appears. The correct data is shown in the figure below. When you are finished, click the [OK] button.

Module #1: OTHER - I/O Module - ID Code = 15

Expansion General Configuration

Vendor ID: 10

Product Type: 16

Product Code: 15

Series/Major Rev/Minor Rev: A

Input Words: 16

Output Words: 16

Extra Data Length: 0

Ignore Configuration Error:

OK Cancel Apply Help

- 5) Repeat steps 1 through 4 for each additional 3102i you wish to add to your project.

Finding the 3102i in Your Project

Each 3102i module will appear in the I/O Configuration section of your project tree with the part number of “OTHER” and a description of “I/O Module - ID Code = 15”. If you have multiple 3102i modules in your project, they will all have the same part number and description. You can have up to four modules in an 1100 project, up to three modules in a 1200 project, and up to five modules in a 1400 project.

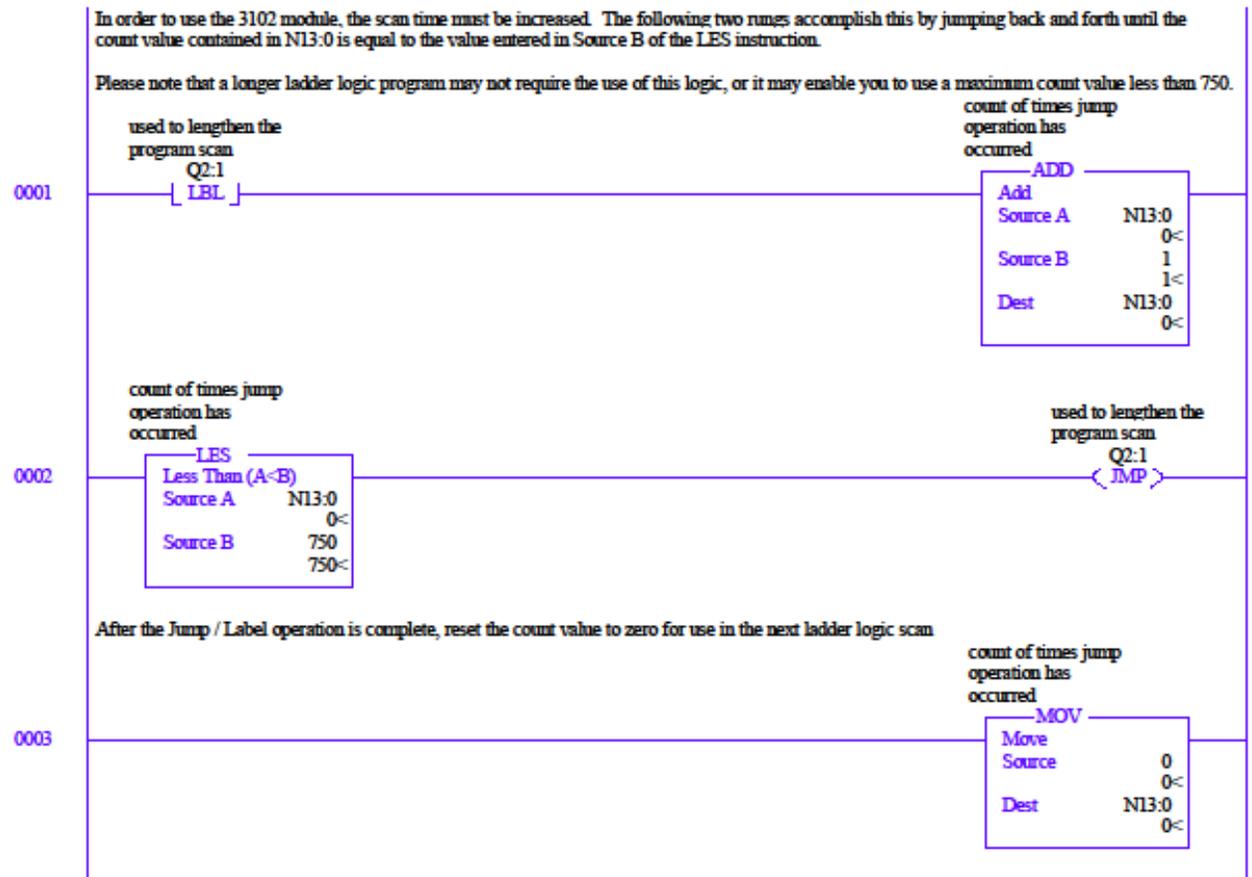
Data Location

Input Data (data from the 3102i to the PLC) will appear in the Input Image Table registers I:X.0 to I:X.15, where X is the module’s slot number.

Output Data (data from the PLC to the 3102i) must be written to the Output Image Table registers O:X.0 to O:X.15, where X is the module’s slot number.

PLC I/O Errors

Your PLC may suffer from I/O Errors if your PLC program is relatively short. This is caused by the fact that the 3102i uses 16 Input data words and 16 Output data words and short PLC programs can be scanned faster than the module I/O can be transferred. The following code, which is also available in the sample program, can be used to increase the PLC scan time. The sample program is available on our website, www.amci.com, under the "Tech Library" heading.



Notes

CONFIGURATION MODE DATA FORMAT

The 3102i has two modes of operation, Configuration Mode, and Command Mode. This chapter describes the format of data written to and read from the module when it is in Configuration Mode.

NOTE

The 3102i will always power up with both axes in Command Mode and report a configuration error. An axis must be configured before you can issue move commands to the axis. Note that the two axes are independent, and it is acceptable to have one axis in Configuration Mode while the other is in Command Mode. However, both axes must be properly configured and in Command Mode before you can issue Interpolated Move commands.

Output Data Format

The following table shows the format of the output image table words when writing configuration data to both axes of the 3102i.

| Word O:X._ | Axis | Function | Units | Range |
|---------------|------|----------------------------------|--------------|-----------------------|
| 0 | 1 | Configuration Bits MSW | | See Description Below |
| 1 | 1 | Configuration Bits LSW | | See Description Below |
| 2 | 1 | Starting Speed MSW | Steps/Second | 1 to 1,000,000 |
| 3 | 1 | Starting Speed LSW | | |
| 4 | 1 | Homing Timeout | Seconds | 0 to 300 |
| 5 | 1 | <i>Reserved, Must be 16#0000</i> | | <i>16#000</i> |
| 6 | 1 | <i>Reserved, Must be 16#0000</i> | | <i>16#000</i> |
| 7 | 1 | <i>Reserved, Must be 16#0000</i> | | <i>16#000</i> |
| 8 | 2 | Configuration Bits MSW | | See Description Below |
| 9 | 2 | Configuration Bits LSW | | See Description Below |
| 10 | 2 | Starting Speed MSW | Steps/Second | 1 to 1,000,000 |
| 11 | 2 | Starting Speed LSW | | |
| 12 | 2 | Homing Timeout | Seconds | 0 to 300 |
| 13 | 2 | <i>Reserved, Must be 16#0000</i> | | <i>16#000</i> |
| 14 | 2 | <i>Reserved, Must be 16#0000</i> | | <i>16#000</i> |
| 15 | 2 | <i>Reserved, Must be 16#0000</i> | | <i>16#000</i> |

Table 7.1 Output Data Format - Configuration Data



Output Data Format (continued)

Configuration Bits MSW

Word 0 configures Axis 1, Word 8 configures Axis 2

| To Make These Settings... | | ...Set These Bits To | | | | | | | | | | | | | | | |
|--|--------------------|----------------------|----|----------|----------|----|----|---|---|---|---|---|---|---|---|---|---|
| | | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Capture Input | Disabled | | | | | | | | | | | | | | | | 0 |
| | Enabled | | | | | | | | | | | | | | | | 1 |
| External Input | Disabled | | | | | | | | | | | | | | | 0 | |
| | Enabled | | | | | | | | | | | | | | | 1 | |
| Home Input | Disabled | | | | | | | | | | | | | | 0 | | |
| | Enabled | | | | | | | | | | | | | | 1 | | |
| CW Input | Not Used | | | | | | | | | | | 0 | 0 | | | | |
| | CW Limit Switch | | | | | | | | | | | 0 | 1 | | | | |
| | E-Stop Input | | | | | | | | | | | 1 | 0 | | | | |
| | Do Not Use | | | | | | | | | | | 1 | 1 | | | | |
| CCW Input | Not Used | | | | | | | | | | 0 | 0 | | | | | |
| | CCW Limit Switch | | | | | | | | | | 0 | 1 | | | | | |
| | E-Stop Input | | | | | | | | | | 1 | 0 | | | | | |
| | Do Not Use | | | | | | | | | | 1 | 1 | | | | | |
| Backplane Home Proximity Operation | Disabled | | | | | | | | | 0 | | | | | | | |
| | Enabled | | | | | | | | | 1 | | | | | | | |
| Quadrature Encoder | Will Not Be Used | | | | | | | | | | 0 | | | | | | |
| | Will Be Used | | | | | | | | | | 1 | | | | | | |
| Diagnostic Feedback | Will Not Be Used | | | | | | | | | | 0 | | | | | | |
| | Will Be Used | | | | | | | | | | 1 | | | | | | |
| Output Pulse Type | CW/CCW | | | | | | | | 0 | | | | | | | | |
| | Step & Direction | | | | | | | | 1 | | | | | | | | |
| Home Operation | To Limit Switch | | | | | | | 0 | | | | | | | | | |
| | To Marker Pulse | | | | | | | 1 | | | | | | | | | |
| Card Axis | To Enable | | 0 | | | | | | | | | | | | | | |
| | To Disable | | 1 | | | | | | | | | | | | | | |
| Operation Mode for Axis | Command Mode | 0 | | | | | | | | | | | | | | | |
| | Configuration Mode | 1 | | | | | | | | | | | | | | | |
| Write in Your Bit Pattern Here: | | | | 0 | 0 | | | | | | | | | | | | |

Table 7.2 Configuration Bits MSW



Do not set the Backplane Home Proximity Operation bit if you intend to only home to a Home Limit Switch. This bit should only be set if you need to use the Backplane Home Proximity bit to force the 3602 to ignore multiple triggers on the home input.



Output Data Format (continued)

Configuration Bits LSW

Word 1 configures Axis 1, Word 9 configures Axis 2

| To Make These Settings... | | ...Set These Bits To | | | | | | | | | | | | | | | |
|---|-------------------------------|----------------------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|---|---|---|---|---|
| | | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Capture Input Active State | Active Low [†] (NC) | | | | | | | | | | | | | | | | 0 |
| | Active High [‡] (NO) | | | | | | | | | | | | | | | | 1 |
| External Input Active State | Active Low (NC) | | | | | | | | | | | | | | | 0 | |
| | Active High (NO) | | | | | | | | | | | | | | | 1 | |
| Home Input Active State | Active Low (NC) | | | | | | | | | | | | | | 0 | | |
| | Active High (NO) | | | | | | | | | | | | | | 1 | | |
| CW (or E-Stop) Input Active State | Active Low (NC) | | | | | | | | | | | | 0 | | | | |
| | Active High (NO) | | | | | | | | | | | | 1 | | | | |
| CCW (or E-Stop) Input Active State | Active Low (NC) | | | | | | | | | | | 0 | | | | | |
| | Active High (NO) | | | | | | | | | | | 1 | | | | | |
| Write in Your Bit Pattern Here: | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | |
| Bits 5 through 15 are Reserved. They must always be set to zero. | | | | | | | | | | | | | | | | | |

- † Active Low inputs are considered active when they are not conducting current. (Voltage on the input is between 0Vdc and 2Vdc.)
- ‡ Active High inputs are considered active when they are conducting current. (Voltage on the input is between 8Vdc and 24Vdc.)

Table 7.3 Configuration Bits LSW

Starting Speed

The Starting Speed parameter sets the step frequency that all moves begin and end at. Portions of a homing operation are also performed at this speed. This parameter is used to avoid a low frequency resonance point when starting a move. This parameter is also useful in micro-stepping applications when it is not necessary to accelerate from a full stop. The range of the Starting Speed parameter is 1 to 1,000,000 steps per second. Because this value can be greater than 16 bits, this parameter is split into two 16 bit words when put into the Output Image Table. The following table shows how to split the Starting Speed parameter value. The 3102i sample program, which is available from the AMCI website, contains a sub-routine called *Separate* that shows how to perform this function in ladder logic.

| Desired Value (dec) | Desired Value (hex) | MSW (hex) | MSW (dec) | LSW (hex) | LSW (dec) |
|---------------------|---------------------|-----------|-----------|-----------|-----------|
| 999 | 16#0000:03E7 | 16#0000 | 0 | 16#03E7 | 999 |
| 50,800 | 16#0000:C670 | 16#0000 | 0 | 16#C670 | -14,736 |
| 94,784 | 16#0001:7240 | 16#0001 | 1 | 16#7240 | 29,248 |

Table 7.4 Starting Speed Sample Values

NOTE

- 1) As a suggestion, a Starting Speed that is equal to the square root of the acceleration rate will create a profile without any jumps in velocity at the beginning of the move profile. The same holds true for the end of the move profile if the Starting Speed is equal to the square root of the deceleration rate.
- 2) The Starting Speed can always be programmed in the range of 1 to 1,000,000 steps per second. This 1 MHz output frequency can only be reached by the differential outputs of the 3102i. The module's 24Vdc open collector outputs are limited to a maximum output frequency of 100,000 steps per second.



Output Data Format (continued)

Homing Timeout

The Homing Timeout parameter defines how long a homing operation will be allowed to continue before motion stops and the 3102i sets its *Home Invalid* status bit. (The *Home Invalid* status bit is available when the axis is in Command Mode.) The Homing Timeout parameter has a range of 0 to 300 seconds. Setting this parameter to a value of 0 disables this feature and the 3102i will search for the home marker until it finds it or the command is aborted by the PLC.

Input Data Format

While in Configuration Mode, the input registers assigned to the 3102i will echo back the last configuration data written to the axis. The only exceptions are bits 13 through 15 in the Configuration Bits MSW of each axis.

The following table shows the format of the input image table words when both axes are in Configuration Mode. Note that the two axes are completely independent, and it is acceptable to have one axis in Configuration Mode while the other is in Command Mode.

| Word I:X._ | Axis | Description |
|------------|------|---|
| 0 | 1 | Mirror of Configuration Bits MSW, (Bits 0 – 12)* |
| 1 | 1 | Mirror of Configuration Bits LSW |
| 2 | 1 | Mirror of Starting Speed MSW |
| 3 | 1 | Mirror of Starting Speed LSW |
| 4 | 1 | Mirror of Homing Timeout |
| 5 | 1 | <i>Not used in Configuration Mode, will always be 16#0000</i> |
| 6 | 1 | <i>Not used in Configuration Mode, will always be 16#0000</i> |
| 7 | 1 | <i>Not used in Configuration Mode, will always be 16#0000</i> |
| 8 | 2 | Mirror of Configuration Bits MSW, (Bits 0 – 12)* |
| 9 | 2 | Mirror of Configuration Bits LSW |
| 10 | 2 | Mirror of Starting Speed MSW |
| 11 | 2 | Mirror of Starting Speed LSW |
| 12 | 2 | Mirror of Homing Timeout |
| 13 | 2 | <i>Not used in Configuration Mode, will always be 16#0000</i> |
| 14 | 2 | <i>Not used in Configuration Mode, will always be 16#0000</i> |
| 15 | 2 | <i>Not used in Configuration Mode, will always be 16#0000</i> |

* Bits 13 - 15 are global status bits and are described on the following page.

Table 7.5 Input Data Format - Configuration Mode



Global Status Bits

Word 0 contains Status Bits for Axis 1, Word 8 contains Status Bits for Axis 2

The upper three bits of each mirror of the *Configuration Bits MSW* words are used to report the status of the axis. These bits have same meaning regardless of the mode the axis is in. While in Configuration mode, they do not echo back the last values of these bits written to the 3102i. The figure below shows meaning of each of these status bits.

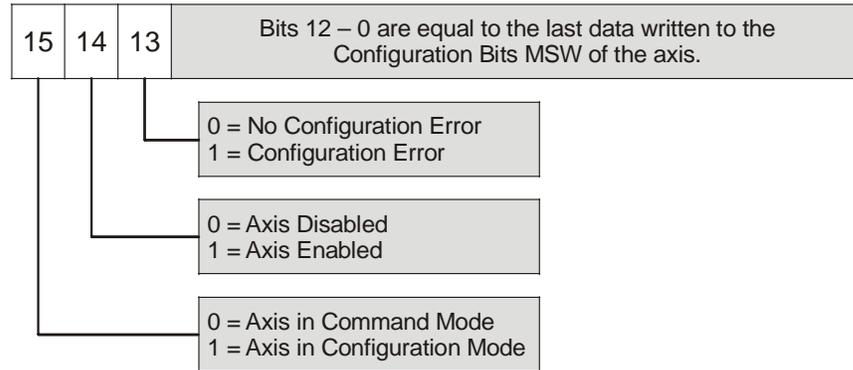


Figure 7.1 Global Status Bits

Invalid Configurations

The following section lists the configuration settings that cause the Configuration Error bit of the axis to be set when an invalid configuration is written to it. I:X.0/13 is the Configuration Error bit for axis 1 and I:X.8/13 is the Configuration Error bit for axis 2, where “X” is the slot number of the 3102i.

In the following tables, all bit places that do not show a value are “don’t care” terms for that error.

Configuration Bits MSW

Word 0 configures Axis 1, Word 8 configures Axis 2

| If these bits are set as shown.... | | | | | | | | | | | | | | | ...a Configuration Error will result because: | |
|---|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|--|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | | 0 |
| NOTE: This bit is always set when in Configuration Mode | | | 1 | 1 | | | | | | | | | | | | ANY of the reserved bits are set |
| | | | | | | | | | | | | 1 | 1 | | | Configuration tried to use an invalid setting for the CW/E-Stop Input. |
| | | | | | | | | | | 1 | 1 | | | | | Configuration tried to use an invalid setting for the CCW/E-Stop Input. |
| | | | | | | 0 | 1 | | | | | | | | | Configuration tried to enable Diagnostic Feedback while the Output Pulse Type was not set to Step/Dir. |
| | | | | | | | 1 | 1 | | | | | | | | Configuration tried to simultaneously enable the Quadrature Encoder and Diagnostic Feedback |
| | | | | | 1 | | | 0 | | | | | | | | Configuration tried to request a <i>Home to Marker Pulse</i> setting without enabling the Quadrature Encoder. |
| | | | | | 1 | | | 1 | 0 | | | | | 0 | | Configuration tried to request a <i>Home to Marker Pulse</i> setting without enabling the Home Input or Backplane Proximity bit. (Marker pulse homing requires one of the two for home proximity sensing.) |
| | | | | 1 | | | 1 | 1 | | | | | | 1 | Configuration tried to request a <i>Home to Marker Pulse</i> setting while enabling both the Home Input and the Backplane Proximity bit. (Marker pulse homing requires one and only one of the two for home proximity sensing.) | |

Table 7.6 Configuration Bits MSW Errors



Invalid Configurations (continued)

Configuration Bits LSW

Word 1 configures Axis 1, Word 9 configures Axis 2

| If these bits are set as shown.... | | | | | | | | | | | | | | ...a Configuration Error will result because: | | |
|------------------------------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|----------------------------------|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | | | | | | ANY of the reserved bits are set |

Table 7.7 Configuration Bits LSW Errors

Starting Speed Parameter

The Starting Speed parameter must not be outside its range of 1 to 1,000,000. This means that a value of zero *will* cause an error. Because the value can be greater than sixteen bits in length, you must consider the values of both the MSW and LSW. Combined, then cannot exceed a hexadecimal value of 16#000F:4240.

Homing Timeout Parameter

This parameter must be in the range of 0 to 300 decimal, 16#0000 to 16#012C in hexadecimal.

Reserved Words

The three reserved words in the configuration programming blocks for each axis must always be zero.

COMMAND MODE DATA FORMAT

Once properly configured, the 3102i can be placed in Command Mode. Moves can only be performed while in Command Mode. Output words are used to command the axis, while the input words are used to report back its status. As in Configuration Mode, the first eight of the sixteen I/O words are associated with axis 1, and the second eight words are associated with axis 2. Interpolated Move commands use all sixteen words.

Multi-Word Parameters

Most of the values used to program the 3102i, as well as values reported back to the PLC, can exceed $\pm 32,768$ counts. Therefore, they require two words of storage and they are transmitted as 32 bit values. The least significant word (LSW) contains the lower sixteen bits of the value and the most significant word (MSW) contains the remaining bits. When determining the values needed when programming the 3102i, it is easiest to convert the parameter value to hexadecimal and enter these values into your data table. (The MicroLogix always shows 16 bit values in 2's-compliment notation. Sixteen bit values between 32,768 and 65,535 will appear as negative numbers when viewed as decimals.

| Value (dec) | Value (hex) | MSW (hex) | MSW (dec) | LSW (hex) | LSW (dec) |
|-------------|--------------|-----------|-----------|-----------|-----------|
| 50,800 | 16#0000:C670 | 16#0000 | 0 | 16#C670 | -14,736 |
| 16,702,650 | 16#00FE:DCBA | 16#00FE | 254 | 16#DCBA | -9030 |
| -16,702,650 | 16#FF01:2346 | 16#FF01 | -255 | 16#2346 | 9,030 |

Table 8.1 Multi-word Data Format

Output Data Format

The following tables shows the format of the output image table words when writing command data to both axes of the 3102i. The left table shows command data when writing single-axis commands. The table on the right shows command data when writing Interpolated Commands.

| Word O:X_ | Axis | Standard Move Functions | Word O:X_ | Interpolated Move Functions |
|-----------|------|---|-----------|---|
| 0 | 1 | Command Bits MSW | 0 | Command Bits MSW |
| 1 | 1 | Command Bits LSW | 1 | Command Bits LSW |
| 2 | 1 | Command Parameters <i>Word meaning depends on the command set to the 3102i</i> | 2 | Interpolated Speed |
| 3 | 1 | | 3 | |
| 4 | 1 | | 4 | Interpolated Acceleration |
| 5 | 1 | | 5 | Interpolated Deceleration |
| 6 | 1 | | 6 | EndPointX Value (Axis1) |
| 7 | 1 | 7 | | |
| 8 | 2 | Configuration Bits MSW | 8 | <i>Reserved, Must be 16#0000</i> |
| 9 | 2 | Configuration Bits LSW | 9 | <i>Reserved, Must be 16#0000</i> |
| 10 | 2 | Command Parameters <i>Word meaning depends on the command set to the 3102i</i> | 10 | EndPointY Value (Axis2) |
| 11 | 2 | | 11 | |
| 12 | 2 | | 12 | <i>Word meaning depends on the command set to the 3102i</i> |
| 13 | 2 | | 13 | |
| 14 | 2 | | 14 | |
| 15 | 2 | 15 | | |

Table 8.2 Output Data Formats - Command Data

Output Data Format (continued)**Command Bits MSW****Word 0 configures Axis 1, Word 8 configures Axis 2**

The Command Bits MSW is used to control the motion of the axis. Consequently, only one bit in the word can be on at a time. AMCI strongly suggests that you use MOV instructions when setting this word instead of bit instructions to set and reset the individual bits.

| MSW value (hex) | # of bit set | Description |
|-----------------|--------------|---|
| 16#0000 | None | This word must equal zero when performing any Interpolated Move. |
| 16#0001 | 0 | Perform an ABSOLUTE MOVE |
| 16#0002 | 1 | Perform an RELATIVE MOVE |
| 16#0004 | 2 | Perform a HOLD MOVE operation |
| 16#0008 | 3 | Perform a RESUME MOVE operation |
| 16#0010 | 4 | Perform an IMMEDIATE STOP operation |
| 16#0020 | 5 | Perform a +FIND HOME (CW) operation |
| 16#0040 | 6 | Perform a -FIND HOME (CCW) operation |
| 16#0080 | 7 | Perform one of the following based on Command Parameter data: 1) +JOG MOVE (CW) 2) +REGISTRATION MOVE (CW) 3) +ENCODER FOLLOWER MOVE |
| 16#0100 | 8 | Perform one of the following based on Command Parameter data: 1) -JOG MOVE (CCW) 2) -REGISTRATION MOVE (CCW) 3) -ENCODER FOLLOWER MOVE |
| 16#0200 | 9 | PRESET the current position. This command can also be used to reset the <i>Move Complete</i> status bit. |
| 16#0400 | 10 | PRESET ENCODER current position |
| 16#0800 | 11 | RESET COMMAND ERRORS |
| 16#1000 | 12 | Run +BLEND MOVE PROFILE (CW) |
| 16#2000 | 13 | Run -BLEND MOVE PROFILE (CCW) |

Table 8.3 Command Bits MSW

The remaining three bits of this word are bits 10, 14 and 15.

- **Bit 10: Reserved:** This bit must always be zero.
- **Bit 14: Reserved:** This bit must always be zero.
- **Bit 15: MODE CONTROL Bit:** A '0' places the 3102i in Command Mode.
A '1' places the 3102i in Configuration Mode.

COMMAND MODE DATA FORMAT

Output Data Format (continued)

Command Bits LSW

Word 1 configures Axis 1, Word 9 configures Axis 2

| To Make These Settings... | | ...Set These Bits To | | | | | | | | | | | | | | | |
|--|----------------------|----------------------|----|----|----|----|-----|---|---|---|---|---|---|---|---|---|---|
| | | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Acceleration Type | Constant | | | | | | | | | | | | | | | 0 | 0 |
| | Constant | | | | | | | | | | | | | | | 1 | 0 |
| | Triangular S | | | | | | | | | | | | | | | 0 | 1 |
| | Trapezoidal S | | | | | | | | | | | | | | | 1 | 1 |
| Registration Move Distance | To leave value as is | | | | | | | | | | | | | 0 | | | |
| | To program value | | | | | | | | | | | | | 1 | | | |
| Encoder Follower Mode | Disabled | | | | | | | | | | | | 0 | | | | |
| | Enabled | | | | | | | | | | | | 1 | | | | |
| Program Blend Move | Disabled | | | | | | | | | | | 0 | | | | | |
| | Enabled | | | | | | | | | | | 1 | | | | | |
| Blend Move Transmit | Accept New Block | | | | | | | | | | | | | | | | |
| | Acknowledge OK'd | | | | | | | | | | | | | | | | |
| Backplane Home Proximity | Not Near Home | | | | | | | | | | | | | | | | |
| | Near Home | | | | | | | | | | | | | | | | |
| G.P. Output State | OFF | | | | | | | | | | | | | | | | |
| | ON | | | | | | | | | | | | | | | | |
| Run any command in Command Bits MSW | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | | | |
| Run a Relative Interpolated Move | | | | | | | | 0 | | | | | | | | | |
| Run an Absolute Interpolated Move | | | | | | | | 1 | | | | | | | | | |
| Run a Linear Interpolated Move | | 0 | 0 | 0 | 0 | 1 | 0/1 | | | | | | | | | | |
| Run a Via Point Circular Interpolated Move | | 0 | 0 | 0 | 1 | 0 | 0/1 | | | | | | | | | | |
| Run a CW Center Point Circular Int. Move | | 0 | 0 | 1 | 0 | 0 | 0/1 | | | | | | | | | | |
| Run a CCW Center Point Circular Int. Move | | 0 | 1 | 0 | 0 | 0 | 0/1 | | | | | | | | | | |
| Run a CW Radius Circular Int. Move | | 1 | 0 | 1 | 0 | 0 | 0/1 | | | | | | | | | | |
| Run a CCW Radius Circular Int. Move | | 1 | 1 | 0 | 0 | 0 | 0/1 | | | | | | | | | | |

Table 8.4 Command Bits LSW

Command Bits Must Transition

Commands, including Interpolated Move commands, are only accepted when the command bit makes a 0→1 transition.

The easiest way to make sure the command bits are zero before writing a new command is by using MOV instructions to clear the *Command Bits MSW* and *Command Bits LSW* words before writing the next command. The Command Bits MSW word can always have a value of zero written into it. The Command Bits LSW will need a value of zero or 128, (16#0080) depending on the state of the General Purpose Output State bit.

Refer to the 3102i sample program on the www.amci.com website for an example.

Single Axis Command Blocks

The following section lists the output data format for the fourteen different commands.

Absolute Move

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-------------------------|--------------|---|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0001 |
| 1 | 9 | <i>Command Bits LSW</i> | | See pg. 59 |
| 2 | 10 | Target Position MSW | Steps | $\pm 1,073,741,823$ (16#C000:001 to 16#3FFF:FFF) |
| 3 | 11 | Target Position LSW | | |
| 4 | 12 | Programmed Speed MSW | Steps/Second | Configured Starting Speed to 1,000,000 (16#000F:4240) |
| 5 | 13 | Programmed Speed LSW | | |
| 6 | 14 | Acceleration | Steps/ms/sec | 1 to 2000 |
| 7 | 15 | Deceleration | Steps/ms/sec | 1 to 2000 |

Table 8.5 Absolute Move Command Block

Relative Move

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-------------------------|--------------|---|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0002 |
| 1 | 9 | <i>Command Bits LSW</i> | | See pg. 59 |
| 2 | 10 | Relative Position MSW | Steps | $\pm 1,073,741,823$ (16#C000:001 to 16#3FFF:FFF) |
| 3 | 11 | Relative Position LSW | | |
| 4 | 12 | Programmed Speed MSW | Steps/Second | Configured Starting Speed to 1,000,000 (16#000F:4240) |
| 5 | 13 | Programmed Speed LSW | | |
| 6 | 14 | Acceleration | Steps/ms/sec | 1 to 2000 |
| 7 | 15 | Deceleration | Steps/ms/sec | 1 to 2000 |

Table 8.6 Relative Move Command Block

Single Axis Command Blocks (continued)

Hold Move

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-------------------------|-------|---------------------------|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0004 |
| 1 | 9 | <i>Command Bits LSW</i> | | See pg. 59 and Note below |
| 2 | 10 | Unused | | 0 (See Note below) |
| 3 | 11 | Unused | | 0 (See Note below) |
| 4 | 12 | Unused | | 0 (See Note below) |
| 5 | 13 | Unused | | 0 (See Note below) |
| 6 | 14 | Unused | | 0 (See Note below) |
| 7 | 15 | Unused | | 0 (See Note below) |

Table 8.7 Hold Move Command Block

Unused words are ignored by the 3102i and can be any value, including parameter values from the previous Absolute or Relative Move command. The Command Bits LSW word can also be left at its previous value or the word can be set to 16#0000 to have the General Purpose Output off, or 16#0080 to have the General Purpose Output on.

Resume Move

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-------------------------|--------------|---|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0008 |
| 1 | 9 | <i>Command Bits LSW</i> | | See pg. 59 |
| 2 | 10 | Unused | | 0 (See Note below) |
| 3 | 11 | Unused | | 0 (See Note below) |
| 4 | 12 | Programmed Speed MSW | Steps/Second | Configured Starting Speed to 1,000,000 (16#000F:4240) |
| 5 | 13 | Programmed Speed LSW | | |
| 6 | 14 | Acceleration | Steps/ms/sec | 1 to 2000 |
| 7 | 15 | Deceleration | Steps/ms/sec | 1 to 2000 |

Table 8.8 Resume Move Command Block

Unused words are ignored by the 3102i and can be any value, including parameter values from the previous Absolute or Relative Move command.

When resuming an Interpolated Move, the command bits in the Command Bits LSW word must be reset. Therefore, this word can be set to 16#0000 to have the General Purpose Output off, or 16#0080 to have the General Purpose Output on.

Single Axis Command Blocks (continued)**Immediate Stop**

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-------------------------|-------|-----------------------------|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0010 |
| 1 | 9 | <i>Command Bits LSW</i> | | See pg. 59 and Note 1 below |
| 2 | 10 | Unused | | 0 (See Note 2 below) |
| 3 | 11 | Unused | | 0 (See Note 2 below) |
| 4 | 12 | Unused | | 0 (See Note 2 below) |
| 5 | 13 | Unused | | 0 (See Note 2 below) |
| 6 | 14 | Unused | | 0 (See Note 2 below) |
| 7 | 15 | Unused | | 0 (See Note 2 below) |

Table 8.9 Immediate Stop Command Block

- 1) When running an Interpolate Move, the 3102i requires an Immediate Stop command on channel 1. An Immediate Stop instruction on channel 1 stops the move and the motor position values on both axes becomes invalid. An Immediate Stop instruction on channel 2 will not stop the move but the 3102i will issue a Command Error message on channel 1.
- 2) Unused words are ignored by the 3102i and can be any value, including parameter values from the previous Absolute or Relative Move command. The Command Bits LSW word can also be left at its previous value or the word can be set to 16#0000 to have the General Purpose Output off, or 16#0080 to have the General Purpose Output on.

Single Axis Command Blocks (continued)

+Find Home (CW)

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-------------------------|--------------|---|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0020 |
| 1 | 9 | <i>Command Bits LSW</i> | | See pg. 59 and Note below |
| 2 | 10 | Reserved | | Must be 0 |
| 3 | 11 | Reserved | | Must be 0 |
| 4 | 12 | Programmed Speed MSW | Steps/Second | Configured Starting Speed to 1,000,000 (16#000F:4240) |
| 5 | 13 | Programmed Speed LSW | | |
| 6 | 14 | Acceleration | Steps/ms/sec | 1 to 2000 |
| 7 | 15 | Deceleration | Steps/ms/sec | 1 to 2000 |

Table 8.10 +Find Home (CW) Command Block

Only Constant Acceleration is available when homing the 3102i axis. Bit 0 of the Command Bits LSW must equal zero.

-Find Home (CCW)

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-------------------------|--------------|---|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0040 |
| 1 | 9 | <i>Command Bits LSW</i> | | See pg. 59 and Note below |
| 2 | 10 | Reserved | | Must be 0 |
| 3 | 11 | Reserved | | Must be 0 |
| 4 | 12 | Programmed Speed MSW | Steps/Second | Configured Starting Speed to 1,000,000 (16#000F:4240) |
| 5 | 13 | Programmed Speed LSW | | |
| 6 | 14 | Acceleration | Steps/ms/sec | 1 to 2000 |
| 7 | 15 | Deceleration | Steps/ms/sec | 1 to 2000 |

Table 8.11 +Find Home (CW) Command Block

Only Constant Acceleration is available when homing the 3102i axis. Bit 0 of the Command Bits LSW must equal zero.

Single Axis Command Blocks (continued)**+Jog Move (CW)**

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-------------------------|--------------|-------------------|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0080 |
| 1 | 9 | <i>Command Bits LSW</i> | | See pg. 59 |
| 2 | 10 | Reserved | | Must be 0 |
| 3 | 11 | Reserved | | Must be 0 |
| 4 | 12 | Programmed Speed MSW | Steps/Second | 1 to 1,000,000 |
| 5 | 13 | Programmed Speed LSW | | 1 to 16#000F:4240 |
| 6 | 14 | Acceleration | Steps/ms/sec | 1 to 2000 |
| 7 | 15 | Deceleration | Steps/ms/sec | 1 to 2000 |

Table 8.12 +Jog Move (CW) Command Block

+Registration Move (CW)

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-------------------------|--------------|--------------------|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0080 |
| 1 | 9 | <i>Command Bits LSW</i> | | See pg. 59 |
| 2 | 10 | Position Offset MSW | Steps | 1 to 1,073,741,823 |
| 3 | 11 | Position Offset LSW | | 1 to 16#3FFF:FFFF |
| 4 | 12 | Programmed Speed MSW | Steps/Second | 1 to 1,000,000 |
| 5 | 13 | Programmed Speed LSW | | 1 to 16#000F:4240 |
| 6 | 14 | Acceleration | Steps/ms/sec | 1 to 2000 |
| 7 | 15 | Deceleration | Steps/ms/sec | 1 to 2000 |

Table 8.13 +Registration (CW) Command Block

+Encoder Follower Move

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|--------------------------|--------------|---|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0080 |
| 1 | 9 | <i>Command Bits LSW</i> | | 16#0008 AND GPO state bit 16#0008 or 16#0088 allowed |
| 2 | 10 | Encoder Count Multiplier | | 1 to 32,767 |
| 3 | 11 | Encoder Count Divisor | | 1 to 32,767 |
| 4 | 12 | Reserved | | Must be 0 |
| 5 | 13 | Reserved | | Must be 0 |
| 6 | 14 | Acceleration | Steps/ms/sec | 0 to 2000 |
| 7 | 15 | Deceleration | Steps/ms/sec | 0 to 2000 |

Table 8.14 +Encoder Follower Move Command Block

Single Axis Command Blocks (continued)

–Jog Move (CCW)

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-------------------------|--------------|-------------------------------------|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0100 |
| 1 | 9 | <i>Command Bits LSW</i> | | See pg. 59 |
| 2 | 10 | Reserved | | Must be 0 |
| 3 | 11 | Reserved | | Must be 0 |
| 4 | 12 | Programmed Speed MSW | Steps/Second | 1 to 1,000,000 1 to 16#000F:4240 |
| 5 | 13 | Programmed Speed LSW | | |
| 6 | 14 | Acceleration | Steps/ms/sec | 1 to 2000 |
| 7 | 15 | Deceleration | Steps/ms/sec | 1 to 2000 |

Table 8.15 –Jog Move (CCW) Command Block

–Registration Move (CCW)

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-------------------------|--------------|---|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0100 |
| 1 | 9 | <i>Command Bits LSW</i> | | See pg. 59 |
| 2 | 10 | Position Offset MSW | Steps | 1 to 1,073,741,823 1 to 16#3FFF:FFFF |
| 3 | 11 | Position Offset LSW | | |
| 4 | 12 | Programmed Speed MSW | Steps/Second | 1 to 1,000,000 1 to 16#000F:4240 |
| 5 | 13 | Programmed Speed LSW | | |
| 6 | 14 | Acceleration | Steps/ms/sec | 1 to 2000 |
| 7 | 15 | Deceleration | Steps/ms/sec | 1 to 2000 |

Table 8.16 –Registration (CCW) Command Block

–Encoder Follower Move (CCW)

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|--------------------------|--------------|---|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0100 |
| 1 | 9 | <i>Command Bits LSW</i> | | 16#0008 AND GPO state bit 16#0008 or 16#0088 allowed |
| 2 | 10 | Encoder Count Multiplier | | 1 to 32,767 |
| 3 | 11 | Encoder Count Divisor | | 1 to 32,767 |
| 4 | 12 | Reserved | | Must be 0 |
| 5 | 13 | Reserved | | Must be 0 |
| 6 | 14 | Acceleration | Steps/ms/sec | 0 to 2000 |
| 7 | 15 | Deceleration | Steps/ms/sec | 0 to 2000 |

Table 8.17 –Encoder Follower Move Command Block

Single Axis Command Blocks (continued)**Preset Position**

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-------------------------|-------|--|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0200 |
| 1 | 9 | <i>Command Bits LSW</i> | | See pg. 59 |
| 2 | 10 | New Position MSW | Steps | ±1,073,741,823 (16#C000:001 to 16#3FFF:FFF) |
| 3 | 11 | New Position LSW | | |
| 4 | 12 | Reserved | | Must be 0 |
| 5 | 13 | Reserved | | Must be 0 |
| 6 | 14 | Reserved | | Must be 0 |
| 7 | 15 | Reserved | | Must be 0 |

Table 8.18 Preset Position Command Block

Presetting the position will also reset the *Move Complete* status bit.

Preset Encoder Position

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|--------------------------|-------|--|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0400 |
| 1 | 9 | <i>Command Bits LSW</i> | | See pg. 59 |
| 2 | 10 | New Encoder Position MSW | Steps | ±1,073,741,823 (16#C000:001 to 16#3FFF:FFF) |
| 3 | 11 | New Encoder Position LSW | | |
| 4 | 12 | Reserved | | Must be 0 |
| 5 | 13 | Reserved | | Must be 0 |
| 6 | 14 | Reserved | | Must be 0 |
| 7 | 15 | Reserved | | Must be 0 |

Table 8.19 Preset Encoder Position Command Block

Reset Errors

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-------------------------|-------|---|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0800 |
| 1 | 9 | <i>Command Bits LSW</i> | | 16#0000 or 16#0080 Only bit 7 (GPO) may be set |
| 2 | 10 | Reserved | | Must be 0 |
| 3 | 11 | Reserved | | Must be 0 |
| 4 | 12 | Reserved | | Must be 0 |
| 5 | 13 | Reserved | | Must be 0 |
| 6 | 14 | Reserved | | Must be 0 |
| 7 | 15 | Reserved | | Must be 0 |

Table 8.20 Reset Errors Command Block

Single Axis Command Blocks (continued)

Run +Blend Move Profile (CW)

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-------------------------|-------|---|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#1000 |
| 1 | 9 | <i>Command Bits LSW</i> | | 16#0000 or 16#0080 Only bit 7 (GPO) may be set |
| 2 | 10 | Reserved | | Must be 0 |
| 3 | 11 | Reserved | | Must be 0 |
| 4 | 12 | Reserved | | Must be 0 |
| 5 | 13 | Reserved | | Must be 0 |
| 6 | 14 | Reserved | | Must be 0 |
| 7 | 15 | Reserved | | Must be 0 |

Table 8.21 Run +Blend Move Profile (CW) Command Block

Run –Blend Move Profile (CCW)

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-------------------------|-------|---|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#2000 |
| 1 | 9 | <i>Command Bits LSW</i> | | 16#0000 or 16#0080 Only bit 7 (GPO) may be set |
| 2 | 10 | Reserved | | Must be 0 |
| 3 | 11 | Reserved | | Must be 0 |
| 4 | 12 | Reserved | | Must be 0 |
| 5 | 13 | Reserved | | Must be 0 |
| 6 | 14 | Reserved | | Must be 0 |
| 7 | 15 | Reserved | | Must be 0 |

Table 8.22 Run –Blend Move Profile (CW) Command Block

Set Min. Registration Move Distance

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-------------------------|-------|---|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0000 |
| 1 | 9 | <i>Command Bits LSW</i> | | 16#0004 AND GPO state bit 16#0004 or 16#0084 allowed |
| 2 | 10 | Min. Reg. Distance MSW | Steps | ±1,073,741,823 (16#C000:001 to 16#3FFF:FFF) |
| 3 | 11 | Min. Reg. Distance LSW | | |
| 4 | 12 | Reserved | | Must be 0 |
| 5 | 13 | Reserved | | Must be 0 |
| 6 | 14 | Reserved | | Must be 0 |
| 7 | 15 | Reserved | | Must be 0 |

Table 8.23 Set Min. Reg. Move Distance Command Block

Programming Blend Move Profiles

The previous *Run +Blend Move Profile (CW)* and *Run -Blend Move Profile (CCW)* sections on page 67 show the word values needed to perform a Blend Move. This section covers how to program a Blend Move Profile. A Blend Move Profile must be programmed into the 3102i before a Blend Move can be executed.

Multiple writes to the 3102i are required to program a single Blend Move Profile. The 3102i uses a simple handshaking protocol to control this transfer of data. Two bits in the Output Image Table and two bits in the Input Image Table are used in the transfer of data blocks.

The two bits in the Output Image Table were introduced in the *Command Bits LSW* section on page 59.

- ▶ **Program Blend Profile** (Bit 4 of Command Bits LSW) Must make a 0→1 transition to start the Blend Move Profile programming cycle and must remain a “1” while programming the Blend Move Profile.
- ▶ **Program Blend Profile Transmit** (Bit 5 of Command Bits LSW) Make a 0→1 transition on this bit to tell the 3102i that new Blend Move Profile data is available. Make a 1→0 transition on this bit after the 3102i sets its *Program Blend Profile Acknowledge* bit in its Input Image Table data to complete the transfer of the block of data.

The two bits in the Input Image Table are introduced here. The remaining values in the Input Image Table are introduced in the following section.

- ▶ **Program Blend Profile Acknowledge** (Bit 9 of Status Bits LSW) This bit is used by the 3102i to tell the PLC when it has accepted a new data block, and when it is ready to receive the next data block.
- ▶ **Invalid Blend Profile** (Bit 9 of Status Bits MSW) This bit is used by the 3102i to tell the PLC when there was an error in the last Program Blend Profile data block sent to the module.

The following logical steps are used to program a Blend Move Profile.

- 1) Your PLC program writes the *Initial Profile Programming Block* to the 3102i with the *Program Blend Profile* and *Program Blend Profile Transmit* bits both set to 1.
- 2) The 3102i responds by storing the information in the data block, setting the *Invalid Blend Profile* bit if there is an error in the data, and setting the *Program Blend Profile Acknowledge* bit.
- 3) When your PLC program see the *Program Blend Profile Acknowledge* bit set, check the value of the *Invalid Blend Profile* bit and act appropriately.
- 4) Your PLC program then needs to reset the *Program Blend Profile Transmit* bit.
- 5) The 3102i responds by resetting its *Program Blend Profile Acknowledge* bit.
- 6) a: If you have additional data blocks that must be sent to the 3102i, repeat these instructions from step 1.
b: If you have sent the last data block, then reset the *Program Blend Profile* bit to end the programming cycle.



Two sample programs for the 3102i are available on our www.amci.com website. One of them includes the sample code needed to program a Blend Move Profile.

Two different data blocks must be used when programming a Blend Move Profile. The first data block type programs the Number of Segments and the acceleration/deceleration types (Constant, Triangular S-Curve, Trapezoidal S-Curve), used between the segments. This must be the first data block written to the 3102i when programming a Blend Move Profile. The remaining data block type is used to program the Segment Length, the Target Speed, and the actual acceleration/deceleration value. This block is used multiple times in the programming cycle, once for each segment in the profile.

Programming Blend Move Profiles (continued)

Initial Profile Programming Block

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-----------------------------|--|---|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0000 |
| 1 | 9 | <i>Command Bits LSW</i> | | 16#0030 AND GPO state bit 16#0030 or 16#00B0 allowed |
| 2 | 10 | Number of Segments | | 2 to 16 |
| 3 | 11 | Acceleration Type MSW | Programs Accel. Type for Segments 9 - 16 | |
| 4 | 12 | Acceleration Type LSW | Programs Accel. Type for Segments 1 - 8 | |
| 5 | 13 | Deceleration Value and Type | Programs final Deceleration Value and Type used at the end of the profile. | |
| 6 | 14 | Reserved | | Must be 0 |
| 7 | 15 | Reserved | | Must be 0 |

Word 3 configures Segments 9 - 16 for Axis 1
Word 4 configures Segments 1 - 8 for Axis 1
Word 11 configures Segments 9 - 16 for Axis 2
Word 12 configures Segments 1 - 8 for Axis 2

| To Make These Settings... | | ...Set These Bits To | | | | | | | | | | | | | | | |
|---|---------------|----------------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| | | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Acceleration Type Segment 1 or 9 | Constant | | | | | | | | | | | | | | | 0 | 0 |
| | Triangular S | | | | | | | | | | | | | | | 0 | 1 |
| | Trapezoidal S | | | | | | | | | | | | | | | 1 | 1 |
| Segment 2 or 10. Same values as segment 1 | | | | | | | | | | | | | | | | | |
| Segment 3 or 11. Same values as segment 1 | | | | | | | | | | | | | | | | | |
| Segment 4 or 12. Same values as segment 1 | | | | | | | | | | | | | | | | | |
| Segment 5 or 13. Same values as segment 1 | | | | | | | | | | | | | | | | | |
| Segment 6 or 14. Same values as segment 1 | | | | | | | | | | | | | | | | | |
| Segment 7 or 15. Same values as segment 1 | | | | | | | | | | | | | | | | | |
| Segment 8 or 16. Same values as segment 1 | | | | | | | | | | | | | | | | | |

Word 5 configures Axis 1, Word 13 configure Axis 2

| To Make These Settings... | | ...Set These Bits To | | | | | | | | | | | | | | | |
|--|---------------|----------------------|----|----|----|----|------------------------------|---|---|---|---|---|---|---|---|---|---|
| | | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Deceleration Value and Type: Final Segment | Constant | 0 | 0 | 0 | 0 | 0 | Deceleration Value 1 to 2000 | | | | | | | | | | |
| | Triangular S | 0 | 1 | 0 | 0 | 0 | | | | | | | | | | | |
| | Trapezoidal S | 1 | 1 | 0 | 0 | 0 | | | | | | | | | | | |

Table 8.24 Initial Profile Programming Block

Programming Blend Move Profiles (continued)**Secondary Profile Programming Block**

| Axis 1 Word | Axis 2 Word | Function | Units | Range |
|-------------|-------------|-------------------------|---------------|--|
| 0 | 8 | <i>Command Bits MSW</i> | | 16#0000 |
| 1 | 9 | <i>Command Bits LSW</i> | | 16#0030 AND GPO state bit 16#0030 or 16#00B0 allowed |
| 2 | 10 | Segment Length MSW | Counts | 1 to (1,073,741,823 - 1) ^{† ‡} 1 to 16#3FFF:FFFE |
| 3 | 11 | Segment Length LSW | | |
| 4 | 12 | Target Speed MSW | Counts/sec | 1 to 1,000,000 1 to 16#000F:4240 |
| 5 | 13 | Target Speed LSW | | |
| 6 | 14 | Accel/Decel Value | Counts/ms/sec | 1 to 2000 |
| 7 | 15 | Reserved | | Must be 0 |

† Combined length of all segments cannot exceed 1,073,741,823 counts

‡ The Segment Length must be long enough to achieve the Target Speed at the programmed Accel/Decel Value.

Interpolated Move Command Blocks

Linear Interpolated Move

| Word | Function | Units | Range |
|------|---------------------------|--------------|---|
| 0 | <i>Command Bits MSW</i> | | 16#0000 |
| 1 | <i>Command Bits LSW</i> | | See Below |
| 2 | Interpolated Speed MSW | Steps/Second | Configured Starting Speed to 1,000,000 (16#000F:4240) |
| 3 | Interpolated Speed LSW | | |
| 4 | Interpolated Accel. Value | Steps/ms/sec | 1 to 2000 |
| 5 | Interpolated Decel. Value | Steps/ms/sec | 1 to 2000 |
| 6 | EndPointX MSW (axis1) | Steps | ±1,073,741,823 (16#C000:001 to 16#3FFF:FFF) |
| 7 | EndPointX LSW (axis1) | | |
| 8 | Reserved | | Must be 0 |
| 9 | Reserved | | Must be 0 |
| 10 | EndPointY MSW (axis2) | Steps | ±1,073,741,823 (16#C000:001 to 16#3FFF:FFF) |
| 11 | EndPointY LSW (axis2) | | |
| 12 | Reserved | | Must be 0 |
| 13 | Reserved | | Must be 0 |
| 14 | Reserved | | Must be 0 |
| 15 | Reserved | | Must be 0 |

Table 8.25 Linear Interpolated Move Command Block

The *Command Bits LSW*, fully described in page 59, contains the command bits as well as the Acceleration Type parameter.

- For a Relative Linear Interpolated Move, the upper byte of the Command Bits LSW must equal 16#04. (Bit 10 set)
- For an Absolute Linear Interpolated Move, the upper byte of the Command Bits LSW must equal 16#06. (Bits 10 and 9 set)
- For Constant Acceleration, the lower four bits of the Command Bits LSW can equal 16#0 or 16#2. (All four bits reset or bit 1 set)
- For Triangular S-Curve Acceleration, the lower four bits of the Command Bits LSW must equal 16#1. (Bit 0 set)
- For Trapezoidal S-Curve Acceleration, the lower four bits of the Command Bits LSW must equal 16#3. (Bits 1 and 0 set)
- If you are using the General Purpose Output, its state must be taken into consideration when issuing this command. Bit 7 of the Command Bits LSW controls the on/off state of the General Purpose Output.

Interpolated Move Command Blocks (continued)**Via Point Circular Interpolated Move**

| Word | Function | Units | Range |
|------|---------------------------|--------------|---|
| 0 | <i>Command Bits MSW</i> | | 16#0000 |
| 1 | <i>Command Bits LSW</i> | | See Below |
| 2 | Interpolated Speed MSW | Steps/Second | Configured Starting Speed to 1,000,000 (16#000F:4240) |
| 3 | Interpolated Speed LSW | | |
| 4 | Interpolated Accel. Value | Steps/ms/sec | 1 to 2000 |
| 5 | Interpolated Decel. Value | Steps/ms/sec | 1 to 2000 |
| 6 | EndPointX MSW (axis1) | Steps | ±1,073,741,823 (16#C000:001 to 16#3FFF:FFF) |
| 7 | EndPointX LSW (axis1) | | |
| 8 | Reserved | | Must be 0 |
| 9 | Reserved | | Must be 0 |
| 10 | EndPointY MSW (axis2) | Steps | ±1,073,741,823 (16#C000:001 to 16#3FFF:FFF) |
| 11 | EndPointY LSW (axis2) | | |
| 12 | ViaPointX MSW (axis1) | Steps | ±1,073,741,823 (16#C000:001 to 16#3FFF:FFF) |
| 13 | ViaPointX LSW (axis1) | | |
| 14 | ViaPointY MSW (axis2) | Steps | ±1,073,741,823 (16#C000:001 to 16#3FFF:FFF) |
| 15 | ViaPointY LSW (axis2) | | |

Table 8.26 Via Point Circular Interpolated Move Command Block

The *Command Bits LSW*, fully described in page 59, contains the command bits as well as the Acceleration Type parameter.

- ▶ For a Relative Via Point Circular Interpolated Move, the upper byte of the Command Bits LSW must equal 16#08. (Bit 11 set)
- ▶ For an Absolute Via Point Circular Interpolated Move, the upper byte of the Command Bits LSW must equal 16#0A. (Bits 11 and 9 set)
- ▶ For Constant Acceleration, the lower four bits of the Command Bits LSW can equal 16#0 or 16#2. (All four bits reset or bit 1 set)
- ▶ For Triangular S-Curve Acceleration, the lower four bits of the Command Bits LSW must equal 16#1. (Bit 0 set)
- ▶ For Trapezoidal S-Curve Acceleration, the lower four bits of the Command Bits LSW must equal 16#3. (Bits 1 and 0 set)
- ▶ If you are using the General Purpose Output, its state must be taken into consideration when issuing this command. Bit 7 of the Command Bits LSW controls the on/off state of the General Purpose Output.

Interpolated Move Command Blocks (continued)

Center Point Circular Interpolated Move

| Word | Function | Units | Range |
|------|---------------------------|--------------|---|
| 0 | <i>Command Bits MSW</i> | | 16#0000 |
| 1 | <i>Command Bits LSW</i> | | See Below |
| 2 | Interpolated Speed MSW | Steps/Second | Configured Starting Speed to 1,000,000 (16#000F:4240) |
| 3 | Interpolated Speed LSW | | |
| 4 | Interpolated Accel. Value | Steps/ms/sec | 1 to 2000 |
| 5 | Interpolated Decel. Value | Steps/ms/sec | 1 to 2000 |
| 6 | EndPointX MSW (axis1) | Steps | ±1,073,741,823 (16#C000:001 to 16#3FFF:FFF) |
| 7 | EndPointX LSW (axis1) | | |
| 8 | Reserved | | Must be 0 |
| 9 | Reserved | | Must be 0 |
| 10 | EndPointY MSW (axis2) | Steps | ±1,073,741,823 (16#C000:001 to 16#3FFF:FFF) |
| 11 | EndPointY LSW (axis2) | | |
| 12 | CenterPointX MSW (axis1) | Steps | ±1,073,741,823 (16#C000:001 to 16#3FFF:FFF) |
| 13 | CenterPointX LSW (axis1) | | |
| 14 | CenterPointY MSW (axis2) | Steps | ±1,073,741,823 (16#C000:001 to 16#3FFF:FFF) |
| 15 | CenterPointY LSW (axis2) | | |

Table 8.27 Center Point Circular Interpolated Move Command Block

The *Command Bits LSW*, fully described in page 59, contains the command bits as well as the Acceleration Type parameter.

- For a Relative Center Point Circular Interpolated Move that moves in a clockwise direction, the upper byte of the Command Word LSW must equal 16#10. (Bit 12 set)
- For a Relative Center Point Circular Interpolated Move that moves in a counter-clockwise direction, the upper byte of the word must equal 16#20. (Bit 13 set)
- For an Absolute Center Point Circular Interpolated Move that moves in a clockwise direction, the upper byte of the word must equal 16#12. (Bits 12 and 9 set)
- For an Absolute Center Point Circular Interpolated Move that moves in a counter-clockwise direction, the upper byte of the word must equal 16#22. (Bits 13 and 9 set)
- For Constant Acceleration, the lower four bits of the Command Bits LSW can equal 16#0 or 16#2. (All four bits reset or bit 1 set)
- For Triangular S-Curve Acceleration, the lower four bits of the Command Bits LSW must equal 16#1. (Bit 0 set)
- For Trapezoidal S-Curve Acceleration, the lower four bits of the Command Bits LSW must equal 16#3. (Bits 1 and 0 set)
- If you are using the General Purpose Output, its state must be taken into consideration when issuing this command. Bit 7 of the Command Bits LSW controls the on/off state of the General Purpose Output.

Interpolated Move Command Blocks (continued)**Radius Circular Interpolated Move**

| Word | Function | Units | Range |
|------|---------------------------|--------------|---|
| 0 | <i>Command Bits MSW</i> | | 16#0000 |
| 1 | <i>Command Bits LSW</i> | | See Below |
| 2 | Interpolated Speed MSW | Steps/Second | Configured Starting Speed to 1,000,000 (16#000F:4240) |
| 3 | Interpolated Speed LSW | | |
| 4 | Interpolated Accel. Value | Steps/ms/sec | 1 to 2000 |
| 5 | Interpolated Decel. Value | Steps/ms/sec | 1 to 2000 |
| 6 | EndPointX MSW (axis1) | Steps | ±1,073,741,823 (16#C000:001 to 16#3FFF:FFF) |
| 7 | EndPointX LSW (axis1) | | |
| 8 | Reserved | | Must be 0 |
| 9 | Reserved | | Must be 0 |
| 10 | EndPointY MSW (axis2) | Steps | ±1,073,741,823 [†] (16#C000:001 to 16#3FFF:FFF) |
| 11 | EndPointY LSW (axis2) | | |
| 12 | Radius MSW | Steps | ±1,073,741,823 [†] (16#C000:001 to 16#3FFF:FFF) |
| 13 | Radius LSW | | |
| 14 | Reserved | | Must be 0 |
| 15 | Reserved | | Must be 0 |

Table 8.28 Radius Circular Interpolated Move Command Block

[†] The Radius method defines two circles on the plane. (See *Radius Method* on page 30 for a full description of this method.) If the Radius value is positive, the move will travel the shortest arc in the specified direction. If the Radius value is negative, the move will travel the longest arc in the specified direction.

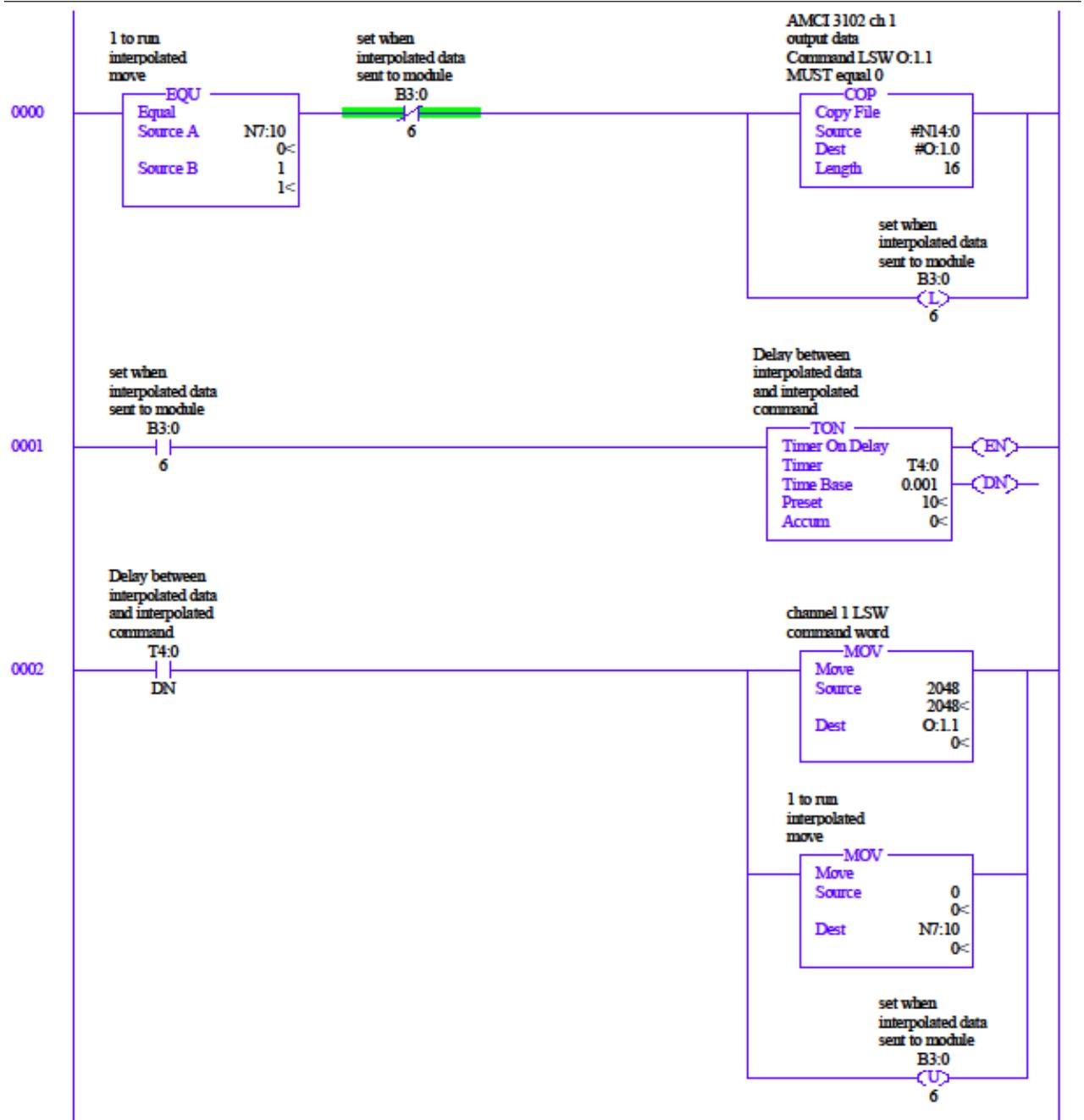
The *Command Bits LSW*, fully described in page 59, contains the command bits as well as the Acceleration Type parameter.

- For a Relative, Radius Circular Interpolated Move that travels in a clockwise direction, the upper byte of the word must equal 16#50. (Bits 14 and 12 set)
- For a Relative, Radius Circular Interpolated Move that travels in a counter-clockwise direction, the upper byte of the word must equal 16#60. (Bits 14 and 13 set)
- For an Absolute, Radius Circular Interpolated Move that travels in a clockwise direction, the upper byte of the word must equal 16#52. (Bits 14, 12, and 9 set)
- For an Absolute, Radius Circular Interpolated Move that travels in a counter-clockwise direction, the upper byte of the word must equal 16#62. (Bits 14, 13, and 9 set)
- For Constant Acceleration, the lower four bits of the Command Bits LSW can equal 16#0 or 16#2. (All four bits reset or bit 1 set)
- For Triangular S-Curve Acceleration, the lower four bits of the Command Bits LSW must equal 16#1. (Bit 0 set)
- For Trapezoidal S-Curve Acceleration, the lower four bits of the Command Bits LSW must equal 16#3. (Bits 1 and 0 set)
- If you are using the General Purpose Output, its state must be taken into consideration when issuing this command. Bit 7 of the Command Bits LSW controls the on/off state of the General Purpose Output.

Writing Interpolated Move Commands to the 3102i

The MicroLogix backplane transmits data in eight word blocks. Because of this, you must write the data for an interpolated move to the 3102i with the command bits reset, wait 10 milliseconds to guarantee the data is written to the module, and then write the command word to the module with the command bits set. If you don't do this, the 3102i may issue a Command Error because words 8 - 15 may be incorrect when the modules attempts to start a move.

The following code, which is also available in the sample program, can be used to write an Interpolated Move Command to the 3102i. The sample program is available on our website, www.amci.com, under the "Tech Library" heading.



Input Data Format

The format of the input image table words when both axes are in Command Mode is shown below.

| Word I:X._ | Ch | Function | Units | Range |
|---------------|----|---|--------------------|--|
| 0 | 1 | Status Bits MSW | | See Description below |
| 1 | 1 | Status Bits LSW | | See Description Below |
| 2 | 1 | Current Position | Steps | -2,147,483,648 to 2,147,483,647 (32 bit signed value) |
| 3 | 1 | | | |
| 4 | 1 | Current Encoder Position / Diagnostic Feedback Value | Counts | -2,147,483,648 to 2,147,483,647 (32 bit signed value) |
| 5 | 1 | | | |
| 6 | 1 | Capture Data | Steps or Counts | -2,147,483,648 to 2,147,483,647 (32 bit signed value) |
| 7 | 1 | | | |
| 8 | 2 | Status Bits MSW | | See Description below |
| 9 | 2 | Status Bits LSW | | See Description below |
| 10 | 2 | Current Position | Steps | -2,147,483,648 to 2,147,483,647 (32 bit signed value) |
| 11 | 2 | | | |
| 12 | 2 | Current Encoder Position / Diagnostic Feedback Value | Counts | -2,147,483,648 to 2,147,483,647 (32 bit signed value) |
| 13 | 2 | | | |
| 14 | 2 | Capture Data | Steps or Counts | -2,147,483,648 to 2,147,483,647 (32 bit signed value) |
| 15 | 2 | | | |

Table 8.29 Input Data Format - Command Data

Status Bits MSW

Word 0 is for Axis 1, Word 8 is for Axis 2

- BIT 0: CW Motion:** Set when the 3102i axis is outputting pulses for clockwise motion.
- BIT 1: CCW Motion:** Set when the 3102i axis is outputting pulses for counter-clockwise motion.
- BIT 2: Hold State:** Set when the 3102i axis has stopped motion as a result of a Hold Move Command. See page 61 for information on the *Hold Move* Command.
- BIT 3: Axis Stopped:** Set when the 3102i axis is not in motion for any reason.
- BIT 4: At Home:** This bit is only set after the successful completion of a homing command. The next command that causes motion will reset this bit. Any move off of the home position followed by any move back on to the home position will not result in this bit being set again. The Home Input State bit, found in the Status Bits LSW will always turn on when the Home Input is active.
- BIT 5: Move Accelerating:** Set when the 3102i axis is accelerating during any move.
- BIT 6: Move Decelerating:** Set when the 3102i axis is decelerating during any move.
- BIT 7: Move Complete:** Set when the 3102i axis has successfully completed an Absolute, Relative, Blend, or Interpolated Move. An Interpolated Move will set the Move Complete bits of both axes. This bit is reset by the next move command or by a Preset Position command.
- BIT 8: Home Invalid Error:** Set when the 3102i could not home the axis because of an error in a \pm Find Home command data, a hardware switch activated during the homing operation, or the homing operation exceeded its programmed timeout value.

Input Data Format (continued)

Status Bits MSW (continued)

BIT 9: Invalid Profile: Set when there was an error in the last Program Blend Profile data block written to the 3102i axis. See the *Programming Blend Move Profiles* section starting on page 68 for information on the format of the data blocks.

BIT 10: Position Invalid: Set under the following conditions:

- 1) Axis switched from Configuration Mode to Command Mode
- 2) An Immediate Stop command was issued to the axis while motion was occurring
- 3) An Emergency Stop input was activated while motion was occurring.
- 4) If either of the CW or CCW Limit inputs become active during a Relative, Absolute, Blend, or Interpolated Move.

BIT 11: Input Error:

±Find Home Commands: Set when an E-Stop input activates during homing operation or when the limit switch that is opposite to the direction of travel activates, for example when the CCW Limit activates during a +Find Home command.

±Jog Move Commands: Set when an E-Stop input activates during the move or when the limit switch that is in the direction of travel activates, for example when the CW Limit activates during a +Jog Move command.

±Registration Move Commands: Set when an E-Stop input activates during the move or when the limit switch that is in the direction of travel activates, for example when the CW Limit activates during a +Registration Move command.

All Other Move Commands: Set when an E-Stop input activates during the move or when either of the end limit switches activates.

Note that the Input Error bit is latched but you do not have to issue a Clear Errors command block to clear the bit. The next move command will clear this bit automatically if none of the inputs are active when the command is issued.

BIT 12: Command Error: Set when the last command issued to the 3102i axis forced an error. When there is an error in an Interpolated Move, only the Command Error bit for axis 1 is set. This bit must be reset by issuing a *Reset Errors* command to the 3102i axis. Other commands, including the *Immediate Stop* command, will not be accepted until this error is reset.

Global Status Bits

**Word 0 is for Axis 1,
Word 8 is for Axis 2**

The upper three bits the Status Bits MSW have same meaning regardless of the mode the axis is in. The figure to the right shows meaning of each of these status bits.

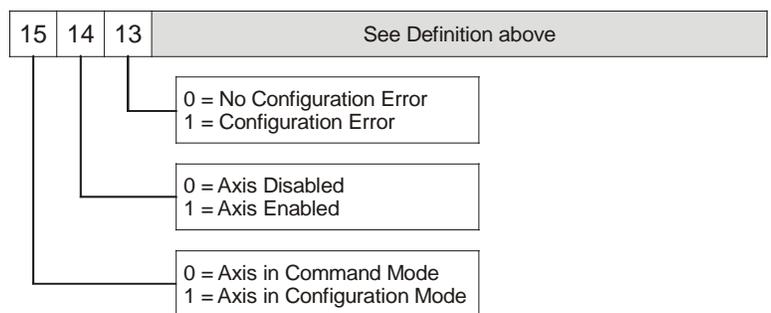


Figure 8.1 Global Status Bit

Input Data Format (continued)**Status Bits LSW**

Word 1 is for Axis 1,
Word 9 is for Axis 2

- BIT 0: Capture Input State:** Set when the Capture Input is active. Its active state is set in the *Configuration Bits LSW* when the axis is configured. See page 53 for additional information.
- BIT 1: External Input State:** Set when the External Input is active. Its active state is set in the *Configuration Bits LSW* when the axis is configured. See page 53 for additional information.
- BIT 2: Home Input State:** Set when the Home Input is active. Its active state is set in the *Configuration Bits LSW* when the axis is configured. See page 53 for additional information.
- BIT 3: CW Limit/Emergency Stop Input State:** Set when the CW Input/E-Stop Input is active. The function of this input is set in the *Configuration Bits MSW* when the axis is configured and its active state is set in the *Configuration Bits LSW*. See pages 52 and 53 for additional information.
- BIT 4: CCW Limit/Emergency Stop Input State:** Set when the CCW Input/E-Stop Input is active. The function of this input is set in the *Configuration Bits MSW* when the axis is configured and its active state is set in the *Configuration Bits LSW*. See pages 52 and 53 for more information.
- BITS 5 & 6: Reserved:** These bits will always equal zero.
- BIT 7: Encoder Follower Mode:** Set when the axis is in Encoder Follower Mode.
- BIT 8: Running Blend Move:** Set the axis is presently running a Blend Move Profile. This bit is reset when the move comes to an end.
- BIT 9: Blend Move Acknowledge Bit:** Set when 3102i has accepted a Blend Move Profile programming block. See *Programming Blend Move Profiles*, which starts on page 68, for additional information.
- BIT 10: Non-Zero Minimum Registration Distance:** Set when the Minimum Registration Move Distance parameter is programmed to a non-zero value.
- BIT 11: Axis 1: Running Interpolated Move:** Set when the 3102i is running an Interpolated Move. This bit is reset when the move comes to an end.
Axis 2: Reserved: This bit will always equal zero.
- BIT 12: Backplane Home Proximity Bit State:** This bit always equals the state of the Backplane Home Proximity bit, which is bit 6 in the *Command Bits LSW* for the axis.
- BIT 13: General Purpose Output State:** This bit always equals the state of the G.P Output State bit, which is bit 7 in the *Command Bits LSW* for the axis.
- BITS 14 & 15: Reserved:** These bits will always equal zero.

Input Data Format (continued)

Current Position

Words 2 & 3 for Axis 1, Words 10 & 11 for Axis 2

These two words form a 32 bit value. This value can be thought of as the motor position and it is affected by move commands to the axis. This value can be preset using the Preset Position command.

Encoder Position/Diagnostic Feedback Value

Words 4 & 5 for Axis 1, Words 12 & 13 for Axis 2

These two words form a 32 bit value. This value can be thought of as the encoder position and it changes in response to pulses on the encoder inputs. This value can be preset using the Preset Encoder Position command.

When running in Diagnostic Feedback Mode, a Preset Position command will preset the motor's Current Position register pair as well as this register pair. (Both register pairs will have the same value after a Position Preset command.) The Preset Encoder Position command will cause a Command Error while in this Diagnostic Feedback Mode.

Capture Data

Words 6 & 7 for Axis 1, Words 14 & 15 for Axis 2

These two words form a 32 bit value. This value is updated when the Capture Input for the axis makes an inactive-to-active transition. (The active state of the Capture Input is configurable in the *Configuration Bits LSW*, as described on page 53.) If the axis has been configured to not use an encoder, then the Capture Data value will be taken from the Current Position value. If the axis is configured to use an encoder, then the Capture Data value will be taken from the Encoder Position value. The axis is configured to use, or not use, an encoder by the state of bit 8 of the *Configuration Bits MSW* as described on page 52.

Notes

TABULATED COMMAND & ERROR CODES

This chapter is broken down into two sections. The first section lists status bits and the conditions that will set the bit. You can assume that the bit will be reset under all conditions that are not listed. The second section lists the available commands and the errors that can occur with them.

Status Bits

Axis Stopped

Axis1: I:X.0/3 Axis2: I:X.8/3

- 1) Set whenever the axis is in command mode and motion is not occurring.

Move Complete Bit

Axis1: I:X.0/7 Axis2: I:X.8/7

- 1) Set after an Absolute, Relative, Blend, or Interpolated Move has been successively completed. The Move Complete bit will be reset when the next move operation is initiated or by a Preset command. This bit is also reset when switching from Configuration Mode to Command Mode.

Home Invalid Bit

Axis1: I:X.0/8 Axis2: I:X.8/8

The Home Invalid bit will be set under the following conditions:

- 1) Issuing a \pm Find Home command with the Programmed Speed less than the configured Starting Speed or greater than 1,000,000 steps/second.
- 2) Issuing a \pm Find Home command using either of the S-Curve Acceleration or Deceleration Types. (Only *Constant Acceleration* is available with \pm Find Home commands.)
- 3) Issuing a \pm Find Home command while having an acceleration or deceleration values outside their valid range of 1 to 2000.
- 4) Issuing a Homing Command with any reserved words not equal to zero.
- 5) If the opposite End Limit Switch is reached during a homing operation. For example, if the CCW Limit Switch is reached during a CW homing operation. Improper wiring, configuration, or placement of the limit switch would most likely cause this error.
- 6) If the CW Limit or CCW Limit activate at the same time as the Home Limit.
- 7) If the time of the homing operation exceeds the configured Homing Timeout value.

Invalid Profile Bit

Axis1: I:X.0/9 Axis2: I:X.8/9

The Invalid Profile bit will be set under the following conditions

- 1) If the Programmed Speed of a Relative, Absolute, Blend, or Interpolated Move is less than the configured Starting Speed
- 2) If any of the move parameters are outside of their valid ranges.
- 3) If the defined profile overflows the internal registers of the 3102i module.
- 4) Setting the number of Blend Move Segments to be less than 2 or greater than 16.
- 5) If a Blend Move segment does not reach the Programmed Speed. That is, if the distance that it takes to accelerate to the programmed speed exceeds the length of the segment.
- 6) Two consecutive Blend Move segments with the same Programmed Speed.
- 7) Blend Move profile Initial Profile Programming Block that specifies an Acceleration Type of "10" in any of the bit pair in words 3,4 & 5 (axis1) or 11,12 & 13 (axis2)

Status Bits (continued)**Invalid Profile Bit (continued)****Axis1: I:X.0/9 Axis2: I:X.8/9**

- 8) Changing the speed of a \pm Jog Move whose initial speed was less than or equal to the configured Starting Speed.
- 9) If invalid changes are made to the velocity, acceleration, or deceleration parameters of a \pm Jog Move while it is occurring. When detected, motion will immediately stop.
- 10) If a non-zero value is entered into the Reserved words (Axis1: 2/3, Axis2: 10/11), while a \pm Jog Move is occurring. Motion will immediately stop.
- 11) If the Registration Move is started with the External Input active.
- 12) Starting a \pm Registration Move if the number of steps programmed into the Position Offset register is less than the number of steps required to decelerate and stop the motion. In this case the \pm Registration Move will not start.
- 13) If the Position Offset of a \pm Registration Move is negative.
- 14) If the value of the Position Offset registers is changed after a \pm Registration Move has been initiated. Motion will immediately stop.
- 15) Increasing the speed of a \pm Registration Move while it is occurring. Motion will immediately stop. Please note that it is possible to decrease the speed of \pm Registration Move while it is occurring.
- 16) Changing the acceleration or deceleration parameters during a \pm Registration Move, however, these values are only checked when the Programmed Speed is changed. Motion will immediately stop.
- 17) Data in reserved registers during an Interpolated Move.

Position Invalid Bit**Axis1: I:X.0/10 Axis2: I:X.8/10**

The Position Invalid will be set under the following conditions

- 1) When switching from Configuration Mode to Command Mode. This will be true even if the position was valid before the configuration operation occurred.
- 2) After an Immediate Stop command has been issued.
- 3) If the Emergency Stop input has been activated.
- 4) If either of the CW Limit or CCW Limit inputs become active during a Relative, Absolute, Blend, Interpolated or Encoder Follower Move.
- 5) If the limit input that is the same as the commanded direction of travel is reached during a Jog or Registration Move. For example, if the CW Limit Switch is reached during a +Jog Move.
- 6) If the limit input that is the opposite of the commanded direction of travel is reached during a homing operation. For example, if the CCW Limit Switch is reached during a +Find Home command.

Input Error Bit**Axis1: I:X.0/11 Axis2: I:X.8/11**

The Input Error bit will be set under the following conditions

- 1) If an Emergency Stop input is activated.
- 2) If either of the CW Limit or CCW Limit inputs become active during a Relative, Absolute, Blend, Encoder Follower, or Interpolated Move.
- 3) If the limit input that is the same as the commanded direction of travel is reached during a Jog or Registration Move. For example, if the CW Limit Switch is reached during a +Jog Move.
- 4) If the limit input that is the opposite of the commanded direction of travel is reached during a homing operation. For example, if the CCW Limit Switch is reached during a +Find Home command.
- 5) Performing a \pm Jog Move in the same direction as the active End Limit Switch. For example, starting a +Jog Move when the CW Limit Switch is active.

Status Bits (continued)

Command Error Bit

Axis1: I:X.0/12 Axis2: I:X.8/12

The Command Error bit will be set under the following conditions

- 1) Attempting to enter Configuration Mode while a move operation is occurring on the axis. The move will run to completion and the axis will then enter Configuration Mode, clearing the error.
- 2) If more than one bit in the Command Bits MSW is set at one time. (Axis1: O:X.0, Axis2: O:X.8)
- 3) If any bit in the Command Bits MSW word is set while also commanding an Interpolated Move with the Command Bits LSW bits 9 through 14.
- 4) If a move operation is started before the previous move operation is completed. The present move will run to completion.
- 5) If the Target Position, Programmed Speed, Acceleration, or Deceleration parameters are outside of their valid ranges.
- 6) If the Programmed Speed is less than the configured Starting Speed for Absolute, Relative, Blend, and Interpolated Moves as well as Find Home commands. Jog and Registration Moves can be run at speeds less than the starting speed.
- 7) Performing an Absolute Move or an Absolute Interpolated Move when the Current Position is not valid.
- 8) Issuing a Hold Move command while a Blend, Registration, or Find Home command is running.
- 9) Issuing a Resume Move command when the axis is not in a Hold state.
- 10) Issuing any command with the Reserved words in the command having a non-zero value.
- 11) Issuing a Find Home command with S-Curve Acceleration selected.
- 12) Issuing a Find Home command before configuring a homing input. (Home Input or Z-pulse)
- 13) Issuing a Preset Encoder command when axis is not configured to use an encoder.

Configuration Error Bit

Axis1: I:X.0/13 Axis2: I:X.8/13

The Configuration Error bit will be set under the following conditions

- 1) At power up, before the 3102i has been configured for the first time.
- 2) If an invalid configuration has been written to the axis.

Command Error Conditions By Command

This section lists the errors that will cause the Command Error bit to be set. The Command Error bit is I:X.0/13 for axis 1 and I:X.8/13 for axis 2.

±Jog Move Command

- 1) Changing the speed of a Jog Move whose initial speed was less than or equal to the configured Starting Speed.
- 2) If invalid changes are made to the velocity, acceleration, or deceleration parameters while the move is occurring. When detected, motion will immediately stop.
- 3) If a non-zero value is entered in the Reserved registers while a Jog Move is occurring. Motion will immediately stop. If a non-zero value is entered when starting a Jog Move, the 3102i axis will perform a Registration Move instead.

Command Error Conditions By Command (continued)**±Blend Move Command**

- 1) If a Blend Move command is initiated before the profile was programmed for the axis.
- 2) Programming a Blend Move Profile without a final deceleration value. This parameter is programmed in word 5 of the Initial Profile Programming Block.
- 3) If a Blend Move command is sent to the axis with any of the Reserved words not set to zero.
- 4) If a blend move profile is programmed while a blend move operation is in progress. If this occurs, the deceleration value of the final segment will be used to bring the move to a Controlled Stop.
- 5) If the Hold Move command is issued while a blend move profile is running. If this occurs, the command error bit will be set while the blend move profile runs to completion. When completed, the command error will be reset.

Registration Move Command

- 1) If the Minimum Registration Distance is outside of the valid range or is negative.
- 2) If the Hold Command bit is set while a Registration Move is occurring.
- 3) If the Registration Move is started with the External Input active.
- 4) If the number of steps programmed into the Position Offset registers during a Registration Move is less than the number of steps required to decelerate and stop the motion. In this case the motion will not start.
- 5) If the number of steps programmed into the Position Offset registers is negative.
- 6) If the value of the Position Offset registers is changed after a Registration Move has been initiated. Motion will immediately stop.
- 7) If the Registration Move speed is increased while motion is occurring. Motion will immediately stop. Please note that it is allowed to decrease the speed while a Registration Move is occurring.
- 8) Changing the acceleration or deceleration parameters. These values are only checked when the Programmed Speed is changed. Motion will immediately stop.

Encoder Follower Operations

- 1) If the 3102i axis is placed in Encoder Follower mode, without first configuring the axis to use an encoder.
- 2) Trying to place the axis in Encoder Follower mode while there is an active Move command running. Please note that even though the command error status bit will be set, the original move operation will run to completion. You can switch to Encoder Follower Mode while motion is occurring on the encoder inputs.
- 3) If the Multiplier or Divisor values are outside of the range of 1 to 32767.
- 4) If either Reserved words are not equal to zero.
- 5) Attempt to change the direction while there is motion on the axis. The Axis Stopped bit (Axis1: I:X.0/3, Axis2: I:X.8/3) must be set before the direction can be changed. You must also end the first Encoder Follower command by resetting the Command Bits MSW word to zero before issuing the Encoder Follower command that reverses direction.
- 6) If any command is issued, including the Hold Command, when the axis module is in Encoder Follower mode. That is, if any of the bits in the Command Bits MSW other than +Jog Move or -Jog Move bits are set when the Encoder Follower Bit is also set.
- 7) If the Encoder Follower Bit is reset while the ±Jog Move bits remain set during an Encoder Follower operation.
- 8) Selecting S curve acceleration with Encoder Follower mode.

Notes



ADVANCED MICRO CONTROLS INC.

20 GEAR DRIVE, TERRYVILLE, CT 06786 T: (860) 585-1254 F: (860) 584-1973

www.amci.com

LEADERS IN ADVANCED CONTROL PRODUCTS