

ANF1 & ANF2

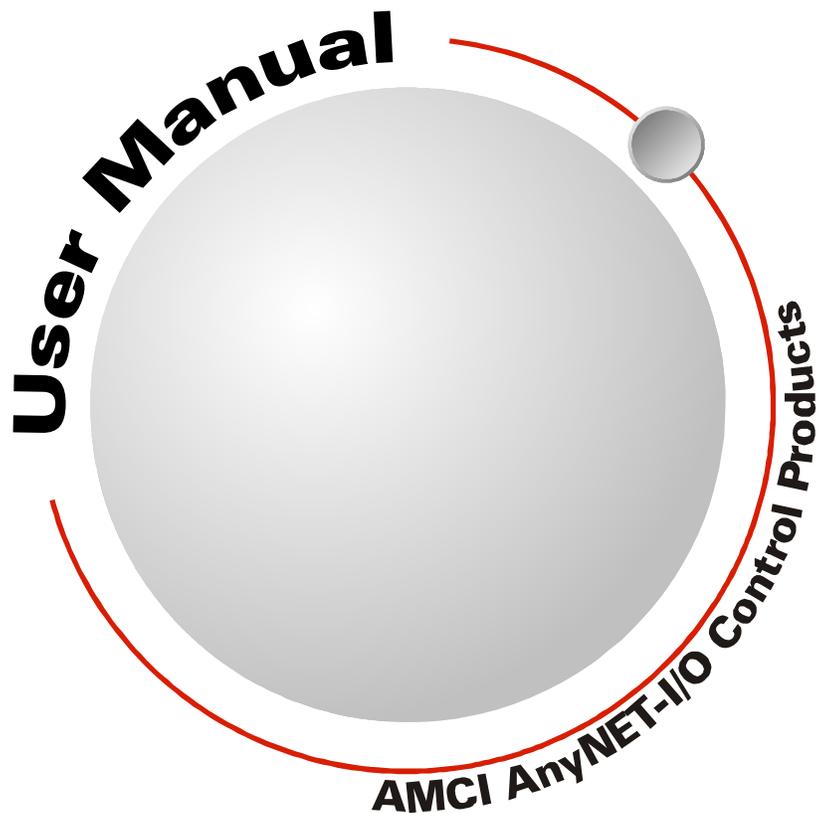
AnyNET-I/O

1 or 2 Axis

Servo/Stepper

Controller

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 2015 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 2015 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	7
Applicable Units	7
Trademark Notices	7
Revision Record	7
Navigating this Manual	7
Manual Conventions	8
Where To Go From Here	8

Chapter 1: Introduction to the ANF1 and ANF2

AnyNET-I/O	9
Module Overview	9
Front Panel Status LED's	10
Modes of Operation	11
Configuration Mode	11
Command Mode	11
Digital Outputs	11
Set_State Bit	12
Specifications	13
Compatible Equipment	14
Servo Equipment	14
Stepper Drivers	14
Stepper Motors	14

Chapter 2: Move Profiles

Units of Measure	15
Definition of Count Direction	15
Definition of Home Position	16
Definition of Starting Speed	16
Definition of Target Position	16
Definition of Acceleration Types	17
Linear Acceleration	17
Triangular S-Curve Acceleration	17
Trapezoidal S-Curve Acceleration	17
A Simple Move	18

Chapter 2: Move Profiles (continued)

Controlled and Immediate Stops	19
Backplane Control	19
Hardware Control	19
Profile Equations	20
Acceleration Equations	20
Total Time Equations	21
Available Move Types	21
Relative Move	21
Absolute Move	21
±Jog Move	22
±Registration Move	23
±Blend Move	24
±Encoder Follower Move	25

Chapter 3: Interpolated Move Profiles

The X-Y Plane	27
Linear Interpolated Moves	27
Relative Linear Move	27
Absolute Linear Move	28
Required Parameters	28
Circular Interpolated Moves	29
Center Point Method	29
Radius Method	30
Via Point Method	31
Controlled and Immediate Stops	31
Backplane Control	31
Hardware Control	32
Welcome to the Integer World	32

Chapter 4: Homing the ANF1/2

Definition of Home Position	33
Position Preset	33
±Find Home Commands	33
Homing Inputs	33
Physical Inputs	33
Backplane Inputs	33
Homing Configurations	34
Homing Profiles	34
Home Input Only Profile	34
Profile with Proximity Input	35
Profile with Overtravel Limit	36

Chapter 5: Installing the ANF1/2

Module Installation 37

 Module Location 37

 Safe Handling Guidelines 37

 Prevent Electrostatic
 Damage37

 Prevent Debris From
 Entering the Module 37

 Remove Power Before
 Servicing 37

Module Mounting 38

 Dimensions 38

 Compatible DIN Rail 39

 Installing IC-5
 Connectors 39

 Mounting the
 AnyNET-I/O Module 39

Stack Addressing 40

Powering the Module 40

 Required Power 40

 Wiring 41

I/O Connector Pin Out 42

Driver Wiring 43

 Differential Wiring 43

 Open Collector Wiring 43

Digital Output Wiring 44

Discrete Input Wiring 45

Encoder Wiring 46

 Differential Encoder Wiring 46

 Single Ended Wiring 47

Diagnostic Feedback Wiring 48

**Chapter 6: Configuration Mode
Data Format**

Output Data Format 49

 Configuration Bits MSW 50

 Configuration Bits LSW 51

 Starting Speed 51

 Homing Timeout 53

Input Data Format 53

Global Status Bits 54

Invalid Configurations 54

 Configuration Bits MSW 54

 Configuration Bits LSW 55

 Starting Speed Parameter 55

 Homing Timeout Parameter 55

 Reserved Words 55

**Chapter 7: Command Mode Data
Format**

Multi-Word Parameters 56

Output Data Format 57

 Command Bits MSW 58

 Command Bits LSW 59

Command Bits Must Transition 59

Single Axis Command Blocks 60

 Absolute Move 60

 Relative Move 60

 Hold Move 61

 Resume Move 61

 Immediate Stop 62

 +Find Home (CW) 63

 –Find Home (CCW) 63

 +Jog Move (CW) 64

 +Registration Move (CW) 64

 +Encoder Follower Move 65

 –Jog Move (CCW) 65

 –Registration Move (CCW) 66

 –Encoder Follower Move 66

 Preset Position 67

 Preset Encoder Position 67

 Reset Errors 68

 Run +Blend Move Profile (CW) ... 68

 Run –Blend Move Profile 69

 Set Min. Registration
 Move Distance 69

Programming Blend Move Profiles 70

 Initial Profile

 Programming Block 71

 Secondary Profile

 Programming Block 72

Interpolated Move Command Blocks 73

 Linear Interpolated Move 73

 Via Point Circular

 Interpolated Move 74

 Center Point Circular

 Interpolated Move 75

 Radius Circular

 Interpolated Move 76

Input Data Format 77

 Status Bits MSW 77

 Status Bits LSW 79

 Network Connectivity, Word 9 80

 Current Position 81

 Encoder Position/Diagnostic

 Feedback Value 81

 Capture Data 81

TABLE OF CONTENTS

Chapter 8: Tabulated Command and Error Codes

Status Bits	82
Axis Stopped	82
Move Complete Bit	82
Home Invalid Bit	82
Profile Invalid Bit	82
Position Invalid Bit	84
Input Error Bit	84
Command Error Bit	85
Configuration Error Bit	85
Command Error Conditions	
By Command	85
±Jog Move Command	85
±Blend Move Command	86
Registration Move Command	86
Encoder Follower Operations	86

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 ANF1 One Axis Servo/Stepper Controller Module, and the ANF2 Two Axis Servo/Stepper Controller Module from AMCI. It is written for the engineer responsible for incorporating the ANF1/2 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 Units

This manual applies to all ANF1 and ANF2 modules, including those that have an integral network connection. If your ANF1/2 module has a network interface, such as the ANF1E, you will have to refer to the appropriate AnyNET-I/O Network Interface manual for information on connecting the module to your network. These manuals can be found in the PDF document section of our website at www.amci.com/documents.asp.

NOTE  The AnyNET-I/O product line is constantly evolving. Check our website, www.amci.com for the latest information on available modules and network interfaces in the AnyNET-I/O line.

Trademark Notices

The AMCI logo and “AnyNET-I/O” are trademarks of Advanced Micro Controls Inc. “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, 940-0A100, is the first release of this manual. It was released on July 17th, 2015.

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 manuals' 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.

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 ANF1 and ANF2</i>	Intended for anyone new to the ANF1/2, 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 in the ANF1/2 and the various move types available.
3	<i>Interpolated Move Profiles</i>	Information on how to define interpolated moves in the ANF2 and the various move types available.
4	<i>Homing the ANF1/2</i>	For some applications, it 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 give detailed information on how the ANF1/2 can help you define this position.
5	<i>Installing the ANF1/2</i>	Information on physically installing the ANF1/2 in you system. Network configuration is covered is a separate manual.
6	<i>Configuration Mode Data Format</i>	This chapter explains how to configure the ANF1/2 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 ANF1/2
8	<i>Tabulated Command and Error Codes</i>	This chapter lists the bits used by the ANF1/2 to signal error conditions and the reasons why the bits may be set.

INTRODUCTION TO THE ANF1 AND ANF2

This manual is designed to get you quickly up and running with the ANF1 or ANF2 Servo/Stepper Controller Module. It is possible to purchase either of these modules with or without a network interface. This manual only covers the functionality unique to the ANF1 or ANF2. Information on connecting to the network interface is available in the appropriate AnyNET-I/O Network Interface manual available on the AMCI website.

AnyNET-I/O

The ANF1 and ANF2 are an expansion to the AnyNET-I/O product line from AMCI. The concept of this product line is simple: specialty and/or high speed I/O that can be attached to any popular industrial network; hence the name *AnyNET-I/O*.

AnyNET-I/O is designed for a broad range of applications, from small machines with a single control enclosure, to large machines that use distributed I/O extensively to minimize wiring costs.

What makes the AnyNET-I/O line different is that modules are available with or without a network interface. Eliminating the need for a separate networking module lowers the total cost of ownership for all applications, but especially for the cost sensitive small machines that only require one or two sophisticated functions.

Like many modern controllers, AnyNET-I/O modules are designed to be DIN rail mounted. Up to six AnyNET-I/O modules can be stacked together and accessed over a single network interface. "Stacking" is accomplished through a small backplane connector that snaps into the DIN rail before the AnyNET-I/O modules are installed. These connectors allow the AnyNET-I/O modules to communicate with each other. To the network, the stack of modules appear as one continuous block of I/O words. The logical position within the stack is determined by a DIP switch setting on the front of the module.



Figure 1.1 AnyNET-I/O Module Stack ANF1E (left) & ANF2 (right)

Module Overview

The ANF1 and ANF2 are the first servo/stepper controllers for the AnyNET-I/O product line. The modules offer one or 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. The ANF2 can synchronize its two axes, giving you the ability to control linear and circular motions in an XY plane.

The ANF1 requires ten input words and ten output words from the host controller while the ANF2 requires twenty of each.

Each axis of the ANF1/2 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.

Module Overview (continued)

The ANF1/2 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 your 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 ANF1/2 for open loop position feedback or in *Encoder Follower Mode*. In Encoder Follower Mode, the ANF1/2 will generate pulses on the outputs in response to changes on the encoder inputs. This allows you to use the ANF1/2 in electronic gearing applications. The ANF1/2 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 ANF1/2.
- Home Input. This single ended DC 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 ANF1/2 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 ANF1/2 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.
- Three Digital Outputs per axis. These single ended outputs allow you externally monitor the status of the module. They can also be monitored and controlled by the host controller through network data.

Front Panel Status LED's

Each axis has a bi-color red/green status LED on the front of the module.

- **Steady Green:** Module OK, no motion on the axis is in progress.
- **Blinking Green:** Module OK, motion on the axis is in progress.
- **Blinking Red:** Configuration Error, Command Error, Input Error, Home Invalid Error, Invalid Profile Error.
- **Steady Red:** Communications failure.
 - AnyNET-I/O Stack has lost communication with the host controller. (Ethernet based networks only)
 - Communications error between modules in the AnyNET-I/O Stack.
 - Communications error between the main processor and the communications co-processor on the AnyNET-I/O module that is acting as the network adapter

You must cycle power to the module to attempt to clear this fault.

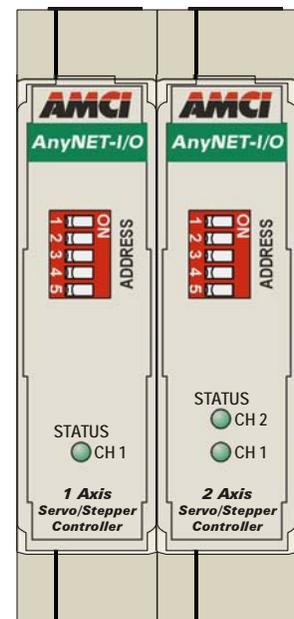


Figure 1.2 Status LED's



Modes of Operation

Each axis of the ANF1/2 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. It also allows you to 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 the axis. The second axis of an ANF2 is disabled by default.
- 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 ANF1/2 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 ANF1/2. 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.

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 3, *Interpolated Move Profiles*, starting on pg 27, 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.

Digital Outputs

There are three DC digital output per axis on the ANF1 and ANF2 modules. The state of the outputs can be controlled and monitored by the host controller.

All outputs are open collector sinking. An output is connected to the Output_Common pin when the output is active. The Output_Common pin must be connected to the output power supply common for operation. The output power supply can be within the range of 5 to 24 Vdc.

An output is "active" when it is conducting current. The output state is reported to the host controller as a "1" when the output is active. When measuring the output voltage on the pin, it will be near zero volts when the output is active.



Digital Outputs (continued)

No_Fault Output: (Axis 1 and 2). This output can only be set to its active state by the host controller. Once active, this output will be turned off by the following conditions:

- Input Error during a move. (E-Stop, CW Limit, or CCW Limit Input activated during move.)
- Axis switched from Command to Configuration Mode
- Loss of network communications
- Reset command sent by host controller.

Move_Complete Output: (Axis 1 and 2) This output will be set to its active state by the following conditions:

- Absolute, Relative, Blend, or Interpolated Move completes without error.
- Set command sent by host controller.

Once active, this output can only be reset to its inactive state by the host controller.

On_Line Output: (Axis 1 only) This output can only be set to its active state by the host controller. The On_Line output will be turned off by the following conditions:

- Loss of network communications
- Reset command sent by host controller.

General Purpose Output: (Axis 2 only) The on/off state of the General Purpose output is only controlled by the host controller.

Set_State Bit

Some host controllers will continuously write new data to the ANF1/2. Because of this, the ANF1/2 implements a “Set_State” bit. Each axis has its own Set_State bit. Output states can only be changed by the host controller when this bit equals “1”.

NOTE

- 1) All three outputs associated with the axis are updated when the associated Set_State bit equals “1”. Your host controller logic must take this into consideration when updating the state of the outputs.
- 2) With the exception of the General Purpose output, the Set_State bit cannot be used to force the outputs on or off. When a No_Fault, Move_Complete, or On_Line event occurs, the outputs will change to their designated states for a maximum of five milliseconds before returning to the states specified by the host controller.
- 3) The Set_State bits should be reset during a move operation. This will prevent the host controller from overriding the No_Fault and Move_Complete outputs while a move is occurring.



Specifications

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

Three per Axis

Output state controlled by host controller. Outputs will also transition on move complete, input error, or on-line fault events.

Encoder Inputs

One Encoder per Axis

+5Vdc differential ABZ inputs

X4 decoding always used by ANF1/2

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.

Number of I/O Words per Module

ANF1: 10 Input Words, 10 Output Words

ANF2: 20 Input Words, 20 Output Words

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.

Digital Outputs

Same electrical specifications as the Single-Ended Stepper Outputs

Digital Inputs

Open Collector Sinking

All inputs share a common return

ON Voltage Range: 8Vdc to 24Vdc

OFF Voltage Range: 0Vdc to 5Vdc

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.

ANF1/2 must be configured on power up.

Power Supply

10 to 28 Vdc, surge to 30Vdc without damage to module. 120 mA @ 24Vdc (2.9W)

Environmental Conditions

Operating Temperature: -4 to 122°F (-20 to 50°C)

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

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

Connectors

Mating connectors are included with the ANF1/2 and are available separately under the following AMCI part numbers.

Connector	AMCI Part #	Wire	Strip Length	Min. Tightening Torque
I/O	MS-2X11	28 - 16 AWG	0.275 inches	Spring Cage Connector
Power	MS-4M	28 - 12 AWG	0.394 inches	4.43lb-in (0.5 Nm)
Backplane	IC-5	Not Applicable		



Compatible Equipment

Servo Equipment

As of the release of this manual, the ANF1/2 has been tested with the Rockwell Automation Ultra 1500 and Ultra 3000 servo drives. Due to its differential and single ended outputs, the ANF1/2 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 Drivers

AMCI has a broad range of stepper motors that are compatible with the ANF1/2.

Product Family	Description
SD7540A 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.

Table 1.1 AMCI Compatible Equipment

Stepper Motors

AMCI has a broad range of stepper motors that are compatible with the above AMCI drivers.

Product Family	Description
'SM' Stepper Motors	A line of NEMA size 23 and 34 motors with holding torques in the range of 130 to 1,175 oz-in. They are all available with or without optical incremental 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 ANF1/2 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.

When a move command is sent to the ANF1/2, 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 ANF1/2, 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 ANF1/2.
- 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 ANF1/2 to the value used in the equations.

Definition of Count Direction

Each axis of the ANF1/2 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.

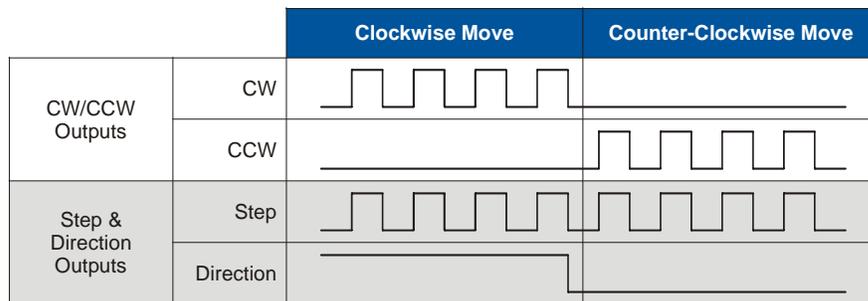


Figure 2.1 Directional Outputs



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 ANF1/2. 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 ANF1/2 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 ANF1/2 module has a "Position Valid" bit that is set to "1" when the Home Position has not yet been defined or has been lost due to an uncontrolled stop condition. This bit will equal "0" when the Home Position has been defined. This bit must equal zero before absolute moves can be run.
- 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 ANF1/2 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.

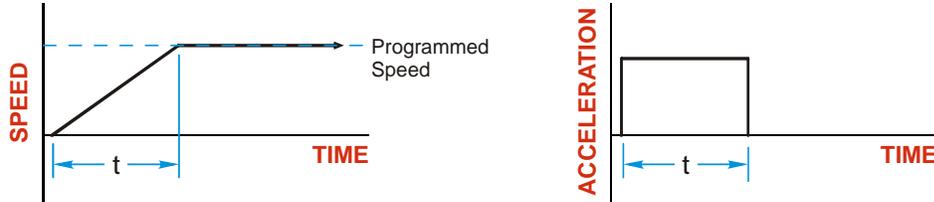


Figure 2.2 Linear Acceleration

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.

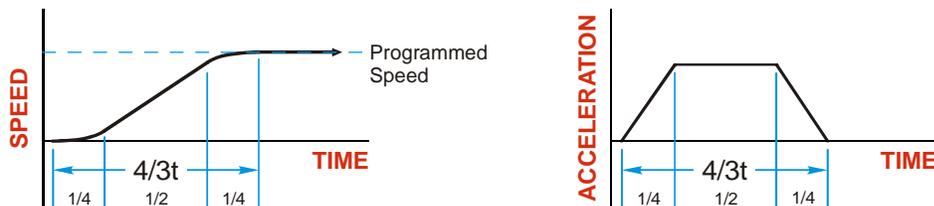


Figure 2.4 Trapezoidal S-Curve Acceleration



A Simple Move

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

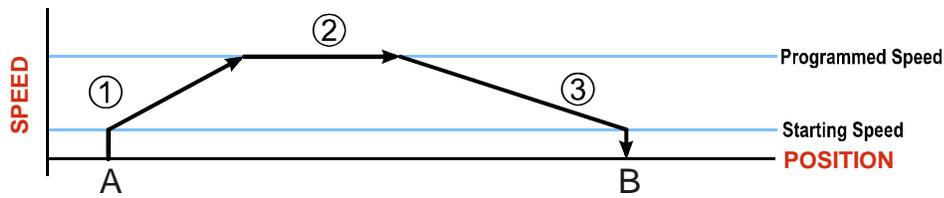


Figure 2.5 A Trapezoidal Profile

- 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 ANF1/2.
- 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 4, [Homing the ANF1/2](#), 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 ANF1/2 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.

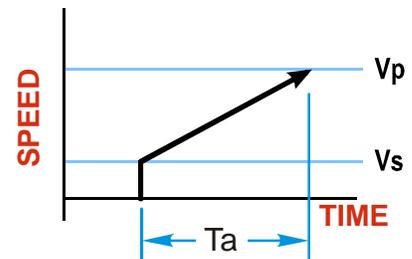
The equations in the rest of this chapter use a unit of measure of steps/second/second (steps/second²) for acceleration and deceleration. However, when programming the ANF1/2, all acceleration and deceleration values must be programmed in the unit of measure of steps/millisecond/second.

- 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 ANF1/2.
- 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 ANF1/2 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** = 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:

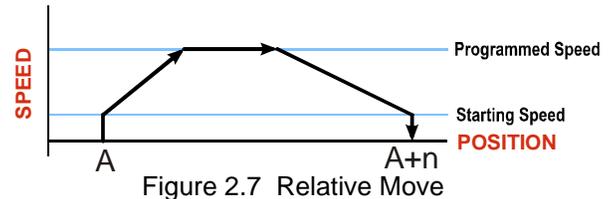
$$T_T = T_A + T_D$$

Available Move Types

This section describes the six single-axis move types available with the ANF1/2. 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.



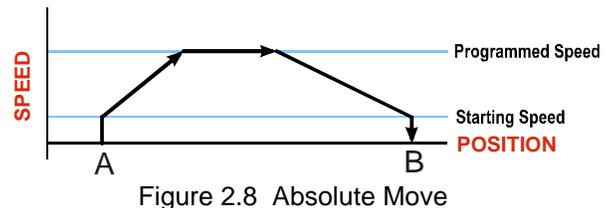
The command's Target Position registers hold the move's offset. The offset can be in the range of $\pm 1,073,741,823$ counts. Positive offsets will result in clockwise moves, while negative offsets result in counter-clockwise moves.

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 ANF1/2 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 ANF1/2 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 ANF1/2 calculates the number of steps and direction needed to move to the given position and moves that number of steps.) If position (B) is greater than (A), the move will be clockwise. A trapezoidal profile is shown to the right, but Absolute Moves can also generate triangular profiles.



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 ANF1/2 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 ANF1/2. A complete description of how to use the Find Home commands is given in chapter 3, *Homing the ANF1/2*, 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 ANF1/2 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 ANF1/2 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 ANF1/2 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 ANF1/2 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 ANF1/2 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 ANF1/2 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.

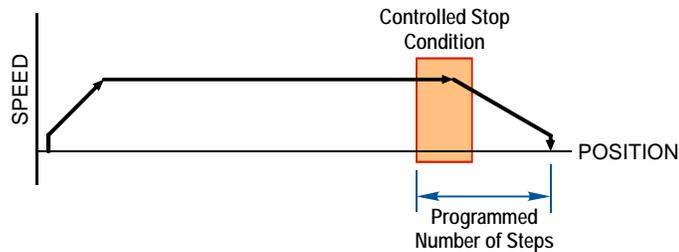


Figure 2.9 Registration Move

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 ANF1/2 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 and the minimum registration distance is set to zero.

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 ANF1/2 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.

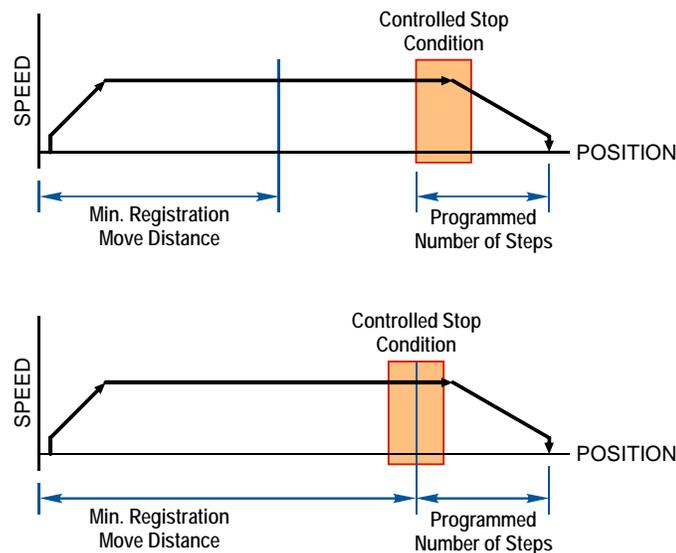


Figure 2.10 Min. Registration Move Distance

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).

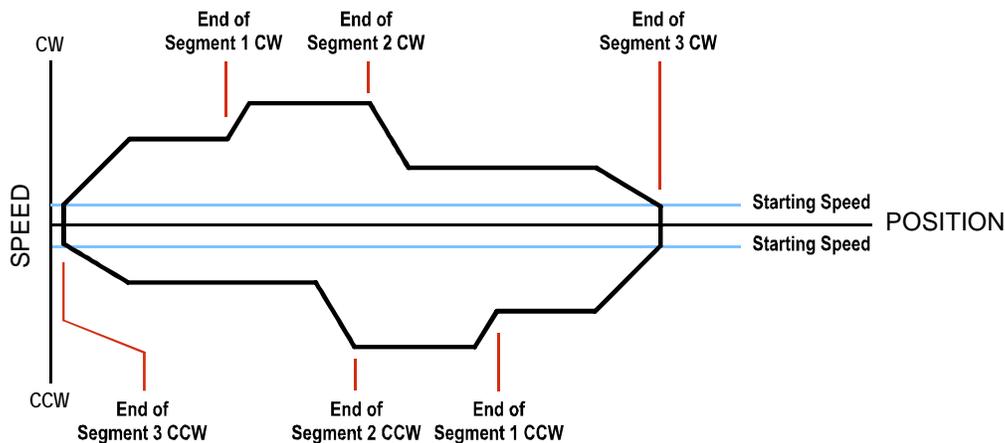


Figure 2.11 Blend Move Profile

Each segment is programmed with a programming block sent from the host controller using the Output Network Data 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 ANF1/2. 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.

Available Move Types (continued)

±Blend Move (continued)

It is not possible to program a direction reversal in the Blend Move profile.

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 Acceleration Value associated with the next segment is used to accelerate or decelerate to the segment's Programmed Speed.
- ▶ To end the final segment, a single Deceleration Value is programmed as part of the move. This value is used to decelerate from the final segment's Programmed Speed to the Starting Speed and then the move stops.

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 Move

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 ANF1/2 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 ANF1/2 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 ANF1/2 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 ANF1/2 will output multiple steps per encoder count.

As with all encoder functions on the ANF1/2, 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 ANF1/2 will output pulses at its 1MHz rate until motor position catches up with the encoder position.

**Available Move Types (continued)****±Encoder Follower Move (continued)**

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 ANF1/2 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.

This chapter introduces the three interpolated move types available with the ANF2. 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.

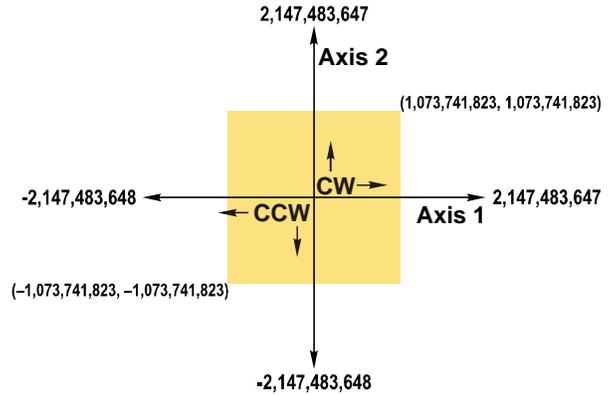


Figure 3.1 X-Y Plane

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 ANF2.

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.

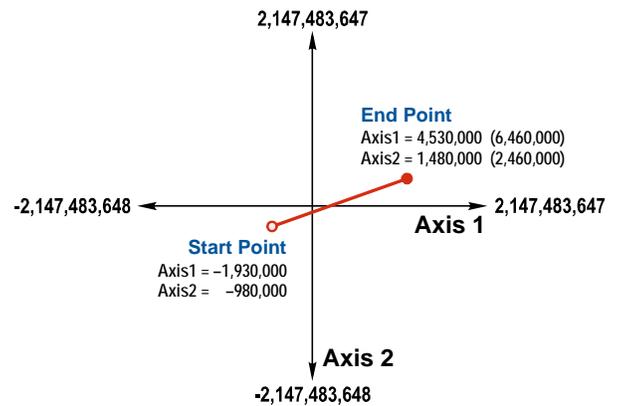


Figure 3.2 Linear Interpolated Move

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 ANF2 calculates the required component vector values for each axis before beginning the move.

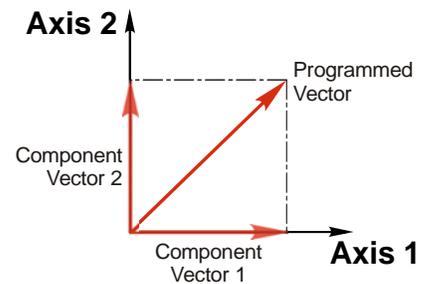


Figure 3.3 Programmed Vectors

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 ANF2 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 ANF2 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.

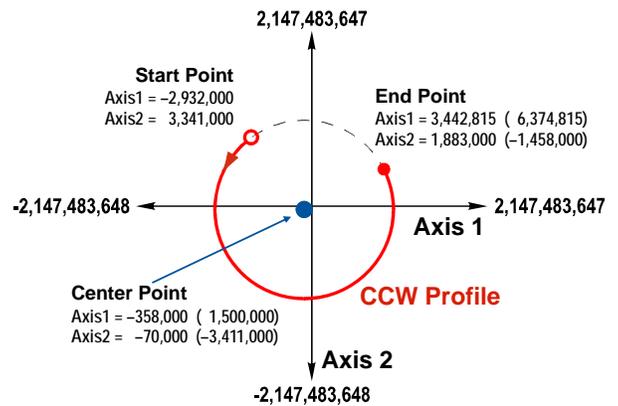


Figure 3.4 Center Point Circular Move

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 ANF2 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 ANF2 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.

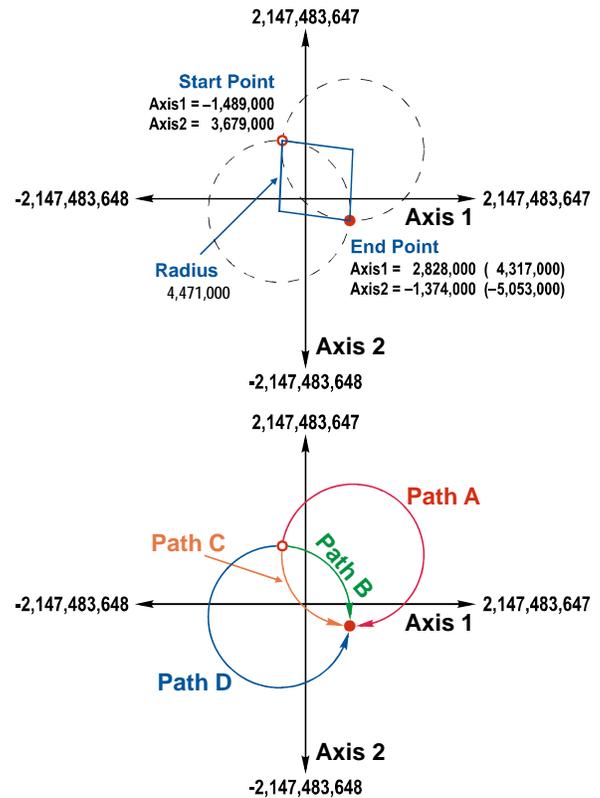


Figure 3.5 Radius Circular Move

Circular Interpolated Moves (continued)

Via Point Method

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 ANF2 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.

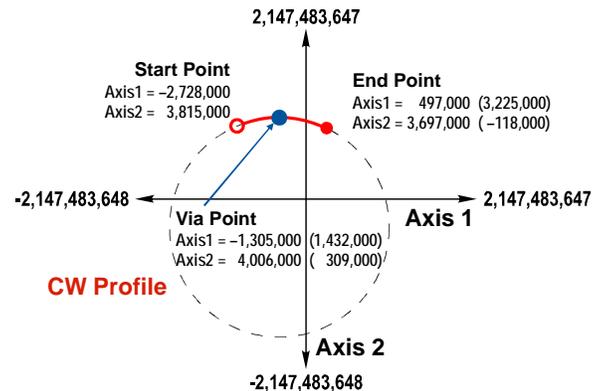


Figure 3.6 Via Point Circular Move

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 ANF2 then calculates the path it needs to take before actually starting the move. For Linear Interpolated Moves, the ANF2 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 ANF2 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 ANF2 generates an error message to the host controller 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 ANF2 to integers before being compared. The ANF2 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 ANF2. 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 ANF2 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. The 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 who require you to recalculate the Center Point to Start Point radius because is will also change.

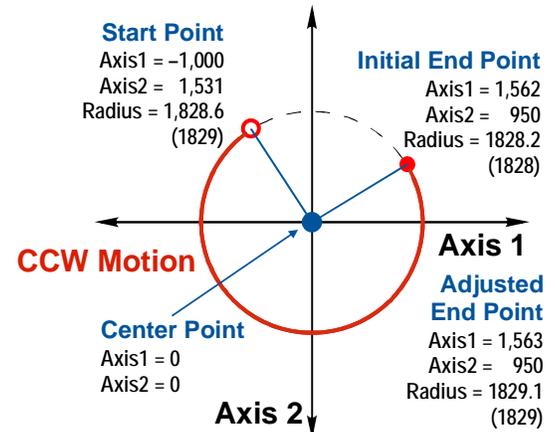


Figure 3.7 Round Off Error Example

HOMING THE ANF1/2

This chapter explains the various ways of homing an axis on the ANF1/2. 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 ANF1/2 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.

NOTE  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 ANF1/2 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 ANF1/2 axis. Before doing this, you will need a way of sensing position outside the ANF1/2 module. One possibility is by using an AMCI ANA2 AnyNET-I/O Resolver Interface Module to sense the absolute position of the machine. Another possibility is an ANE2 AnyNET-I/O SSI Interface Module and an SSI sensor. In either case, the position data must be brought into the host controller, the correct preset value calculated, and this value written to the ANF1/2 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 ANF1/2 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 ANF1/2 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 ANF1/2 to use an encoder, you have the option of using the encoder's marker pulse to home the ANF1/2.
- **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 ANF1/2 can be configured to ignore changes on the physical homing input until the Home Proximity Bit makes a 0→1 transition. The ANF1/2 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 ANF1/2 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 ANF1/2 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.

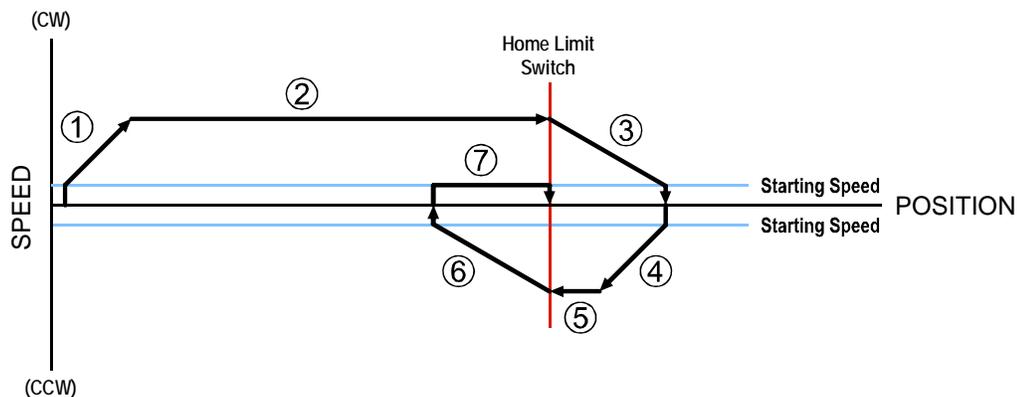


Figure 4.1 Home Input Profile

- 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

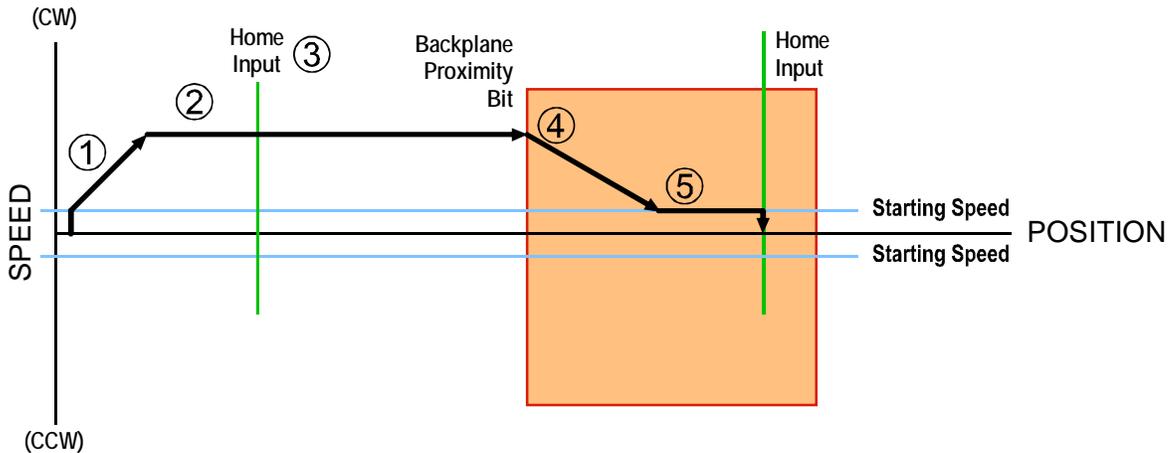


Figure 4.2 Homing with Proximity

- 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 ANF1/2 reaches its home position. This is valid, but does not have to occur. As stated in step 4, the ANF1/2 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 ANF1/2 will stop the axis with an error if both overtravel limits are activated while the ANF1/2 is trying to find the home position.

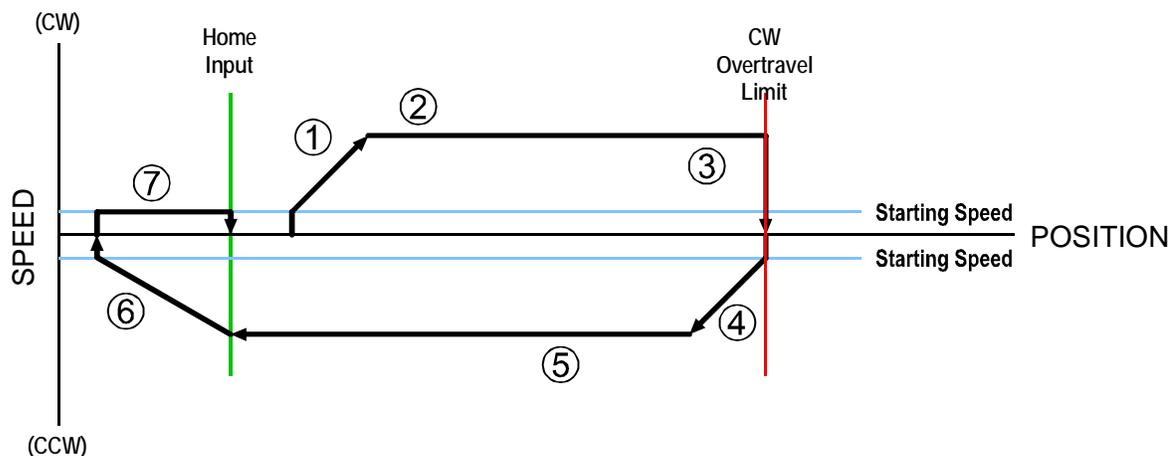


Figure 4.3 Profile with Overtravel Limit

- 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.

This chapter contains information on properly handling the ANF1/2 as well as mounting the module in a safe environment. This chapter also covers how to wire your field I/O to the ANF1/2.

Module Installation

Module Location

AnyNET-I/O modules are suitable for use in industrial environments 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. 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 an AnyNET-I/O module if you touch the rear 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 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. The DIN rail for the modules should be securely installed and grounded before the modules are mounted on it.

Remove Power Before Servicing in a Hazardous Environment



Remove power before removing or installing any modules in a hazardous environment.

Module Installation (continued)

Module Mounting

Dimensions

The figures below show the dimensions of the ANF1 and ANF2 AnyNET-I/O modules. These modules are low power modules that do not require any additional spacing when mounting the unit. Refer to the installation instructions of the appropriate AnyNET-I/O network interface module for complete information on spacing needed to install the module.

NOTE ⚡ You may need to ground the sensor cable shields at the module. There is an earth ground pin on the power connector, but if you have multiple shields, it will be better to ground them to the DIN rail. If you decide to do this, make sure your DIN rail is long enough to mount the AnyNET-I/O modules and ground the cable shields.

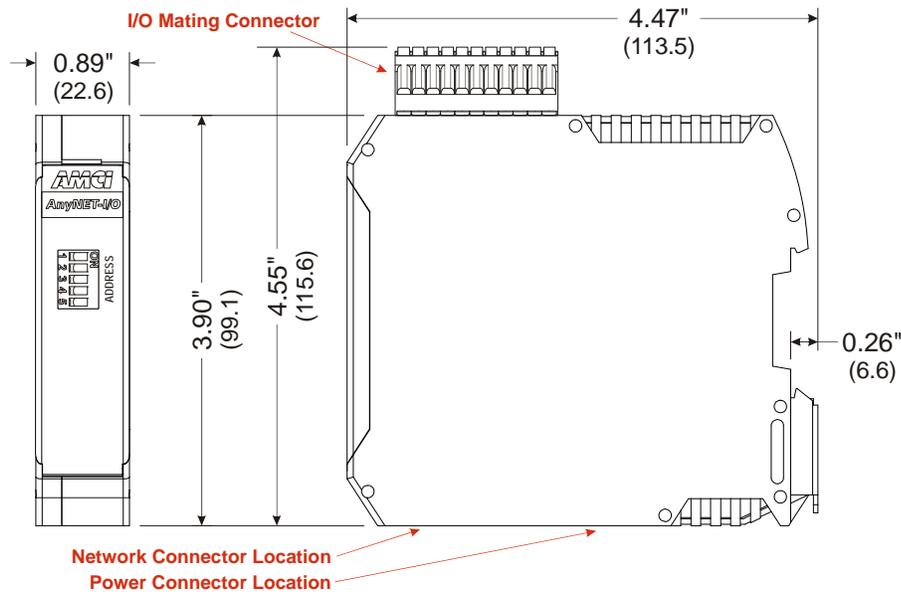


Figure 5.1 ANF1 Outline Drawing

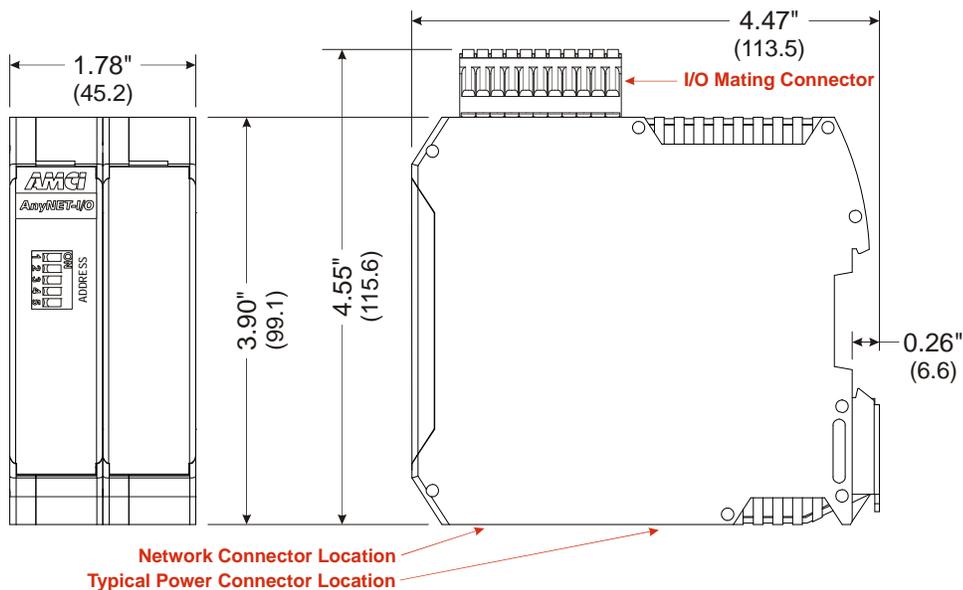


Figure 5.2 ANF2 Outline Drawing

Module Installation (continued)

Module Mounting (continued)

Compatible DIN Rail

The ANF1/2 can be mounted on following DIN rail:

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

If you are only installing one ANF1/2 module instead of an interconnected stack, then you can also use the following DIN rail:

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

NOTE The EN 05 022 - 35 x 15 DIN rail is unacceptable when installing multiple AnyNET-I/O modules as a stack. The IC-5 connector that is used to interconnect the modules requires the EN 05 022 - 35 x 7.5 DIN rail for proper operation.

The DIN Rail must be securely mounted to a panel and solidly grounded before the module is installed. Grounding is usually accomplished through the mounting hardware, by first removing any paint or other material from all surfaces that may interfere with proper grounding. Another option is to install a heavy gauge wire from the DIN rail to your system's ground bus.

Installing IC-5 Connectors

IC-5 connectors must be used when more than one module is installed as an interconnected stack. The ANF1 uses one IC-5 connector, while the ANF2 uses two IC-5 connectors. Note that a single ANF2 module does not need IC-5 connectors if it is the only module. Figure 5.3 shows how to install the IC-5 connectors in the DIN rail.

NOTE

- 1) EN 05 022 - 35 x 7.5 DIN rail must be used. The IC-5 connectors are not properly supported in EN 05 022 - 35 x 15 DIN rail
- 2) Note the orientation of the IC-5 connectors when installing them. The module key goes towards the bottom of the DIN rail.

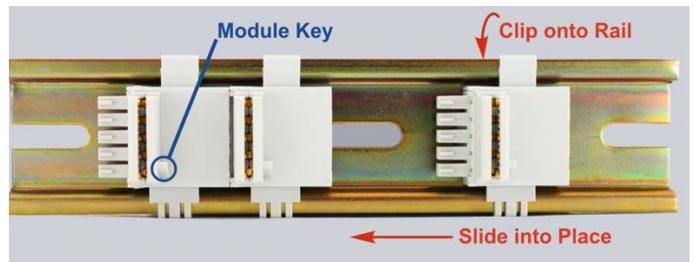


Figure 5.3 IC-5 Connector Installation

Mounting the AnyNET-I/O Module

Mounting an AnyNET-I/O module is a very simple process thanks to the design of the enclosure.

- 1) Partially engage the connector into the enclosure.
- 2) Engage the top clip in the enclosure with the top of the DIN rail and rotate the module down until the metal bracket snaps on to the DIN Rail.

Once all of your modules are installed, it is strongly suggested to use the end caps from Phoenix Contact with the part number of 271 37 80 to secure the modules on the DIN Rail. These end caps prevent the module from sliding along the DIN rail if it is subjected to shock or vibration during machine operation.

Module Installation (continued)

Stack Addressing

Each module in a stack needs to be given an address before the system will operate correctly. The address is set with the five position DIP switch on the front of the module.

NOTE

- 1) The module with the network interface for the stack must have all of its DIP switches off. (This is called address 0.) This is the only module in the stack that can have a direct network connection.
- 2) Only a single switch should be in the "ON" position when setting the address.
- 3) If a module has a network interface, this interface is disabled if the module has a non-zero address. This allows two modules with network interfaces to work in a single stack.

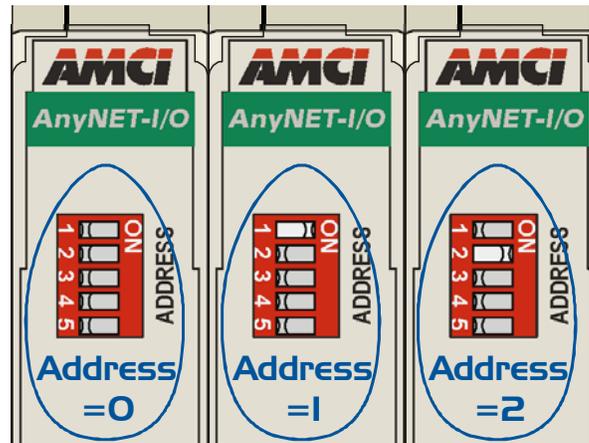


Figure 5.4 Addressing Example

Figure 5.4 is a close up of three modules in an AnyNET-I/O Stack. The module on the left is an unit with a network interface, and has an address of zero (All DIP switches off.) This module has the active network interface and connects the stack to the network. Reading left to right, the remaining modules have addresses of one and two respectively. These modules can be any type of AnyNET-I/O modules. If they have network interfaces, these interfaces are disabled.

Powering the Module

Required Power

The ANF1/2 requires 12 to 24 Vdc \pm 20% supply to power the module. Surges to 30 Vdc will not damage it. Current draw is 120 mA @ 24 Vdc (2.9W).

This current draw does not include current through the inputs or outputs. If using the same supply to power the I/O, their load must be considered when sizing the supply.

- Inputs require 10 mA each.
- Differential outputs can supply up to 20 mAdc each.
- Single ended outputs can sink up to 30 mAdc each.

Powering the Module (continued)

Wiring

As shown in the figure below, the power connector is located on the bottom of the module. An ANF1 is shown in the figure. The ANF2 is a dual width module with one power connector. The mating connector is included with the ANF1/2. Spares are available from AMCI under the part number MS-4M. They are also available from Phoenix Contact under their part number 187 80 37.

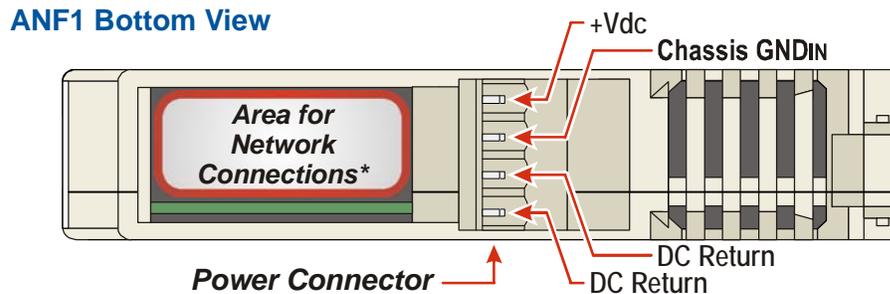


Figure 5.5 Power Connector Location

Power connections should be tight, as loose connections may lead to arcing which will heat the connector. Phoenix Contact specifies a tightening torque of 4.4 to 5.4 lb-in (0.5 to 0.6 Nm).

The power supply is connected to the pins marked “+Vdc In” and “DC Return”. The “Chassis GND” pin is used to attach the ANF1/2 to earth ground.

NOTE

- 1) AnyNET-I/O modules are electrically isolated from the DIN rail by their mounting, but the Chassis GND connection is common to all of the modules in the stack through a pin of the IC-5 connector. At least one module in the AnyNET-I/O Stack must be attached to earth ground through a heavy gauge stranded wire to ensure reliable operation of the stack.
- 2) Each ANF1/2 module must have its own power connection.
- 3) AnyNET-I/O modules will not be damaged if it is plugged into the InterConnect bus while power is applied, but the stack will not reconfigure itself to accept the new module.

I/O Connector Pin Out

Figure 5.6 below shows the pin out of the I/O connectors on the ANF1 and ANF2 modules. Note that the dual width ANF2 is shown. The ANF1 is a single width module that only has the J1/2 connector. All mating connectors are included with the module. Additional connectors can be order for AMCI under the part number MS-2x11.

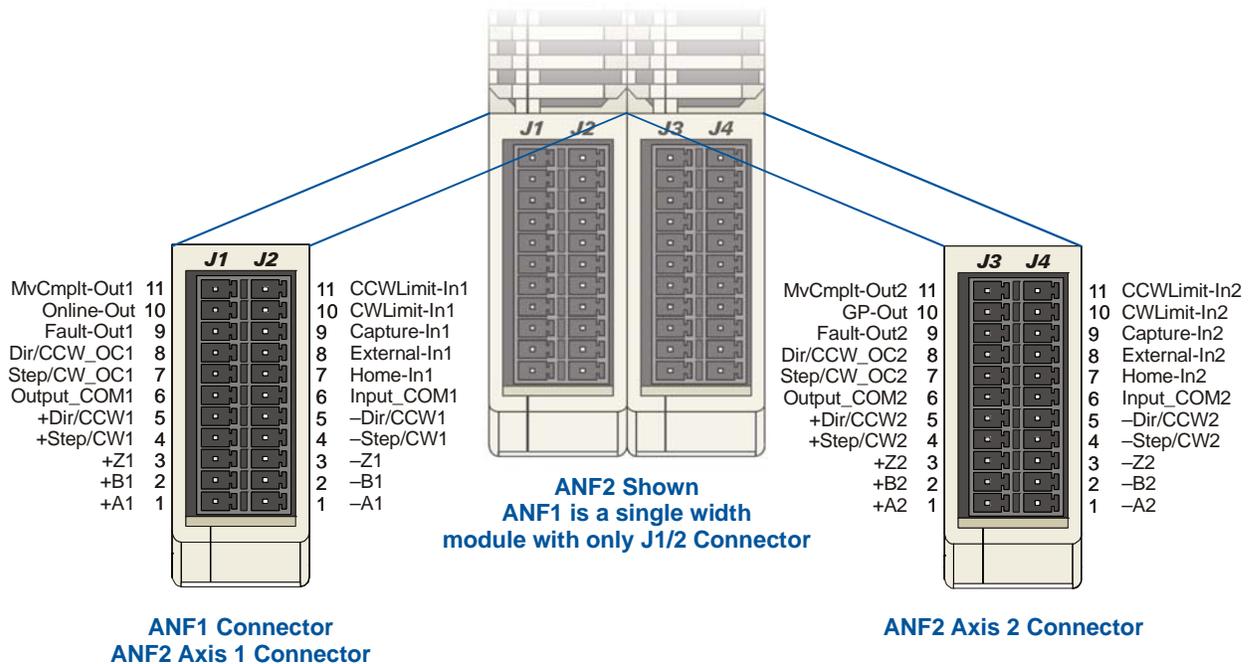


Figure 5.6 I/O Connector Pin Out

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.

Differential Output From ANF1/2

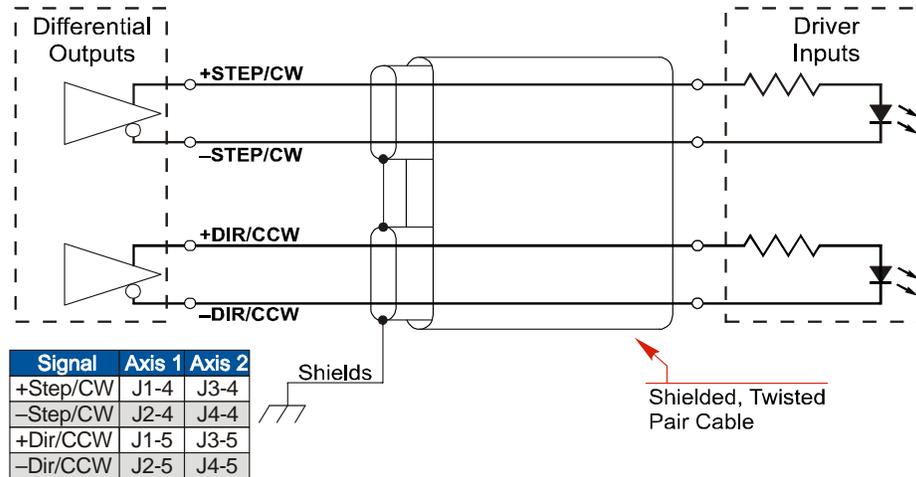


Figure 5.7 Differential Driver Wiring

Open Collector Wiring

All singled ended outputs in the ANF1/2 are open collector sinking. They share a single common pin that must be tied to the ground connection of the power supply powering the outputs. Figure 5.8 below shows how to wire the outputs to open collector sourcing inputs on your driver.

ANF1/2 Output Connection to Sourcing Load

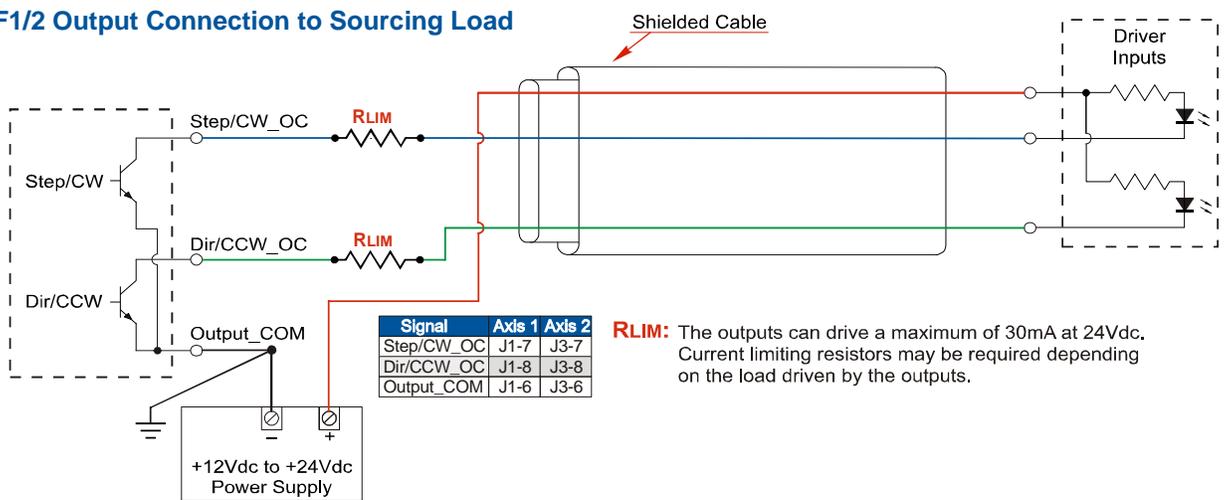


Figure 5.8 Open Collector Driver Wiring (Sourcing Inputs)

NOTE RLIM may be needed to limit current flow. The ANF1/2 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 ANF1/2 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 ANF1/2 output is in a logic “1” state.

ANF1/2 Output Connection to Sinking Load

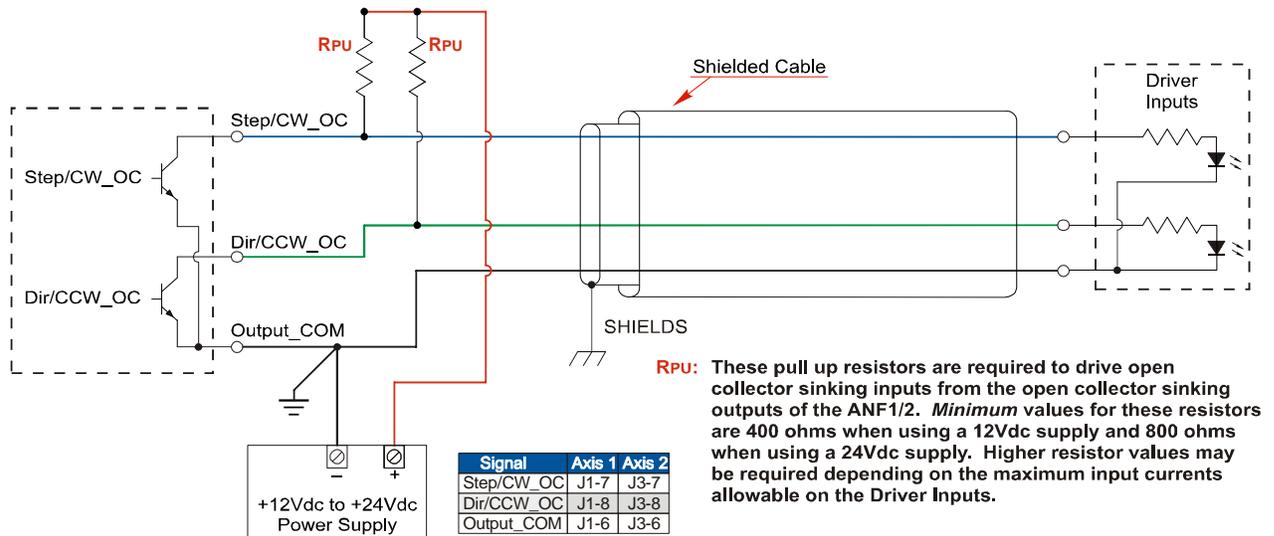


Figure 5.9 Open Collector Driver Wiring (Sinking Inputs)

CAUTION Carefully select the value of the pull up resistors. (RPU in the figure.) The ANF1/2 outputs can conduct a maximum of 30 milliamperes 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.

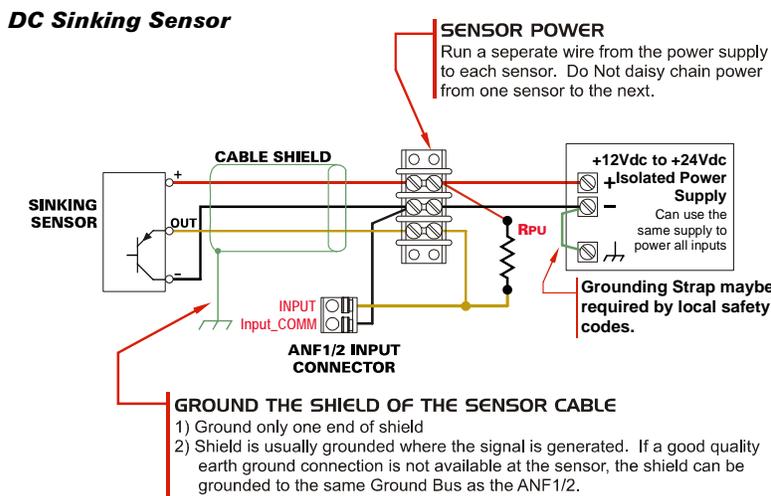
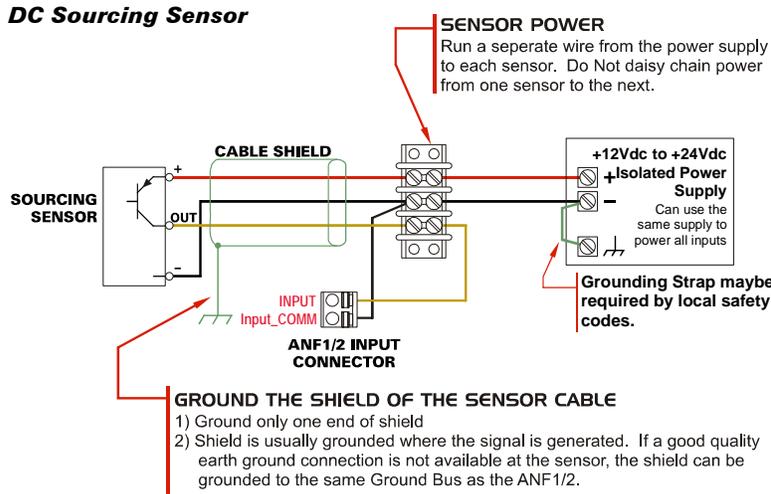
Digital Output Wiring

Each axis has three digital outputs. Both axes have Move_Complete and Axis Fault outputs. Axis 1 has an Online output, while axis 2 has a General purpose output. These outputs are electrically identical to the open collector Step/CW_OC and Dir/CCW_OC outputs available on each axis. These digital outputs share a common ground pin with the driver outputs. Wiring is identical to the diagrams given in figures 5.8 and 5.9.

NOTE The digital outputs can, but do not have to, use the power supply that powers the open collector driver outputs. If you use two different supplies, the low side of the supplies must be tied together at pin J1-6 for axis 1 and J3-6 for axis 2. 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

Figure 5.10 below shows how to wire discrete DC sourcing and sinking sensors to the open collector sinking inputs of the ANF1/2.



Input Pins

ANF1 ANF2 Axis 1		ANF2 Axis 2	
Signal	Pin	Signal	Pin
Input_COM1	J2-6	Input_COM2	J4-6
Home-In1	J2-7	Home-In2	J4-7
External-In1	J2-8	External-In2	J4-8
Capture-In1	J2-9	Capture-In2	J4-9
CWLimit-In1	J2-10	CWLimit-In2	J4-10
CCWLimit-In1	J2-11	CCWLimit-In2	J4-11

Figure 5.10 Input Wiring

Input Specifications:

Single Ended. Accepts 8 to 27Vdc without the need for an external current limiting resistor.

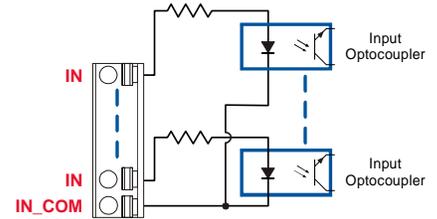


Figure 5.11 DC Input Schematic

Because they are low power signals, cabling from the sensor to the ANF1/2 should be done using a twisted pair cable with an overall shield. The shield should be grounded at the end when the signal is generated, which is the sensor end. If this is not practical, the shield should be grounded to the same ground bus as the ANF1/2.

Sinking Output Sensors Require a Pull Up Resistor

Sinking output sensors require an external pull up resistor because the input to the ANF1/2 also sinks current. Table 5.1 below shows the values of pull up resistors that will allow the ANF1/2 input to activate along with the current that the sensor must be able to sink when it is active.

Input Voltage	Pull Up Resistor	Sensor Current When Active
12	1.4K	8.6mA
24	3.8K	6.3mA

Table 5.1 Pull Up Resistor

The logical states of the sensor and ANF1/2 input will be reversed. The ANF1/2 input is off when the sensor is active. You can set the logic state of the ANF1/2 input when you configure the unit.

Encoder Wiring

Differential Encoder Wiring

The encoder inputs on the ANF1/2 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 5.12 below shows a typical encoder installation using an AMCI optical encoder.

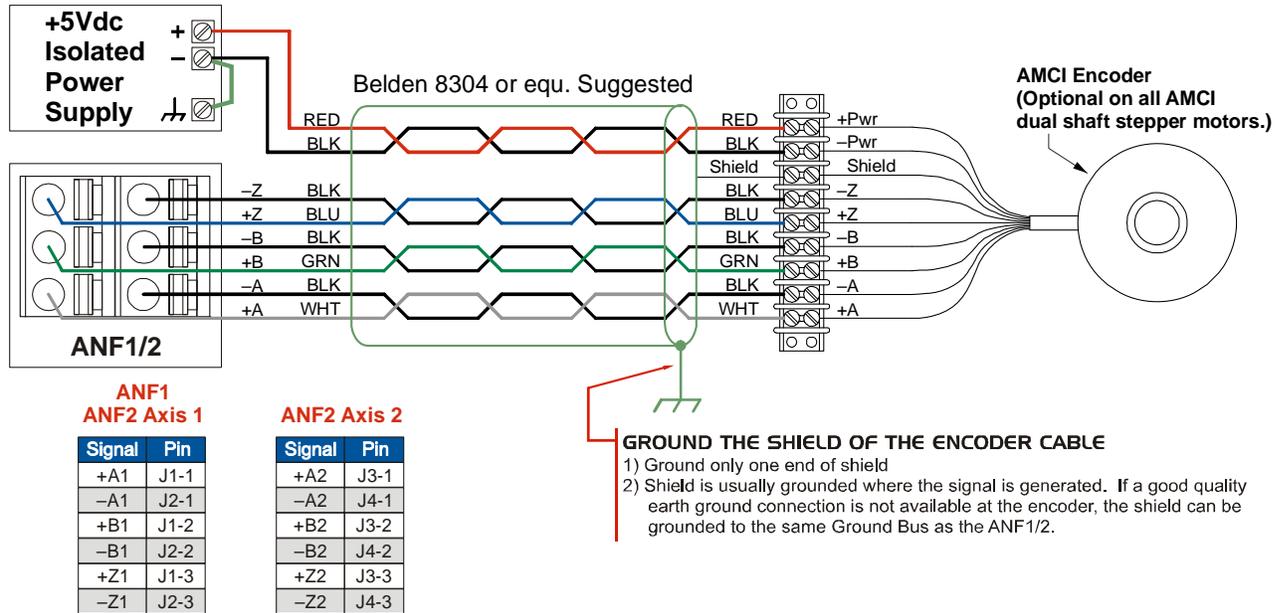


Figure 5.12 Differential Encoder Wiring

The table below lists the wire color code for the AMCI encoder used at the time this manual was released. These encoders have an eighteen inch integral cable. 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.

Function	Wire Color
+Pwr (+5 Vdc)	Red
-PWR (Gnd)	Black
+A	Brown
-A	White
+B	Blue
-B	Green
+Z	Orange
-Z	Yellow

Table 5.2 AMCI Encoder Wire Color Chart

Encoder Wiring (continued)

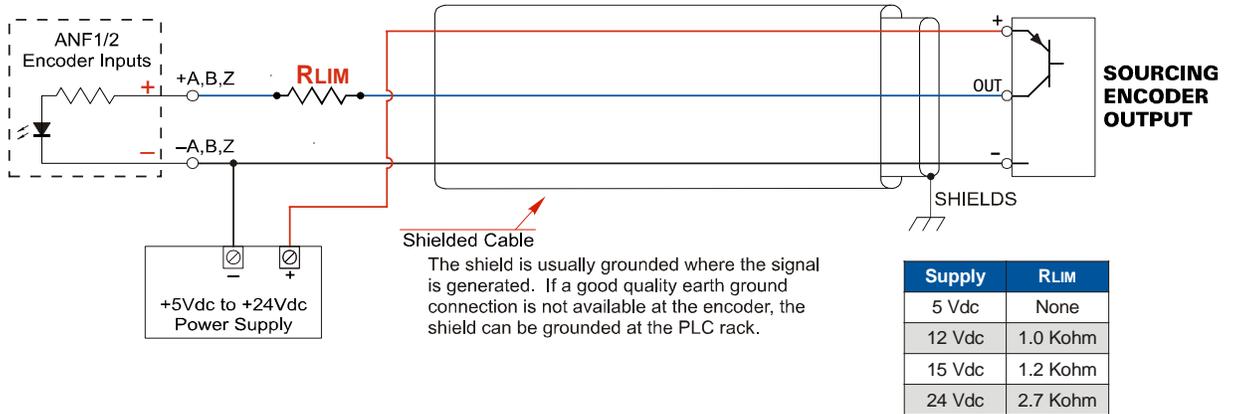
Single Ended Wiring

Figure 5.13 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 ANF1/2 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.

ANF1/2 Encoder Input Connection to Sourcing 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.

ANF1/2 Encoder Input Connection to Sinking Encoder Output

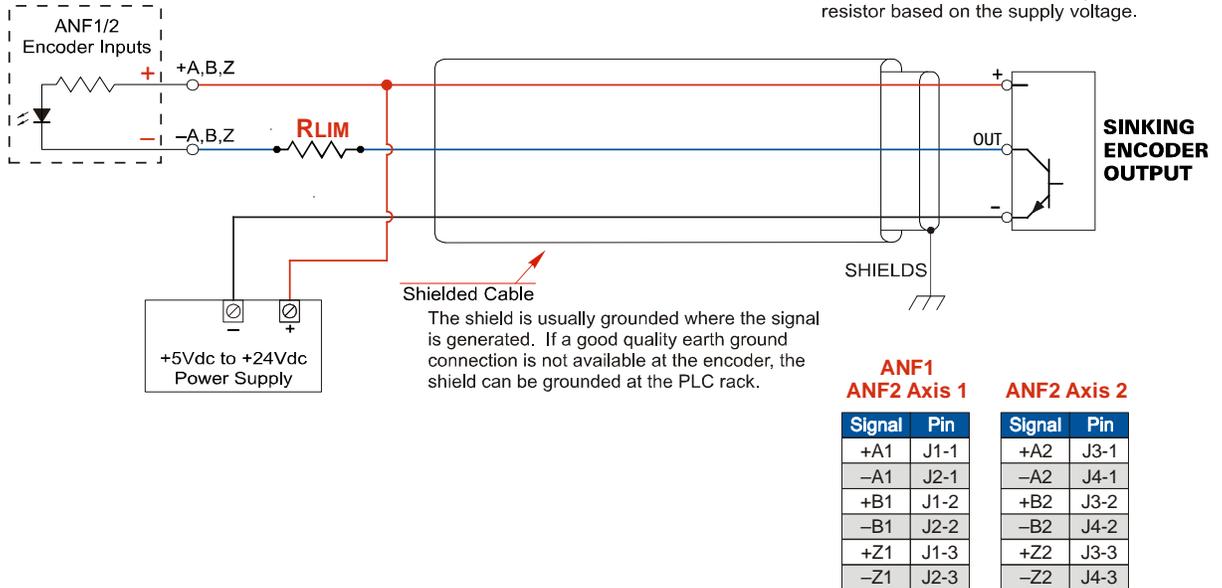


Figure 5.13 Single Ended Encoder Wiring

Diagnostic Feedback Wiring

The ANF1/2 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	+Step1	J1-4	J1-1	+A1
1	-Step1	J2-4	J2-1	-A1
1	+Direction1	J1-5	J1-2	+B1
1	-Direction1	J2-5	J2-2	-B1
2	+Step2	J3-4	J3-1	+A2
2	-Step2	J4-4	J4-1	-A2
2	+Direction2	J3-5	J3-2	+B2
2	-Direction2	J4-5	J4-2	-B2

Table 5.3 Diagnostic Feedback Wiring

The axis you are testing must be configured to output step and direction pulses. The ANF1/2 will issue a configuration error if the axis is configured to output CW/CCW pulses.

CONFIGURATION MODE DATA FORMAT

Each axis of the ANF1/2 has two modes of operation, Configuration Mode and Command Mode. This chapter describes the format of data written to and read from the axis when it is in Configuration Mode. Output words are used to program the axis, while the input words are used to report back its status. The ANF1 uses ten input and output words. The ANF2 uses twenty input and output words, where the first ten are used to control axis 1 and the last ten words are used to control axis 2.



NOTE The ANF1/2 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 network data words when writing configuration data to both axes of the ANF1/2.

		Output Word	Axis	Function	Units	Range
ANF1	ANF1	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	1	<i>Reserved, Must be 16#0000</i>		<i>16#000</i>
		9	1	<i>Reserved, Must be 16#0000</i>		<i>16#000</i>
ANF2	ANF2	10	2	Configuration Bits MSW		See Description Below
		11	2	Configuration Bits LSW		See Description Below
		12	2	Starting Speed MSW	Steps/Second	1 to 1,000,000
		13	2	Starting Speed LSW		
		14	2	Homing Timeout	Seconds	0 to 300
		15	2	<i>Reserved, Must be 16#0000</i>		<i>16#000</i>
		16	2	<i>Reserved, Must be 16#0000</i>		<i>16#000</i>
		17	2	<i>Reserved, Must be 16#0000</i>		<i>16#000</i>
		18	2	<i>Reserved, Must be 16#0000</i>		<i>16#000</i>
		19	2	<i>Reserved, Must be 16#0000</i>		<i>16#000</i>

Table 6.1 Output Data Format - Configuration Data

Output Data Format (continued)

Configuration Bits MSW

Word 0 configures Axis 1, Word 10 configures Axis 2 on an ANF2

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			Bits 12 & 13 are Reserved. They must always be set to zero.													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:		1		0	0													

Table 6.2 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 ANF1/2 to ignore multiple triggers on the home input.

Output Data Format (continued)

Configuration Bits LSW

Word 1 configures Axis 1, Word 11 configures Axis 2 on an ANF2

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 6.3 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 any low frequency resonance points 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 Network Data. The format is a 32-bit integer with the most significant bits in the first word. The following table shows how to split the Starting Speed parameter value. The ANF1/2 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 6.4 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 ANF1/2. 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 ANF1/2 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 ANF1/2 will search for the home marker until it finds it or the command is aborted by the host controller.

Input Data Format

While in Configuration Mode, the input registers assigned to the ANF1/2 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 network data 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.

	Input Word	Axis	Description
ANF1	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	1	<i>Not used in Configuration Mode, will always be 16#0000</i>
	9	1	<i>Not used in Configuration Mode, will always be 16#0000</i>
ANF2	10	2	Mirror of Configuration Bits MSW, (Bits 0 – 12)*
	11	2	Mirror of Configuration Bits LSW
	12	2	Mirror of Starting Speed MSW
	13	2	Mirror of Starting Speed LSW
	14	2	Mirror of Homing Timeout
	15	2	<i>Not used in Configuration Mode, will always be 16#0000</i>
	16	2	<i>Not used in Configuration Mode, will always be 16#0000</i>
	17	2	<i>Not used in Configuration Mode, will always be 16#0000</i>
	18	2	<i>Not used in Configuration Mode, will always be 16#0000</i>
	19	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 6.5 Input Data Format - Configuration Mode

Global Status Bits

Word 0 contains Status Bits for Axis 1, Word 10 contains Status Bits for Axis 2 on an ANF2

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 ANF1/2. The figure below shows meaning of each of these status bits.

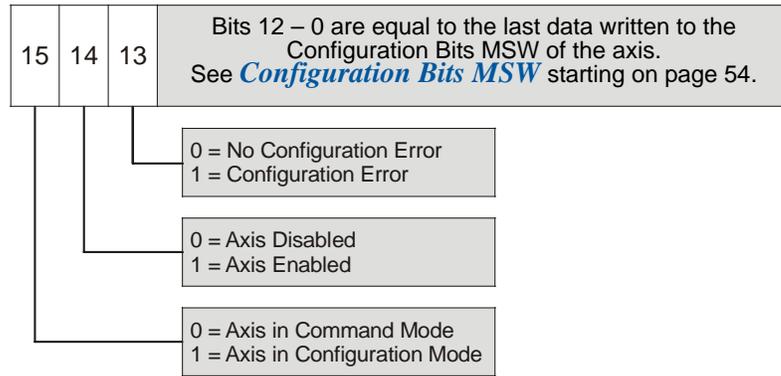


Figure 6.1 Global Status Bits

Invalid Configurations

The following section lists the configuration settings that will cause a configuration error on the axis. Input word 0, bit 13 is the Configuration Error bit for axis 1 and input word 10, bit 13 is the Configuration Error bit for axis 2.

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 10 configures Axis 2 of an ANF2

		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 6.6 Configuration Bits MSW Errors

Invalid Configurations (continued)**Configuration Bits LSW**

Word 1 configures Axis 1, Word 9 configures Axis 2 of an ANF2

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 6.7 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, they 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 five reserved words in the configuration programming blocks for each axis must always be zero.

COMMAND MODE DATA FORMAT

Once properly configured, the ANF1/2 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. The ANF1 uses ten input and output words. The ANF2 uses twenty input and output words, where the first ten are used to control axis 1 and the last ten words are used to control axis 2.

Multi-Word Parameters

Most of the values used to program the ANF1/2, as well as values reported back to the host controller, 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 ANF1/2, it is easiest to convert the parameter value to hexadecimal and enter these values into your data table. Many host controllers display 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
-1234	16#FFFF:FB2E	16#FFFF	-1	16#FB2E	-1234

Table 7.1 Multi-word Data Format

Depending on the host controller, it may be possible to directly copy 32 bit integer values directly into the two 16 bit registers used by the ANF1/2 module.



Output Data Format

The following tables shows the format of the output network data words when writing command data to both axes of the ANF1/2. The left table shows command data when writing single-axis commands. The table on the right shows command data when writing Interpolated Commands.

		Word	Axis	Standard Move Functions			Word	Interpolated Move Functions
ANF2	ANF1	0	1	Command Bits MSW	ANF1	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 ANF1/2</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	1			<i>Reserved. Must be 16#0000</i>	8	<i>Reserved, Must be 16#0000</i>
	9	1	<i>Reserved. Must be 16#0000</i>	9	<i>Reserved, Must be 16#0000</i>			
	10	2	Configuration Bits MSW	10	<i>Reserved, Must be 16#0000</i>			
	ANF2	ANF1	11	2	Configuration Bits LSW	ANF1	11	<i>Reserved, Must be 16#0000</i>
			12	2	Command Parameters <i>Word meaning depends on the command set to the ANF1/2</i>		12	EndPointY Value (Axis2)
			13	2			13	
			14	2			14	<i>Word meaning depends on the command set to the ANF1/2</i>
			15	2			15	
			16	2			16	
			17	2			17	
			18	2			<i>Reserved. Must be 16#0000</i>	18
19			2	<i>Reserved. Must be 16#0000</i>	19		<i>Reserved, Must be 16#0000</i>	

Table 7.2 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)
16#4000	14	Set Minimum Registration Move Distance

Table 7.3 Command Bits MSW

The remaining bit of this word is bit 15.

- ▶ **Bit 15: MODE CONTROL Bit:** A '0' places the ANF1/2 in Command Mode.
A '1' places the ANF1/2 in Configuration Mode.



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
No_Fault Output when Set_State = "1"	Off													0			
	On													1			
On_Line Output† when Set_State = "1"	Off												0				
	On												1				
Move_Complete when Set_State = "1"	Off											0					
	On											1					
Set_State	Off										0						
	On										1						
Backplane Home Proximity	Not Near Home									0							
	Near Home									1							
Blend Move Transmit	Accept New Block									0→1							
	Acknowledge OK'd									1→0							
Encoder Follower Mode	Disabled							0									
	Enabled							1									
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	0	1	0/1									
Run a Via Point Circular Interpolated Move		0	0	0	1	0	0	0/1									
Run a CW Center Point Circular Int. Move		0	0	1	0	0	0	0/1									
Run a CCW Center Point Circular Int. Move		0	1	0	0	0	0	0/1									
Run a CW Radius Circular Int. Move		1	0	1	0	0	0	0/1									
Run a CCW Radius Circular Int. Move		1	1	0	0	0	0	0/1									
Program Blend Move	Disabled	0															
	Enabled	1															

† Axis 1 only. This is the General Purpose Output for Axis 2 on an ANF2.

Table 7.4 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 clearing the *Command Bits MSW* and *Command Bits LSW* words before writing the next command.

Refer to the ANF1/2 sample program on the <http://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	10	<i>Command Bits MSW</i>		16#0001
1	11	<i>Command Bits LSW</i>		See pg. 59
2	12	Target Position MSW	Steps	±1,073,741,823 (16#C000:001 to 16#3FFF:FFF)
3	13	Target Position LSW		
4	14	Programmed Speed MSW	Steps/Second	Configured Starting Speed to 1,000,000 (16#000F:4240)
5	15	Programmed Speed LSW		
6	16	Acceleration	Steps/ms/sec	1 to 2000
7	17	Deceleration	Steps/ms/sec	1 to 2000
8	18	<i>Reserved</i>		16#0000
9	19	<i>Reserved</i>		16#0000

Table 7.5 Absolute Move Command Block

Relative Move

Axis 1 Word	Axis 2 Word	Function	Units	Range
0	10	<i>Command Bits MSW</i>		16#0002
1	11	<i>Command Bits LSW</i>		See pg. 59
2	12	Relative Position MSW	Steps	±1,073,741,823 (16#C000:001 to 16#3FFF:FFF)
3	13	Relative Position LSW		
4	14	Programmed Speed MSW	Steps/Second	Configured Starting Speed to 1,000,000 (16#000F:4240)
5	15	Programmed Speed LSW		
6	16	Acceleration	Steps/ms/sec	1 to 2000
7	17	Deceleration	Steps/ms/sec	1 to 2000
8	18	<i>Reserved</i>		16#0000
9	19	<i>Reserved</i>		16#0000

Table 7.6 Relative Move Command Block

The MSW position word must be -1 to perform a negative relative move of less than 65,535 counts.

**Single Axis Command Blocks (continued)****Hold Move**

Axis 1 Word	Axis 2 Word	Function	Units	Range
0	10	<i>Command Bits MSW</i>		16#0004
1	11	<i>Command Bits LSW</i>		See pg. 59 and Note below
2	12	Unused		0 (See Note below)
3	13	Unused		0 (See Note below)
4	14	Unused		0 (See Note below)
5	15	Unused		0 (See Note below)
6	16	Unused		0 (See Note below)
7	17	Unused		0 (See Note below)
8	18	Unused		0 (See Note below)
9	19	Unused		0 (See Note below)

Table 7.7 Hold Move Command Block

Unused words are ignored by the ANF1/2 and can be any value, including parameter values from the previous Absolute or Relative Move command.

Resume Move

Axis 1 Word	Axis 2 Word	Function	Units	Range
0	10	<i>Command Bits MSW</i>		16#0008
1	11	<i>Command Bits LSW</i>		See pg. 59
2	12	Unused		0 (See Note below)
3	13	Unused		0 (See Note below)
4	14	Programmed Speed MSW	Steps/Second	Configured Starting Speed to 1,000,000 (16#000F:4240)
5	15	Programmed Speed LSW		
6	16	Acceleration	Steps/ms/sec	1 to 2000
7	17	Deceleration	Steps/ms/sec	1 to 2000
8	18	Unused		0 (See Notes below)
9	19	Unused		0 (See Notes below)

Table 7.8 Resume Move Command Block

Unused words are ignored by the ANF1/2 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.



Single Axis Command Blocks (continued)

Immediate Stop

Axis 1 Word	Axis 2 Word	Function	Units	Range
0	10	<i>Command Bits MSW</i>		16#0010
1	11	<i>Command Bits LSW</i>		See pg. 59 and Note 1 below
2	12	Unused		0 (See Note 2 below)
3	13	Unused		0 (See Note 2 below)
4	14	Unused		0 (See Note 2 below)
5	15	Unused		0 (See Note 2 below)
6	16	Unused		0 (See Note 2 below)
7	17	Unused		0 (See Note 2 below)
8	18	Unused		0 (See Note 2 below)
9	19	Unused		0 (See Note 2 below)

Table 7.9 Immediate Stop Command Block

- 1) When running an Interpolate Move, the ANF2 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 ANF2 will issue a Command Error message on channel 1.
- 2) Unused words are ignored by the ANF1/2 and can be any value, including parameter values from the previous move command.

**Single Axis Command Blocks (continued)****+Find Home (CW)**

Axis 1 Word	Axis 2 Word	Function	Units	Range
0	10	<i>Command Bits MSW</i>		16#0020
1	11	<i>Command Bits LSW</i>		See pg. 59 and Note below
2	12	Reserved		Must be 0
3	13	Reserved		Must be 0
4	14	Programmed Speed MSW	Steps/Second	Configured Starting Speed to 1,000,000 (16#000F:4240)
5	15	Programmed Speed LSW		
6	16	Acceleration	Steps/ms/sec	1 to 2000
7	17	Deceleration	Steps/ms/sec	1 to 2000
8	18	<i>Reserved</i>		16#0000
9	19	<i>Reserved</i>		16#0000

Table 7.10 +Find Home (CW) Command Block

Only Constant Acceleration is available when homing an ANF1/2 axis. Bit 0 of the Command Bits LSW must equal zero.

-Find Home (CCW)

Axis 1 Word	Axis 2 Word	Function	Units	Range
0	10	<i>Command Bits MSW</i>		16#0040
1	11	<i>Command Bits LSW</i>		See pg. 59 and Note below
2	12	Reserved		Must be 0
3	13	Reserved		Must be 0
4	14	Programmed Speed MSW	Steps/Second	Configured Starting Speed to 1,000,000 (16#000F:4240)
5	15	Programmed Speed LSW		
6	16	Acceleration	Steps/ms/sec	1 to 2000
7	17	Deceleration	Steps/ms/sec	1 to 2000
8	18	<i>Reserved</i>		16#0000
9	19	<i>Reserved</i>		16#0000

Table 7.11 -Find Home (CCW) Command Block

Only Constant Acceleration is available when homing an ANF1/2 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	10	<i>Command Bits MSW</i>		16#0080
1	11	<i>Command Bits LSW</i>		See pg. 59
2	12	<i>Reserved</i>		16#0000
3	13	<i>Reserved</i>		16#0000
4	14	Programmed Speed MSW	Steps/Second	1 to 1,000,000
5	15	Programmed Speed LSW		1 to 16#000F:4240
6	16	Acceleration	Steps/ms/sec	1 to 2000
7	17	Deceleration	Steps/ms/sec	1 to 2000
8	18	<i>Reserved</i>		16#0000
9	19	<i>Reserved</i>		16#0000

Table 7.12 +Jog Move (CW) Command Block

+Registration Move (CW)

Axis 1 Word	Axis 2 Word	Function	Units	Range
0	10	<i>Command Bits MSW</i>		16#0080
1	11	<i>Command Bits LSW</i>		See pg. 59
2	12	Position Offset MSW	Steps	1 to 1,073,741,823
3	13	Position Offset LSW		1 to 16#3FFF:FFFF
4	14	Programmed Speed MSW	Steps/Second	1 to 1,000,000
5	15	Programmed Speed LSW		1 to 16#000F:4240
6	16	Acceleration	Steps/ms/sec	1 to 2000
7	17	Deceleration	Steps/ms/sec	1 to 2000
8	18	<i>Reserved</i>		16#0000
9	19	<i>Reserved</i>		16#0000

Table 7.13 +Registration (CW) Command Block

**Single Axis Command Blocks (continued)****+Encoder Follower Move**

Axis 1 Word	Axis 2 Word	Function	Units	Range
0	10	<i>Command Bits MSW</i>		16#0080
1	11	<i>Command Bits LSW</i>		16#0010
2	12	Encoder Count Multiplier		1 to 32,767
3	13	Encoder Count Divisor		1 to 32,767
4	14	Reserved		Must be 0
5	15	Reserved		Must be 0
6	16	Acceleration	Steps/ms/sec	0 to 2000
7	17	Deceleration	Steps/ms/sec	0 to 2000
8	18	<i>Reserved</i>		16#0000
9	19	<i>Reserved</i>		16#0000

Table 7.14 +Encoder Follower Move Command Block

-Jog Move (CCW)

Axis 1 Word	Axis 2 Word	Function	Units	Range
0	10	<i>Command Bits MSW</i>		16#0100
1	11	<i>Command Bits LSW</i>		See pg. 59
2	12	<i>Reserved</i>		16#0000
3	13	<i>Reserved</i>		16#0000
4	14	Programmed Speed MSW	Steps/Second	1 to 1,000,000 1 to 16#000F:4240
5	15	Programmed Speed LSW		
6	16	Acceleration	Steps/ms/sec	1 to 2000
7	17	Deceleration	Steps/ms/sec	1 to 2000
8	18	<i>Reserved</i>		16#0000
9	19	<i>Reserved</i>		16#0000

Table 7.15 -Jog Move (CCW) Command Block



-Registration Move (CCW)

Axis 1 Word	Axis 2 Word	Function	Units	Range
0	10	<i>Command Bits MSW</i>		16#0100
1	11	<i>Command Bits LSW</i>		See pg. 59
2	12	Position Offset MSW	Steps	1 to 1,073,741,823 1 to 16#3FFF:FFFF
3	13	Position Offset LSW		
4	14	Programmed Speed MSW	Steps/Second	1 to 1,000,000 1 to 16#000F:4240
5	15	Programmed Speed LSW		
6	16	Acceleration	Steps/ms/sec	1 to 2000
7	17	Deceleration	Steps/ms/sec	1 to 2000
8	18	<i>Reserved</i>		16#0000
9	19	<i>Reserved</i>		16#0000

Table 7.16 -Registration (CCW) Command Block

-Encoder Follower Move (CCW)

Axis 1 Word	Axis 2 Word	Function	Units	Range
0	10	<i>Command Bits MSW</i>		16#0100
1	11	<i>Command Bits LSW</i>		16#0010
2	12	Encoder Count Multiplier		1 to 32,767
3	13	Encoder Count Divisor		1 to 32,767
4	14	Reserved		Must be 0
5	15	Reserved		Must be 0
6	16	Acceleration	Steps/ms/sec	0 to 2000
7	17	Deceleration	Steps/ms/sec	0 to 2000
8	18	<i>Reserved</i>		16#0000
9	19	<i>Reserved</i>		16#0000

Table 7.17 -Encoder Follower Move Command Block

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

Axis 1 Word	Axis 2 Word	Function	Units	Range
0	10	<i>Command Bits MSW</i>		16#0200
1	11	<i>Command Bits LSW</i>		See pg. 59
2	12	New Position MSW	Steps	±1,073,741,823 (16#C000:0001 to 16#3FFF:FFFF)
3	13	New Position LSW		
4	14	<i>Reserved</i>		16#0000
5	15	<i>Reserved</i>		16#0000
6	16	<i>Reserved</i>		16#0000
7	17	<i>Reserved</i>		16#0000
8	18	<i>Reserved</i>		16#0000
9	19	<i>Reserved</i>		16#0000

Table 7.18 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	10	<i>Command Bits MSW</i>		16#0400
1	11	<i>Command Bits LSW</i>		See pg. 59
2	12	New Encoder Position MSW	Steps	±1,073,741,823 (16#C000:0001 to 16#3FFF:FFFF)
3	13	New Encoder Position LSW		
4	14	<i>Reserved</i>		16#0000
5	15	<i>Reserved</i>		16#0000
6	16	<i>Reserved</i>		16#0000
7	17	<i>Reserved</i>		16#0000
8	18	<i>Reserved</i>		16#0000
9	19	<i>Reserved</i>		16#0000

Table 7.19 Preset Encoder Position Command Block



Single Axis Command Blocks (continued)

Reset Errors

Axis 1 Word	Axis 2 Word	Function	Units	Range
0	10	<i>Command Bits MSW</i>		16#0800
1	11	<i>Command Bits LSW</i>		16#0000
2	12	<i>Reserved</i>		16#0000
3	13	<i>Reserved</i>		16#0000
4	14	<i>Reserved</i>		16#0000
5	15	<i>Reserved</i>		16#0000
6	16	<i>Reserved</i>		16#0000
7	17	<i>Reserved</i>		16#0000
8	18	<i>Reserved</i>		16#0000
9	19	<i>Reserved</i>		16#0000

Table 7.20 Reset Errors Command Block

Run +Blend Move Profile (CW)

Axis 1 Word	Axis 2 Word	Function	Units	Range
0	10	<i>Command Bits MSW</i>		16#1000
1	11	<i>Command Bits LSW</i>		16#0000
2	12	<i>Reserved</i>		16#0000
3	13	<i>Reserved</i>		16#0000
4	14	<i>Reserved</i>		16#0000
5	15	<i>Reserved</i>		16#0000
6	16	<i>Reserved</i>		16#0000
7	17	<i>Reserved</i>		16#0000
8	18	<i>Reserved</i>		16#0000
9	19	<i>Reserved</i>		16#0000

Table 7.21 Run +Blend Move Profile (CW) Command Block



Single Axis Command Blocks (continued)

Run –Blend Move Profile (CCW)

Axis 1 Word	Axis 2 Word	Function	Units	Range
0	10	<i>Command Bits MSW</i>		16#2000
1	11	<i>Command Bits LSW</i>		16#0000
2	12	<i>Reserved</i>		16#0000
3	13	<i>Reserved</i>		16#0000
4	14	<i>Reserved</i>		16#0000
5	15	<i>Reserved</i>		16#0000
6	16	<i>Reserved</i>		16#0000
7	17	<i>Reserved</i>		16#0000
8	18	<i>Reserved</i>		16#0000
9	19	<i>Reserved</i>		16#0000

Table 7.22 Run –Blend Move Profile (CW) Command Block

Set Min. Registration Move Distance

Axis 1 Word	Axis 2 Word	Function	Units	Range
0	10	<i>Command Bits MSW</i>		16#4000
1	11	<i>Command Bits LSW</i>		16#0000
2	12	Min. Reg. Distance MSW	Steps	±1,073,741,823 (16#C000:0001 to 16#3FFF:FFFF)
3	13	Min. Reg. Distance LSW		
4	14	<i>Reserved</i>		16#0000
5	15	<i>Reserved</i>		16#0000
6	16	<i>Reserved</i>		16#0000
7	17	<i>Reserved</i>		16#0000
8	18	<i>Reserved</i>		16#0000
9	19	<i>Reserved</i>		16#0000

Table 7.23 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 show the word values needed to perform a Blend Move. These sections are on pages 68 and 69. This section covers how to program a Blend Move Profile. A Blend Move Profile must be programmed into the ANF1/2 before a Blend Move can be executed.

Multiple writes to the ANF1/2 are required to program a single Blend Move Profile. The ANF1/2 uses a simple handshaking protocol to control this transfer of data. Two bits in the Output Network Data and two bits in the Input Network Data are used in the transfer of data blocks.

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

- ▶ **Program Blend Profile** (Bit 15 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 7 of Command Bits LSW) Make a 0→1 transition on this bit to tell the ANF1/2 that new Blend Move Profile data is available. Make a 1→0 transition on this bit after the ANF1/2 sets its *Program Blend Profile Acknowledge* bit in its Input Network Data data to complete the transfer of the block of data.

The two bits in the Input Network Data are introduced here. The remaining values in the Input Network Data are introduced in the following section.

- ▶ **Program Blend Profile Acknowledge** (Bit 9 of Status Bits LSW) This bit is used by the ANF1/2 to tell the host controller 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 ANF1/2 to tell the host controller 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 host controller's program writes the *Initial Profile Programming Block* to the ANF1/2 with the *Program Blend Profile* and *Program Blend Profile Transmit* bits both set to 1.
- 2) The ANF1/2 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) Your host controller's program must detect when the *Program Blend Profile Acknowledge* bit is set. When this bit transitions, check the value of the *Invalid Blend Profile* bit and act appropriately.
- 4) Your host controller's program then needs to reset the *Program Blend Profile Transmit* bit.
- 5) The ANF1/2 responds by resetting its *Program Blend Profile Acknowledge* bit.
- 6) a: If you have additional data blocks that must be sent to the ANF1/2, repeat these instructions from step 1, sending the next *Secondary Profile Programming Block*.
b: If you have sent the last data block, then reset the *Program Blend Profile* bit to end the programming cycle.

NOTE  Two sample programs for the ANF1/2 are available on the <http://www.amci.com> website. One of them includes the sample code needed to program a Blend Move Profile.

Two different types of 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 used between the segments. (Constant, Triangular S-Curve, Trapezoidal S-Curve) This must be the first data block written to the ANF1/2 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 values. 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#8080
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
8	18	Reserved		Must be 0
9	19	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 7.24 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#8080
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
8	18	Reserved		Must be 0
9	19	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#C000:0001 to 16#3FFF:FFFF)
7	EndPointX LSW (axis1)		
8	Reserved		Must be 0
9	Reserved		Must be 0
10	Reserved		Must be 0
11	Reserved		Must be 0
12	EndPointY MSW (axis2)	Steps	±1,073,741,823 (16#C000:0001 to 16#3FFF:FFFF)
13	EndPointY LSW (axis2)		
14	Reserved		Must be 0
15	Reserved		Must be 0
16	Reserved		Must be 0
17	Reserved		Must be 0
18	Reserved		Must be 0
19	Reserved		Must be 0

Table 7.25 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 two bits of the Command Bits LSW can equal 2#00 or 2#10. (Both bits reset or bit 1 set)
- For Triangular S-Curve Acceleration, the lower two bits of the Command Bits LSW must equal 2#01. (Bit 0 set)
- For Trapezoidal S-Curve Acceleration, the lower two bits of the Command Bits LSW must equal 2#11. (Bits 1 and 0 set)



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#C000:0001 to 16#3FFF:FFFF)
7	EndPointX LSW (axis1)		
8	Reserved		Must be 0
9	Reserved		Must be 0
10	Reserved		Must be 0
11	Reserved		Must be 0
12	EndPointY MSW (axis2)	Steps	±1,073,741,823 (16#C000:0001 to 16#3FFF:FFFF)
13	EndPointY LSW (axis2)		
14	ViaPointX MSW (axis1)	Steps	±1,073,741,823 (16#C000:0001 to 16#3FFF:FFFF)
15	ViaPointX LSW (axis1)		
16	ViaPointY MSW (axis2)	Steps	±1,073,741,823 (16#C000:0001 to 16#3FFF:FFFF)
17	ViaPointY LSW (axis2)		
18	Reserved		Must be 0
19	Reserved		Must be 0

Table 7.26 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 two bits of the Command Bits LSW can equal 2#00 or 2#10. (Both bits reset or bit 1 set)
- For Triangular S-Curve Acceleration, the lower two bits of the Command Bits LSW must equal 2#01. (Bit 0 set)
- For Trapezoidal S-Curve Acceleration, the lower two bits of the Command Bits LSW must equal 2#11. (Bits 1 and 0 set)

**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#C000:0001 to 16#3FFF:FFFF)
7	EndPointX LSW (axis1)		
8	Reserved		Must be 0
9	Reserved		Must be 0
10	Reserved		Must be 0
11	Reserved		Must be 0
12	EndPointY MSW (axis2)	Steps	±1,073,741,823 (16#C000:0001 to 16#3FFF:FFFF)
13	EndPointY LSW (axis2)		
14	CenterPointX MSW (axis1)	Steps	±1,073,741,823 (16#C000:0001 to 16#3FFF:FFFF)
15	CenterPointX LSW (axis1)		
16	CenterPointY MSW (axis2)	Steps	±1,073,741,823 (16#C000:0001 to 16#3FFF:FFFF)
17	CenterPointY LSW (axis2)		
18	Reserved		Must be 0
19	Reserved		Must be 0

Table 7.27 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 two bits of the Command Bits LSW can equal 2#00 or 2#10. (Both bits reset or bit 1 set)
- For Triangular S-Curve Acceleration, the lower two bits of the Command Bits LSW must equal 2#01. (Bit 0 set)
- For Trapezoidal S-Curve Acceleration, the lower two bits of the Command Bits LSW must equal 2#11. (Bits 1 and 0 set)



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#C000:0001 to 16#3FFF:FFFF)
7	EndPointX LSW (axis1)		
8	Reserved		Must be 0
9	Reserved		Must be 0
10	Reserved		Must be 0
11	Reserved		Must be 0
12	EndPointY MSW (axis2)	Steps	±1,073,741,823 (16#C000:0001 to 16#3FFF:FFFF)
13	EndPointY LSW (axis2)		
14	Radius MSW	Steps	±1,073,741,823 [†] (16#C000:0001 to 16#3FFF:FFFF)
15	Radius LSW		
16	Reserved		Must be 0
17	Reserved		Must be 0
18	Reserved		Must be 0
19	Reserved		Must be 0

Table 7.28 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 two bits of the Command Bits LSW can equal 2#00 or 2#10. (Both bits reset or bit 1 set)
- For Triangular S-Curve Acceleration, the lower two bits of the Command Bits LSW must equal 2#01. (Bit 0 set)
- For Trapezoidal S-Curve Acceleration, the lower two bits of the Command Bits LSW must equal 2#11. (Bits 1 and 0 set)

**Input Data Format**

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

Word	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	1	Reserved		16#0000
9	1	Network Connectivity		See Description below
10	2	Status Bits MSW		See Description below
11	2	Status Bits LSW		See Description below
12	2	Current Position	Steps	-2,147,483,648 to 2,147,483,647 (32 bit signed value)
13	2			
14	2	Current Encoder Position / Diagnostic Feedback Value	Counts	-2,147,483,648 to 2,147,483,647 (32 bit signed value)
15	2			
16	2	Capture Data	Steps or Counts	-2,147,483,648 to 2,147,483,647 (32 bit signed value)
17	2			
18	2	Reserved		16#0000
19	2	Reserved		16#0000

Table 7.29 Input Data Format - Command Data

Status Bits MSW

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

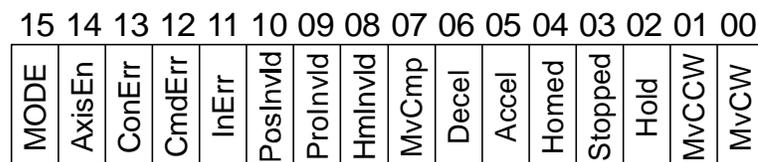
Status Bits MSW

Figure 7.1 Status Bits MSW

- BIT 0: CW Motion:** Set when the ANF1/2 axis is outputting pulses for clockwise motion.
- BIT 1: CCW Motion:** Set when the ANF1/2 axis is outputting pulses for counter-clockwise motion.
- BIT 2: Hold State:** Set when the ANF1/2 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 ANF1/2 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.



Input Data Format (continued)

Status Bits MSW (continued)

- BIT 5: Move Accelerating:** Set when the ANF1/2 axis is accelerating during any move.
- BIT 6: Move Decelerating:** Set when the ANF1/2 axis is decelerating during any move.
- BIT 7: Move Complete:** Set when the ANF1/2 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 ANF1/2 could not home the axis because of an error in a \pm Find Home command data, the homing operation exceeded its programmed timeout value, an E-Stop input activates, or the limit switch that is opposite to the current direction of travel activates.
- BIT 9: Profile Invalid:** Set when there was an error in the last Program Blend Profile data block written to the ANF1/2 axis. See the *Programming Blend Move Profiles* section starting on page 70 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:**
- \pm 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.
- \pm 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.
- \pm 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 ANF1/2 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 ANF1/2 axis. Other commands, including the *Immediate Stop* command, will not be accepted until this error is reset.
- BIT 13: Configuration Error:** Set to “1” when the axis has a configuration error.
- BIT 14: Axis Enabled:** Set to “1” when the axis is enabled. Axis 2 of an ANF2 is disabled by default. An axis is automatically enabled when valid configuration data is written to it.
- BIT 15: Axis Mode:** Set to “1” when the axis is in Configuration Mode. Reset to “0” when the axis is in Command Mode.



Input Data Format (continued)

Status Bits LSW

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

Status Bits LSW

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
0	0	EFMode	HmProx	IntrpMv	NonORD	BMvAck	BlndMv	MvCmpOut	OnLineOut	NoFltOut	CCWIn	CWIn	HomeIn	ExtIn	CapIn

Figure 7.2 Status Bits LSW

- 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 51 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 51 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 51 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 50 and 51 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 50 and 51 for more information.
- BIT 5: No_Fault Output State:** This bit shows the state of the No_Fault output for the axis. This bit must be set to its active state (“1”) by the host controller. Once set, this bit will be reset by the following conditions:
 - Input Error during a move. (E-Stop, CW Limit, or CCW Limit Input activated during move.)
 - Axis switched from Command Mode to Configuration Mode
 - Loss of network communications
 - Reset by host controller.
- BIT 6: On_Line / GP Output State:** This bit shows the state of the On_Line output for axis 1 and the state of the General Purpose output for axis 2. The On_Line bit must be set to its active state (“1”) by the host controller. Once set, this bit will be reset by the following conditions:
 - Loss of network communications
 - Reset by host controller.

The state of the General Purpose output is only controlled by the host controller.
- BIT 7: Move_Complete Output State:** This bit shows the state of the Move Complete output for the axis. This bit will be set by the following conditions:
 - Absolute, Relative, Blend, or Interpolated Move completes without error.
 - Set by host controller.

Once set, this bit must be reset to its inactive state (“0”) by the host controller.
- BIT 8: Running Blend Move:** Set when the axis is presently running a Blend Move Profile. This bit is reset when the move completes.
- BIT 9: Blend Move Acknowledge Bit:** Set when ANF1/2 has accepted a Blend Move Profile programming block. See *Programming Blend Move Profiles*, which starts on page 70, for additional information.



Input Data Format (continued)

Status Bits LSW (continued)

- 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 ANF1/2 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: Encoder Follower Mode:** Set when the axis is in Encoder Follower Mode.
- BITS 14 & 15: Reserved:** These bits will always equal zero.

Network Connectivity, Word 9

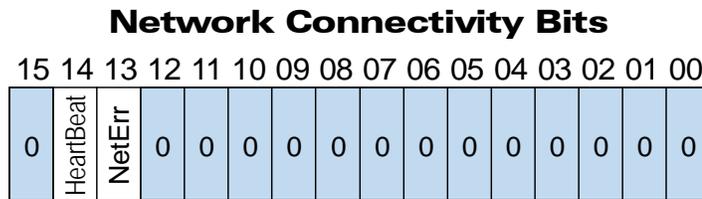


Figure 7.3 Network Connectivity Bits

- BITS 0 - 12: Reserved:** These bits will always equal zero.
- BIT 13: Unacknowledged Network Disconnect:** This bit is set to “1” if the module loses its network connection. This bit is set until a Reset Errors command is sent to the module.
- BIT 14: Heartbeat Bit:** This bit changes state every 500 milliseconds and can be used to verify active network communications.
- BIT 15: Reserved:** This bit will always equal zero.

***Input Data Format (continued)*****Current Position****Words 2 & 3 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 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 14 & 15 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 16 & 17 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 51.) 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 50.

TABULATED COMMAND AND 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: Input Word 0, bit 3 Axis 2: Input Word 10, bit 3

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

Move Complete Bit

Axis1: Input Word 0, bit 7 Axis 2: Input Word 10, bit 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: Input Word 0, bit 8 Axis 2: Input Word 10, bit 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.

Profile Invalid Bit

Axis1: Input Word 0, bit 9 Axis 2: Input Word 10, bit 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 ANF1/2 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: Input Word 0, bit 9 Axis 2: Input Word 10, bit 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 Minimum Registration Distance parameter equals zero and a Registration Move is started while 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: Input Word 0, bit 10 Axis 2: Input Word 10, bit 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: Input Word 0, bit 11 Axis 2: Input Word 10, bit 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: Input Word 0, bit 12 Axis 2: Input Word 10, bit 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: Output Word 0, Axis2: Output Word 10)
- 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: Input Word 0, bit 13 Axis 2: Input Word 10, bit 13

The Configuration Error bit will be set under the following conditions

- 1) At power up, before the ANF1/2 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 Input Word 0, bit 12 for axis 1 and Input Word 10, bit 12 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 ANF1/2 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 ANF1/2 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