



HE693STPxxx User's Manual

Stepper Positioning Module (SPM30)
For GE Series 90-30 PLC

Seventh Edition
10 September 2002

MAN0084-07

PREFACE

This manual explains how to use the Horner APG Stepper Positioning Module for GE Series 90-30 Programmable Logic Controllers.

Copyright (C) 2002, Horner APG, LLC, 640 North Sherman Drive, Indianapolis, Indiana 46201. All rights reserved. No part of this publication can be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise, without the prior agreement and written permission of Horner APG, LLC.

All software described in this document or media is also copyrighted material subject to the terms and conditions of the Horner Software License Agreement.

Information in this document is subject to change without notice and does not represent a commitment on the part of Horner APG, LLC.

Windows, Windows NT, Windows 95 and Windows 98 are registered trademarks of Microsoft Corporation.

For user manual updates, contact Horner APG, Technical Support Division, at (317) 916-4274 or visit our website at www.heapg.com.

LIMITED WARRANTY AND LIMITATION OF LIABILITY

Horner APG, LLC. ("HE-APG") warrants to the original purchaser that Stepper Positioning Module manufactured by HE-APG is free from defects in material and workmanship under normal use and service. The obligation of HE-APG under this warranty shall be limited to the repair or exchange of any part or parts which may prove defective under normal use and service within two (2) years from the date of manufacture or eighteen (18) months from the date of installation by the original purchaser whichever occurs first, such defect to be disclosed to the satisfaction of HE-APG after examination by HE-APG of the allegedly defective part or parts. THIS WARRANTY IS EXPRESSLY IN LIEU OF ALL OTHER WARRANTIES EXPRESSED OR IMPLIED INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR USE AND OF ALL OTHER OBLIGATIONS OR LIABILITIES AND HE-APG NEITHER ASSUMES, NOR AUTHORIZES ANY OTHER PERSON TO ASSUME FOR HEAPG, ANY OTHER LIABILITY IN CONNECTION WITH THE SALE OF THIS Stepper Positioning Module. THIS WARRANTY SHALL NOT APPLY TO THIS Stepper Positioning Module OR ANY PART THEREOF WHICH HAS BEEN SUBJECT TO ACCIDENT, NEGLIGENCE, ALTERATION, ABUSE, OR MISUSE. HE-APG MAKES NO WARRANTY WHATSOEVER IN RESPECT TO ACCESSORIES OR PARTS NOT SUPPLIED BY HE-APG. THE TERM "ORIGINAL PURCHASER", AS USED IN THIS WARRANTY, SHALL BE DEEMED TO MEAN THAT PERSON FOR WHOM THE Stepper Positioning Module IS ORIGINALLY INSTALLED. THIS WARRANTY SHALL APPLY ONLY WITHIN THE BOUNDARIES OF THE CONTINENTAL UNITED STATES.

In no event, whether as a result of breach of contract, warranty, tort (including negligence) or otherwise, shall HE-APG or its suppliers be liable of any special, consequential, incidental or penal damages including, but not limited to, loss of profit or revenues, loss of use of the products or any associated equipment, damage to associated equipment, cost of capital, cost of substitute products, facilities, services or replacement power, down time costs, or claims of original purchaser's customers for such damages.

To obtain warranty service, return the product to your distributor with a description of the problem, proof of purchase, post paid, insured and in a suitable package.

ABOUT PROGRAMMING EXAMPLES

Any example programs and program segments in this manual or provided on accompanying diskettes are included solely for illustrative purposes. Due to the many variables and requirements associated with any particular installation, Horner APG cannot assume responsibility or liability for actual use based on the examples and diagrams.

It is the sole responsibility of the system designer utilizing the Stepper Positioning Module to appropriately design the end system, to appropriately integrate the Stepper Positioning Module and to make safety provisions for the end equipment as is usual and customary in industrial applications as defined in any codes or standards which apply.

Note: The programming examples shown in this manual are for illustrative purposes only. Proper machine operation is the sole responsibility of the system integrator.

REVISIONS TO THIS MANUAL

This version (MAN0084-07) of the Stepper Positioning Module User's Manual contains the following revisions, additions, and deletions:

1. Added text for new features in Section 1.3 and Section 5.3.6 (Auto Repeat of Relative and Indexed Moves).
2. Added text for a new feature allowing the optional use of DIR as a move in progress output (for indexed moves only) in Section 1.3 and in a note contained in Section 5.3.6.
3. Incorporated the information contained in Chapter 6 (formerly **Figuring Parameters**) into the Chapter 4 (**Registers**). Chapter 6 is now replaced by **Using Encoders** (formerly Chapter 7), and Chapter 7 is now replaced by **Wiring** (formerly Chapter 8).
4. Added new table to Chapter 4 and numbered it as Table 4.1: STP100 Module OCS Register Summary. Tables and other information have been renumbered throughout Chapter 4 to reflect the incorporation of Chapter 6 material.
5. Eliminated the S-Curve feature (Section 1.3 and 1.7).
6. Replaced Figure 1.5 (S-Curve Motion Profile) with new drawing (Triangular Motion Profile).
7. Added new figure to Chapter 1 (Figure 1.1) and Chapter 7 (Figure 7.6).
8. Made clarifications and punctuation corrections throughout edition as needed.
9. Added Table 6.1 covering example encoder multiplier and divisor values.

TABLE OF CONTENTS

PREFACE **3**

LIMITED WARRANTY AND LIMITATION OF LIABILITY **4**

ABOUT PROGRAMMING EXAMPLES **4**

CHAPTER 1: INTRODUCTION **9**

 1.1 General 9

 1.2 Overview 9

 1.3 Features 9

 1.4 LED Indicators 10

 1.5 Model-Dependent Features 10

 1.5.1 Model-Dependent Feature Descriptions 10

 1.6 Specifications 11

 1.7 Motion Profiles 12

CHAPTER 2: INSTALLATION **13**

 2.1 General 13

 2.2 Module Placement 13

 2.3 I/O Connector Wiring Summary 13

 2.4 I/O Connector Signal Descriptions 15

 2.4.1 Non-Isolated Inputs 15

 2.4.2 Isolated Inputs 15

 2.4.3 Differential Inputs 16

 2.4.4 Differential Outputs 16

 2.5 I/O Connector Wiring Examples 17

 2.5.1 Switch Wiring 17

 2.5.2 Encoder Wiring 18

 2.5.3 Translator Drive Wiring 19

CHAPTER 3: CONFIGURATION **21**

 3.1 General 21

 3.2 HHP Configuration 21

 3.3 LM90 Configuration 21

 3.4 Configurable PLC Resources 22

 3.5 Configuration Parameter Description 22

 3.5.1 Find Home Method 22

 3.5.2 Stop Delay Factor 23

 3.5.3 Encoder Type, Encoder Multiplier, Encoder Divisor and Encoder Tolerance 23

CHAPTER 4: REGISTERS **25**

 4.1 General 25

 4.2 SPM30 Module PLC I/O Register Details 26

 4.2.1 Error and Status Input Bit Register Details (%I1 - %I16) 26

 4.2.2 Status Word Input Register Details (%AI1 - %AI4) 27

 4.2.3 Command Word Output Register Details (%AQ1 - %AQ14) 28

 4.2.4 Command Bit Output Register Details (%Q1 - %Q16) 31

CHAPTER 5: COMMANDS **33**

 5.1 General 33

 5.2 Issuing Commands 33

 5.3 Available Commands 33

 5.3.1 Select Axis Commands 33

 5.3.2 Find Home Up and Find Home Down Commands 34

 5.3.3 Jog Up and Jog Down Commands 35

 5.3.4 Move Relative and Move Absolute Commands 36

 5.3.5 Resume Move Command 37

 5.3.6 Move Indexed Command 38

 5.3.7 Repeat Move Command 39

 5.3.8 Set Current Position Command 39

5.3.9	Clear Error(s) Command	40
5.3.10	Decelerate and Stop Command	40
5.3.11	Immediate Stop Command	40
CHAPTER 6: ENCODERS	41
6.1	General	41
6.2	Encoder Type Configuration	41
6.2.1	Quadrature Encoder	41
6.2.2	Up / Down Encoder	41
6.3	Encoder Multiplier and Divisor Configuration	42
6.3.1	Encoder Multiplier and Divisor Configuration Example	42
6.4	Encoder Tolerance Configuration	43
6.4.1	Motor Position Validation	43
6.4.2	Motor Stall Detection	43
INDEX	44

CHAPTER 1: INTRODUCTION

1.1 General

Chapter One provides a brief introduction to Horner APG, LLC Stepper Positioning Modules.

1.2 Overview

The Stepper Positioning Module (SPM30) is an intelligent, programmable motion control option module, for the GE Fanuc Series 90-30 family of Programmable Logic Controllers (PLC).

Within a stepper motor control system, the SPM30 acts as a programmable indexer, which is capable of interfacing to a wide variety of stepper motor translator drives, limit switches, and encoder feedback devices.

In addition, its high maximum step rate and wide dynamic range position control make the SPM30 compatible with micro-stepping translator drives, for smooth, quiet operation.

1.3 Features

- ❖ One and three axis (multiplexed) models
- ❖ Up to 245,730 steps (or micro-steps) per second
- ❖ Motion completely controlled by application ladder program
- ❖ Automatic find home and manual jog moves supported
- ❖ **On-the-fly velocity changes** for manual jog moves
- ❖ Absolute, relative and indexed moves
- ❖ Relative and indexed moves **can be auto-repeated**
- ❖ Power-up / watchdog timeout safety interlock
- ❖ Home and over-travel inputs (isolated on some models)
- ❖ Emergency stop input (isolated on some models)
- ❖ Incremental encoder feedback inputs (for axis 1)
- ❖ Programmable position, velocity and acceleration
- ❖ Automatic ramp-down deceleration calculation
- ❖ Programmable deceleration (on some models)
- ❖ Trapezoidal and triangular velocity profiles

1.4 LED Indicators

All SPM30 models provide three (3) LED indicators, as shown in Table 1.1.

Table 1.1 – LED Indicators	
Indicator Color	Function
Red	Motor is Stopped
Yellow	Motor is Accelerating or Decelerating
Green	Motor has reached Running Velocity

1.5 Model-Dependent Features

Table 1.2 shows the ten available SPM30 models along with their model-dependent features.

Table 1.2 – Models and their Features							
Model	Features						
HE693-	1 or 3 Axis	Encoder Support	Isolated Inputs	Programmable Deceleration	End Limit Triggered Moves	Active Hi End Limits	Indexed Moves
STP100	1	No	No	No	No	No	No
STP110	1	Yes	No	No	No	No	No
STP300	3	No	No	No	No	No	No
STP310	3	Yes	No	No	No	No	No
STP101	1	No	Yes	Yes	Yes	No	No
STP111	1	Yes	Yes	Yes	Yes	No	No
STP301	3	No	Yes	Yes	Yes	No	No
STP311	3	Yes	Yes	Yes	Yes	No	No
STP104	1	No	Yes	Yes	Yes	Yes	No
STP113	1	Yes	Yes	Yes	Yes	No	Yes

1.5.1 Model-Dependent Feature Descriptions

Table 1.3 describes the model-dependent features in more detail.

Table 1.3 – Model-Dependent Feature Descriptions	
Feature	Description
1 or 3 Axis	STP1xx are <u>single</u> -axis Stepper Positioning Modules. STP3xx are <u>three</u> -axis Stepper Positioning Modules (multiplexed).
Encoder Support	STPx0x <u>do not</u> support encoder feedback. STPx1x <u>do</u> support encoder feedback.
Isolated Inputs	STPx0x have <u>non-isolated</u> Home, End Limit and Emergency Stop inputs. STPx1x, STP104, STP113 have <u>isolated</u> Home, End Limit and Emergency Stop inputs.
Programmable Deceleration	STPx0x <u>do not</u> have separate programmable deceleration. STPx1x, STP104, STP113 <u>do</u> have separate programmable deceleration.
End Limit Triggered Moves	STPx0x moves <u>can not</u> be triggered by hardware End Limit inputs. STPx1x, STP104, STP113 <u>can</u> be triggered by hardware End Limit inputs.
Active Hi End Limits	STP104 has <u>active-high</u> End Limit inputs. All other models have <u>active-low</u> End Limit inputs.
Indexed Moves	STP113 <u>does</u> support Indexed Moves. All other models <u>do not</u> support Indexed Moves.

1.6 Specifications

Table 1.3 – I/O Connector Specifications			
Parameter	Minimum	Maximum	Units
All Models (STPxxx)			
Step Output Frequency	DC	245	KHz
Step Output HI (-20mA)	2.5	-	Vdc
Step Output LO (+20mA)	-	0.5	Vdc
Direction Output Setup Time			
Direction Output Setup Time	2	-	Milliseconds
Direction Output HI (-60 mA)	2.5	-	Vdc
Direction Output LO (+60 mA)	-	0.5	Vdc
Encoder Input Frequency			
Encoder Input Frequency	DC	1.0	MHz
Encoder Single-Ended Threshold	1.2	1.6	Vdc
Encoder Differential Threshold HI	-	+0.2	Vdc
Encoder Differential Threshold LO	-0.2	-	Vdc
Non-Isolated Models (STPxx0)			
Home Input ON (+2mA)	-	0.8	Vdc
End Limit Input ON (+2mA)	-	0.8	Vdc
Emergency Stop Input OFF (+2mA)	-	0.8	Vdc
Isolated Models (STPxx1, STP113 and STP104)			
Home, End Limit and Emergency Stop Common Mode Isolation Voltage	500	-	Vdc
Home Input ON (+1mA)	-	9.0	Vdc
End Limit Input ON (+1mA) (STPxx1 and STP113)	-	9.0	Vdc
End Limit Input OFF (+1mA) (STP104)	∞	9.0	Vdc
Emergency Stop Input OFF (+1mA)	-	9.0	Vdc

Table 1.4 – Power Load Specifications			
Parameter	Minimum	Maximum	Units
+5 Vdc (Logic)	200	500	mA
+24 Vdc (Relay)	-	0	mA
+24 Vdc (Isolated)	-	0	mA

Table 1.5 – Environmental Specifications			
Parameter	Minimum	Maximum	Units
Operating Temperature	0	60	°C
Storage Temperature	-40	85	°C
Humidity (non-condensing)	5	95	%RH

1.7 Motion Profiles

For a typical move, the SPM30 starts at a low Base Velocity, accelerates up to the Running Velocity, and later decelerates down to the Base Velocity, and then stops. This is known as the trapezoidal motion profile, and is shown in Figure 1.1 below.

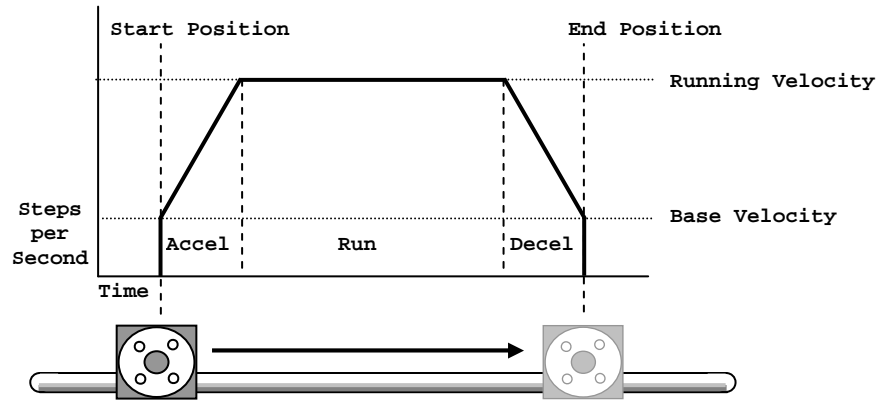


Figure 1.1 – Trapezoidal Motion Profile

For a given set of programmed Acceleration and Deceleration Times, a particular move can be too short to allow the motor to accelerate all the way up to Running Velocity. If it is time to start decelerating before accelerating to Running Velocity has completed, the motor will start decelerating at that point. In this case, the acceleration and deceleration rates (in pulses-per-second²) are preserved, but the acceleration and deceleration times are decreased. When this happens, the velocity profile becomes triangular, as shown in Figure 1.2 below.

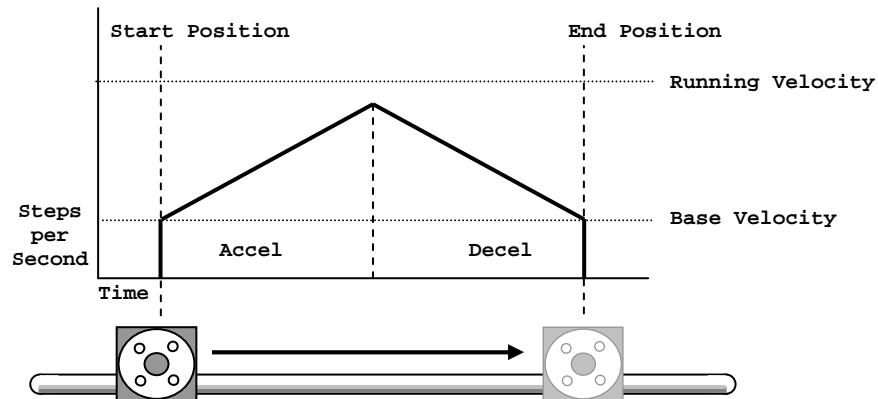


Figure 1.2 – Triangular Motion Profile

CHAPTER 2: INSTALLATION

2.1 General

Chapter Two describes the installation of the Stepper Positioning Module into a GE Series 90-30 PLC.

2.2 Module Placement

The Stepper Positioning Module (SPM30) can be placed in any available I/O slot of a GE Fanuc Series 90-30 PLC.

WARNING: The SPM30 must never be inserted or removed, while power is applied to the host PLC, or when the cables to the motor drive are attached to the SPM30 I/O connector. Follow the guidelines defined in the Series 90-30 literature for proper module insertion and removal.

2.3 I/O Connector Wiring Summary

The Stepper Positioning Module (SPM30) is equipped with a 20-contact removable terminal block. The pinout of the removable terminal block is illustrated on the following page.

Terminals are provided for interfacing the SPM30 with a variety of devices, including:

- a) Stepper Motor Drives (translators)
- b) Incremental encoders (quadrature or up/down)
- c) Mechanical and proximity type limit switches
- d) E-stop pushbuttons

The sign (+ or –) after each signal name indicates what state the signal is in when it is active. Active-high is indicated by +, while active-low is indicated by –.

The motor step and direction outputs can be connected to translator drives, which have either differential or single-ended inputs. For single-ended drives, select the SPM30 motor output whose active signal state (+ / –) matches the drive's input.

The encoder inputs can be connected to an encoder, which has either differential or single-ended outputs. For single-ended encoders, use the active-high encoder inputs (EA+ and EB+).

If the home and/or end limit inputs are used, they should be connected to ground (pin 20) via normally open dry-contact switches or NPN-type proximity switches. (The STP104 is the exception to this rule, because its end limit inputs are active-high, requiring normally closed switches to ground.)

The emergency stop input should be connected to ground (pin 20) via a normally closed pushbutton.

The highest voltage with respect to ground (pin 20) which can be applied to the home, end limit and emergency stop inputs is 5 Vdc for non-isolated models, and 24 Vdc for isolated models.

The removable terminal block I/O connections have the following functions:

Table 2.1 – I/O Connector Pin-Out				
Pin	Signal Name	Notes	Signal Description	Signal Type
1	DGND		Common return for pins 2 through 13	-
2	S1-		Axis 1 motor step outputs	Differential
3	S1+			
4	S2-	1	Axis 2 motor step outputs	Differential
5	S2+	1		
6	S3-	1	Axis 3 motor step outputs	Differential
7	S3+	1		
8	DR-		Motor direction outputs	Differential
9	DR+			
10	EB-	2	Axis 1 Phase B incremental encoder inputs	Differential
11	EB+	2		
12	EA-	2	Axis 1 Phase A incremental encoder inputs	Differential
13	EA+	2		
14	H3- or INDEX-	3	Axis 3 home input or axis 1 index input	Single-ended
15	HL- or HL+	4	Upper end limit input	Single-ended
16	H2-	1	Axis 2 home input	Single-ended
17	LL- or LL+	5	Lower end limit input	Single-ended
18	H1-		Axis 1 home input	Single-ended
19	ES+		Emergency stop input	Single-ended
20	DGND or IGND	6	Common return for pins 14 through 19	-

Note 1: Pins 4 through 7, and 16 are not used for STP1xx (single-axis models).

Note 2: Pins 10 through 13 are not used for STPx0x (non-encoder models).

Note 3: Pin 14 is H3- for STP3xx (three-axis models).
Pin 14 is INDEX- for STP113 (indexed move model).
Pin 14 is not used for other models.

Note 4: Pin 15 is HL- for STPxx0, STPxx1 and STP113 (active-low end limit models).
Pin 15 is HL+ for STP104 (active-high end limit model).

Note 5: Pin 17 is LL- for STPxx0, STPxx1 and STP113 (active-low end limit models).
Pin 17 is LL+ for STP104 (active-high end limit model).

Note 6: Pin 20 is DGND (same as pin 1) for STPxx0 (non-isolated models).
Pin 20 is IGND for STPxx1, STP113 and STP104 (isolated models).

2.4 I/O Connector Signal Descriptions

Emergency Stop input is an active high signal. A Normally Closed (N/C) switch is used to connect this signal to ground for normal operation.

On most models, the **Home**, **Low Limit**, **High Limit** and **Index** inputs are all active low signals. A Normally Open (N/O) switch is used to connect these signals to ground to activate them. On the HE693STP104, the **Low Limit** and **High Limit** inputs are active high signals.

Depending on the SPM30 model, the **Home**, **Low Limit** and **High Limit** inputs are either Non-Isolated or Isolated, as shown in the following 2 sections. The **Index** input (only on HE693STP113) is always isolated.

2.4.1 Non-Isolated Inputs

For models with non-isolated inputs (STPxx0), the **Emergency Stop**, **Home**, **Low Limit**, and **High Limit** inputs are connected using the following circuitry. **Home**, **Low Limit**, and **High Limit** inputs can accept either mechanical switches or NPN-type proximity switches.

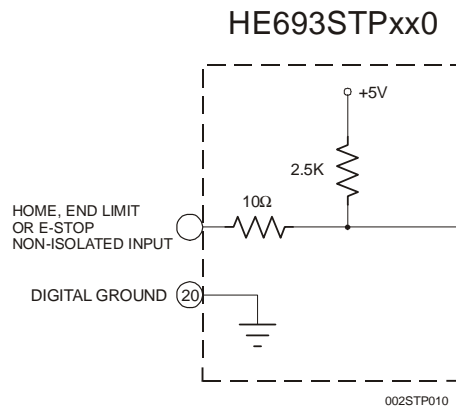


Figure 1.1 – Non-Isolated Input Schematic

2.4.2 Isolated Inputs

For models with isolated inputs (STPxx1, **STP104** and STP113), the **Emergency Stop**, **Home**, **Low Limit**, **High Limit** and **Index** inputs are optically isolated using the following circuitry. **Home**, **Low Limit**, **High Limit** and **Index** inputs can accept either mechanical switches or NPN-type proximity switches.

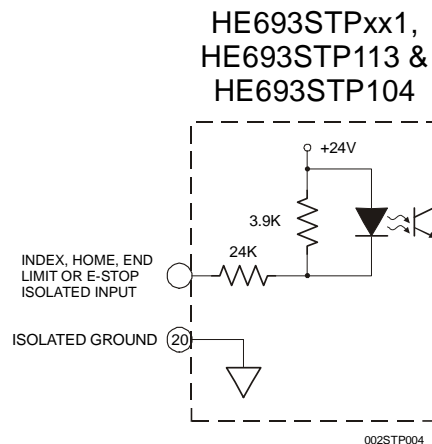


Figure 1.2 – Isolated Input Schematic

2.4.3 Differential Inputs

For models that support encoders (STPx1x), the **Encoder A** and **Encoder B** signals are differential input pairs for connection to encoders with differential (line driver) outputs. In addition, the + inputs can be connected to encoders with single-ended (TTL) outputs.

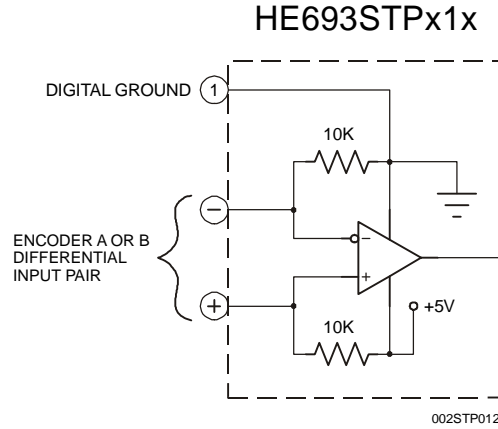


Figure 1.3 – Differential Input Schematic

2.4.4 Differential Outputs

For all models (STPxxx) the **Step** and **Direction** signals are differential output pairs for connection to stepper translator drives with differential (line driver) inputs. In addition, either the + or – outputs can be connected to stepper translator drives with single-ended (TTL) inputs.

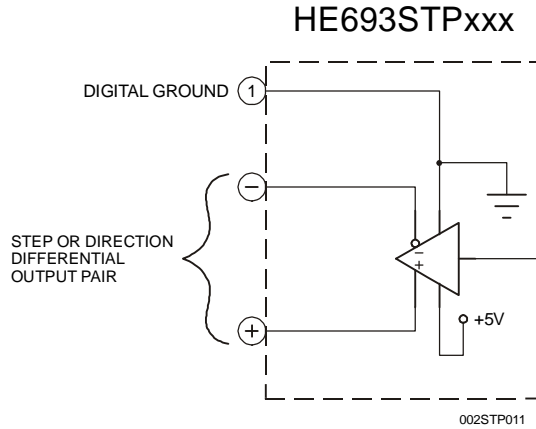


Figure 1.4 – Differential Output Schematic

2.5 I/O Connector Wiring Examples

The following sections provide wiring diagram examples for interconnection between SPM30 Indexer Modules and external equipment, such as switches, encoders and translator drives.

2.5.1 Switch Wiring

Mechanical switches and NPN-type proximity switches can be connected to SPM30 Indexer Modules as shown in Figures 1.5 and 1.6 below.

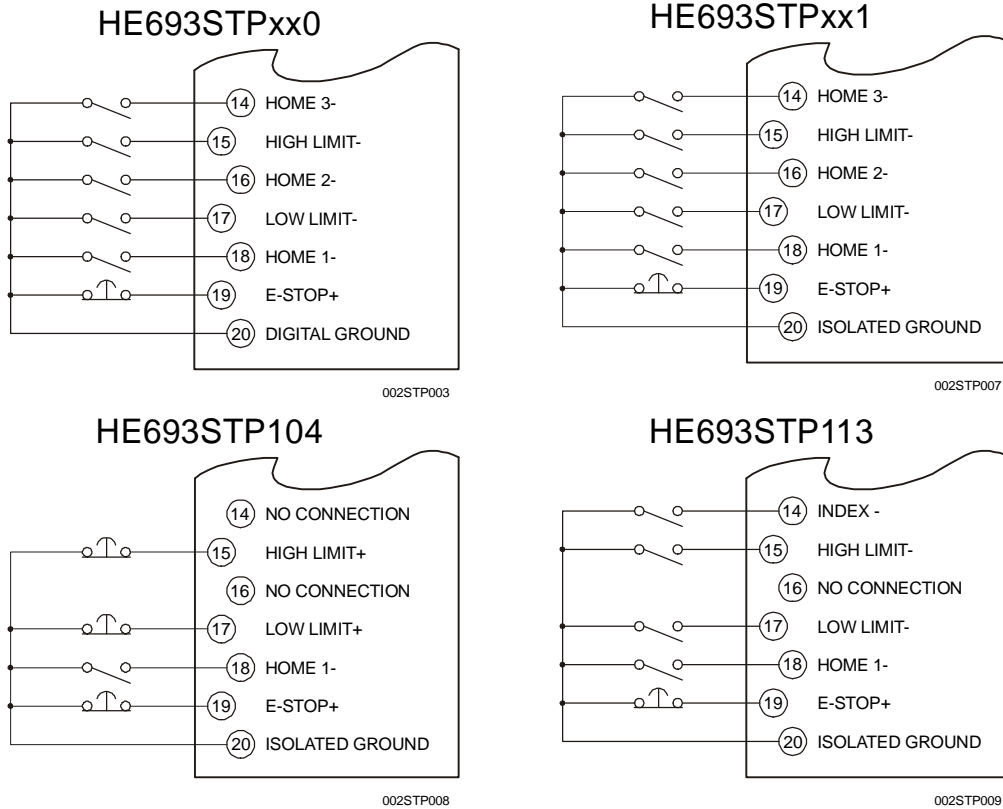


Figure 1.5 – Mechanical Switch Wiring

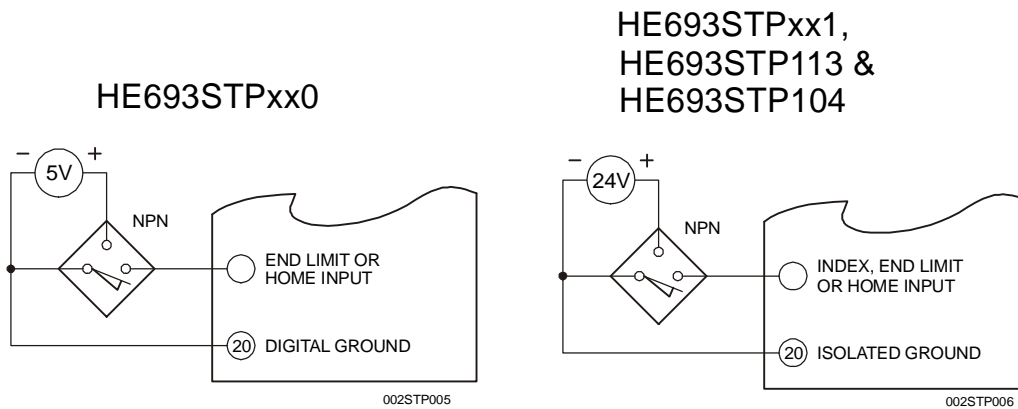


Figure 1.6 – Proximity Switch Wiring

2.5.2 Encoder Wiring

Encoder devices can be optionally connected to those SPM30 models that support encoders (STPx1x). These models are compatible with encoders, which produce either 5VDC differential (line driver), or 5VDC single-ended (TTL) signals.

Differential and single-ended encoders can be connected to SPM30 Indexer Modules as shown in Figure 1.7 below.

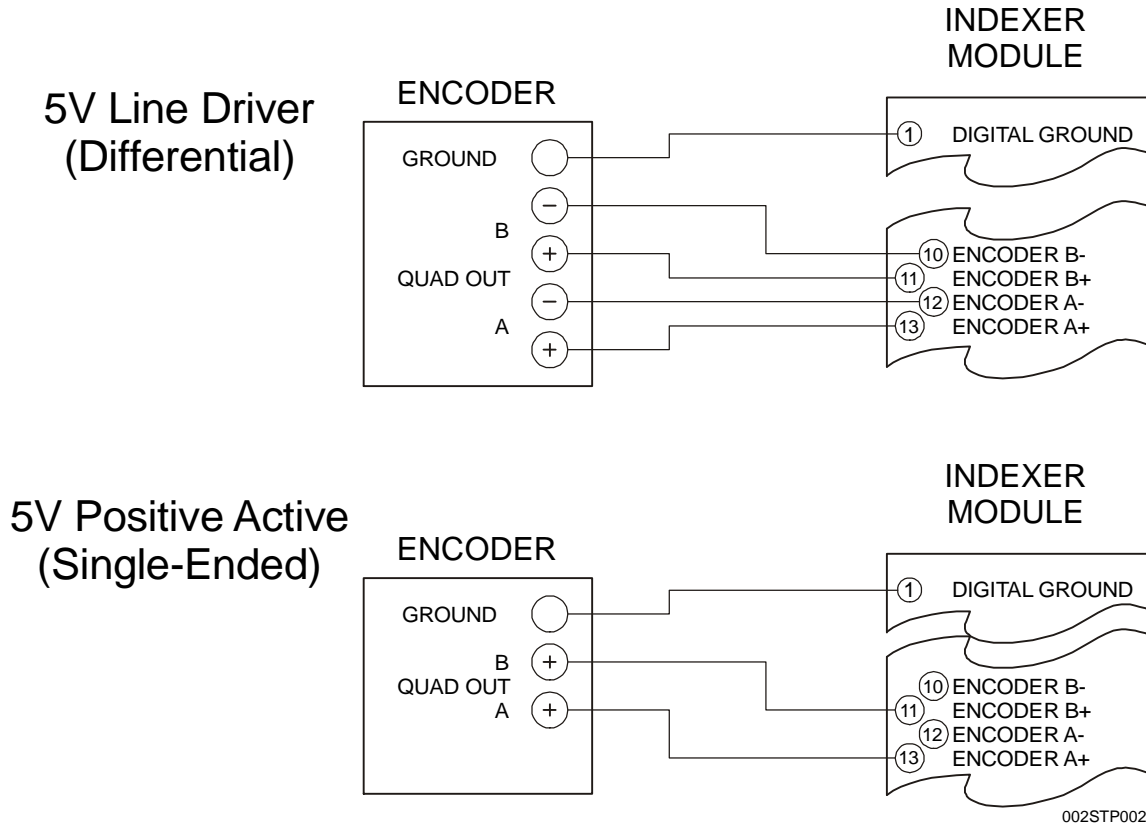


Figure 1.7– Differential and Single-Ended Encoder Wiring

002STP002

2.5.3 Translator Drive Wiring

SPM30 Indexer Modules do not provide the necessary signals to drive a stepper motor directly. A Translator Drive is required. The Translator Drive can be in a separate enclosure, or it can be contained within the motor housing itself. The Translator Drive accepts Step and Direction signals from the SPM30 Module and generates the signals necessary to drive the motor.

SPM30 Indexer Modules are compatible with three common Translator Drive electrical interfaces: differential (line driver), 5VDC positive going and 5VDC negative going. Note that 5VDC signals are also referred to as TTL-compatible signals.

Figures 1.8 below shows the proper wiring for the three different Translator Drive interface types.

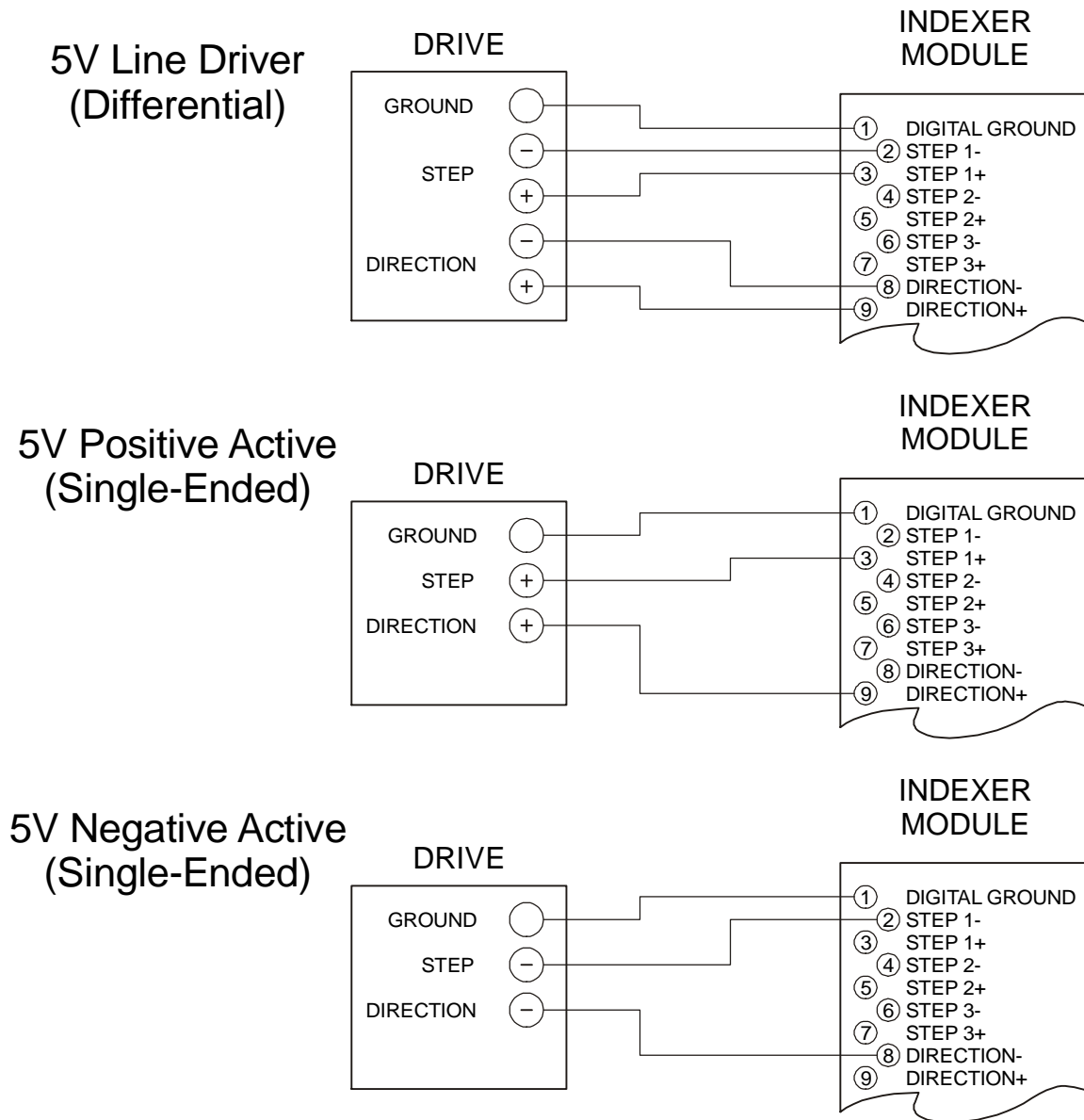


Figure 1.8– Differential and Single-Ended Translator Drive Wiring

002STP001

NOTES

CHAPTER 3: CONFIGURATION

3.1 General

Chapter Three describes the steps necessary to configure the Stepper Positioning Module (SPM30) and the Series 90-30 PLC. Before the SPM30 Module can be used, it must be configured using a Hand-Held Programmer (HHP) or LogicMaster 90 (LM90).

3.2 HHP Configuration

When using a Hand-Held-Programmer:

1. Select configuration mode by pressing the MODE, 4 and ENTER keys.
2. Press the DOWN arrow until the slot containing the stepper module is selected.
3. Press READ, and then the ENTER button.
4. Configure the %I, %Q, %AI and %AQ reference addresses using the numeric and ENTER keys.
5. Press the RIGHT arrow key to display configuration parameters.
6. Use the +/- and ENTER keys to edit the **FIND HOME** and **ENCODER** parameters.
7. Use the numeric and ENTER keys to edit the **STOP DELAY**, **ENCODER MLT**, **ENCODER DIV** and **ENCODER TOL** parameters.

3.3 LM90 Configuration

When using LogicMaster:

1. Connect the serial port of the PC to the PLC's main programming port and run the LM90 software.
2. Enter the Configuration Package from the Main Menu <F2>.
3. Select the proper folder.
4. Choose I/O Configurations from the Configuration Menu <F1>.
5. Cursor over to the slot containing the SPM30 Interface module.
6. Select Other <F8> and Foreign <F3>.
7. The Foreign Module configuration screen appears as follows:

```

Catalog #: FOREIGN

Module ID : 3
%I Ref Adr : %I0001 Byte 1 : 00000001 Byte 9 : 00
%I Size : 16 Byte 2 : xxxxxxxx Byte 10 : 00
%Q Ref Adr : %Q0001 Byte 3 : xx Byte 11 : 00
%Q Size : 16 Byte 4 : xx Byte 12 : 00
%AI Ref Adr: %AI0001 Byte 5 : xx Byte 13 : 00
%AI Size : 4 Byte 6 : xx Byte 14 : 00
%AQ Ref Adr: %AQ001 Byte 7 : xx Byte 15 : 00
%AQ Size : 14 Byte 8 : 00 Byte 16 : 00
%R Ref Adr : %R0001 %R Ref Adr : %R0001
%R(in) Size: 0 %R(out)Size: 0

```

Figure 3.1 – SPM30 Configuration Using LM90 Configuration Software

%I and %Q Ref Adr must be on a byte boundary (1, 9, 17, etc). %AI and %AQ Ref Adr can be set for any valid PLC reference addresses. The remainder of this document assumes reference addresses have been set to %I1, %Q1, %AI1 and %AQ1. Also, depending on which SPM30 model is being configured, %AI Size will be either 2 or 4 and %AQ Size will be 6, 7 or 14, as shown in Table 3.1 on the next page.

3.4 Configurable PLC Resources

Configurable PLC resources, used by SPM30 modules, consist of I/O registers and configuration parameters.

Table 3.1 shows the ten available SPM30 models, along with the PLC resources they use.

Table 3.1 – Models and PLC Resources they Use											
Model	Number of I/O Registers Used				Configuration Parameters Used						
	%I	%Q	%AI	%AQ	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
HE693-			2								
STP100	16	16	4	6	Yes	Yes	Yes	Not used – set to 0			
STP110			2					Yes			
STP300			4	Not used – set to 0							
STP310			2	Yes							
STP101			4	7				Not used – set to 0			
STP111			2					Yes			
STP301			4					Not used – set to 0			
STP311			2	Yes							
STP104			4	Not used – set to 0							
STP113			14	Yes							

Chapter 4 describes the I/O registers used by SPM30 modules. Table 3.2 summarizes the configuration parameters used by SPM30 modules.

Table 3.2 – Configuration Parameter Summary				
Byte	Min	Max	Parameter Name	Description
1	1	1	Reserved	Must always be 00000001
2	0	1	Find Home	Find Home Method (0 = Normal, 1 = Simple)
3	0	255	Stop Delay	Stop Delay Factor
4	0	3	Encoder	Encoder Type (0 = None, 1 = Quad, 2 = U/D, 3 = QuadNM)
5	1	255	Encoder Mlt	Encoder Multiplier
6	1	16	Encoder Div	Encoder Divisor
7	0	255	Encoder Tol	Encoder Tolerance
8 thru 16	0	0	Reserved	Not used - set to 0

Note: All **Min** and **Max** values are shown in decimal. On the LM90 Foreign Module screen (**Fig. 3.1**), configuration parameter Bytes 1 and 2 are entered and displayed in binary while Bytes 3 through 16 are entered and displayed in hexadecimal.

3.5 Configuration Parameter Description

The configuration parameters, summarized in Table 3.2 above, are described in the following sections:

3.5.1 Find Home Method

The **Find Home** parameter selects either the **Normal Method** or **Simple Method**. With the **Normal Method**, when a Find Home command is issued, the SPM30 moves the motor quickly until the Home position is located, then backs up and re-approaches the Home position at a slower rate. With the **Simple Method**, the SPM30 moves the motor until the Home position is located and then stops. The **Simple Method** assumes the motor is initially moving slowly enough, such that it does not overshoot the Home position.

3.5.2 *Stop Delay Factor*

If the %I9 status bit (Current Position Valid) goes OFF at the end of a move, it means that the position validation check has failed. This can be caused by setting Deceleration Time too small, or by excessive system vibration. Due to system vibration, encoder feedback might not be immediately stable at the end of a move. To compensate for this, the **Stop Delay** parameter can be used to require the Encoder Position to be stable for a period of time, before the position validation check is performed. Setting **Stop Delay** to a higher value increases the delay before position validation checking is performed.

3.5.3 *Encoder Type, Encoder Multiplier, Encoder Divisor and Encoder Tolerance*

For SPM30 models, which support encoder feedback (STPx1x), the **Encoder Type**, **Encoder Multiplier**, **Encoder Divisor** and **Encoder Tolerance** parameters should be set appropriately for the application. See Chapter 6 for details.

NOTES

CHAPTER 4: REGISTERS**4.1 General**

Chapter Four defines the PLC I/O registers used by SPM30 modules, and describes how they are used.

For this discussion, it is assumed that the SPM30 module is configured to use PLC I/O registers starting with %I1, %Q1, %AI1 and %AQ1. Table 4.1 summarizes SPM30 register usage.

Table 4.1 – SPM30 Module I/O Register Summary			
Error and Status Bit Input Registers		Command Bit Output Registers	
Register	Description	Register	Description
%I1	Emergency Stop Error	%Q1	Select Axis 1
%I2	Low End Limit Error	%Q2	Select Axis 2
%I3	High End Limit Error	%Q3	Select Axis 3
%I4	Illegal Move Error	%Q4	Find Home Up
%I5	Motor Stalled Error	%Q5	Find Home Down
%I6	Spare Error (future use)	%Q6	Jog Up
%I7	Spare Error (future use)	%Q7	Jog Down
%I8	Power-Up / Watchdog Error	%Q8	Move Relative
%I9	Current Position Valid Status	%Q9	Move Absolute
%I10	Pre-empted Move Resumable Status	%Q10	Resume Move
%I11	Axis 2 Active Status	%Q11	Move Indexed
%I12	Axis 3 Active Status	%Q12	Repeat Move
%I13	Current Axis Home Input Status	%Q13	Set Current Position
%I14	Motor is Accelerating Status	%Q14	Clear Error(s)
%I15	Motor is Decelerating Status	%Q15	Decelerate and Stop
%I16	Motor is Moving Status	%Q16	Immediate Stop
Status Word Input Registers			
Register	Description	Minimum	Maximum
%AI1	Motor Position Lo Word	-8,388,608	+8,388,607
%AI2	Motor Position Hi Word		
%AI3	Encoder Position Lo Word	-8,388,608	+8,388,607
%AI4	Encoder Position Hi Word		
Command Word Output Registers			
Register	Description	Minimum	Maximum
%AQ1	Destination Position Lo Word	-8,388,608	+8,388,607
%AQ2	Destination Position Hi Word		
%AQ3	Velocity Divisor	20	65,535
%AQ4	Base Velocity (must be < %AQ5)	1	8,191
%AQ5	Running Velocity (must be > %AQ4)	1	8,191
%AQ6	Acceleration Time (mS)	1	27,300
%AQ7	Deceleration Time (mS)	0	27,300
%AQ8	Indexed Destination Position Lo Word	1	16,777,215
%AQ9	Indexed Destination Position Hi Word		
%AQ10	Indexed Deceleration Time (mS)	0	27,300
%AQ11	Indexed Window Open Position Lo Word	1	16,777,215
%AQ12	Indexed Window Open Position Hi Word		
%AQ13	Indexed Window Close Position Lo Word	1	16,777,215
%AQ14	Indexed Window Close Position Hi Word		

4.2 SPM30 Module PLC I/O Register Details

This section contains detailed information regarding the SPM30 module's use of PLC I/O registers.

4.2.1 Error and Status Input Bit Register Details (%I1 - %I16)

The SPM30 uses sixteen (16) Digital Input (%I) registers as Error and Status bits as summarized in Table 4.1 above. Table 4.2 shows these bits and the status conditions they represent.

Table 4.2 – Error and Status Input Bit Details		
Register	Description	Condition
%I1	Emergency Stop Error	Emergency Stop input activated during move
%I2	Low End Limit Error	Low Limit input activated during move up
%I3	High End Limit Error	High Limit input activated during move down
%I4	Illegal Move Error	Previous commanded move was <u>not</u> possible
%I5	Motor Stalled Error	Motor is not moving in response to pulses
%I6	Spare Error (future use)	Not Used
%I7	Spare Error (future use)	Not Used
%I8	Power-Up / Watchdog Error	Always ON immediately after power-on or reset
%I9	Current Position Valid Status	Initially OFF; comes ON after successful Find Home or Set Current Position command; set OFF again if motor is stopped suddenly or stalls
%I10	Pre-empted Move Resumable Status	Pre-empted move can be resumed to completion
%I11	Axis 2 Active Status	ON if axis 2 is selected (see %Q2 command)
%I12	Axis 3 Active Status	ON if axis 3 is selected (see %Q3 command)
%I13	Current Axis Home Input Status	Home Input of currently selected axis is active
%I14	Motor is Accelerating Status	Motor is accelerating
%I15	Motor is Decelerating Status	Motor is decelerating
%I16	Motor is Moving Status	Motor is moving

At power-up or after a watchdog timer reset, all error and status bits are OFF except the Power-Up / Watchdog error bit (%I8), which will be ON.

If any error or status bit is ON, it means the corresponding condition is True. Note that if one of the error bits (%I1 - %I8) is ON, an error has occurred. Error bits stay ON till a Clear Error command (%Q14) is issued. No other commands can be issued while an error bit is ON. The status bits (%I9 - %I16) reflect various other SPM30 status conditions and are not affected by the Clear Error command.

4.2.2 Status Word Input Register Details (%A11 - %A14)

The SPM30 use either two or four Analog Input (%AI) status words. All SPM30 models use %A11 and %A12, but only SPM30 models, which support encoder feedback, use %A13 and %A14. These status words are summarized in Table 4.1 above.

4.2.2.1 Motor Position (%A11 and %A12)

The first two status words (%A11 and %A12) are treated as a single 32-bit signed integer value representing the current Motor Position.

Motor Position is continuously updated (up or down) based on step pulses sent by the SPM30 to the stepper motor translator drive. Note that at power-up or after a watchdog timer reset, this value is always zero and is considered invalid. This is reflected by the fact that the Current Position Valid Status bit (%I9) is OFF.

The Motor Position continues to be invalid until a Find Home Up (%Q4), Find Home Down (%Q5) or Set Current Position (%Q13) command is executed successfully. Until this happens, the SPM30 will not obey the Move Absolute command (%Q9).

Note also, that Motor Position can become invalid again if motion stops suddenly, as a result of an Emergency Stop Error (%I1), a Low End Limit Error (%I2), a High End Limit Error (%I3), Motor Stalled Error (%I5), or an Immediate Stop Command (%Q16).

4.2.2.2 Encoder Position (%A13 and %A14)

For SPM30 models, which support encoder feedback, the %A13 and %A14 status words are treated as a single 32-bit signed integer value representing current Encoder Position for axis 1.

Encoder Position is continuously updated (up or down) based on feedback pulses sent by the encoder to the SPM30. Note that at power-up or after a watchdog timer reset, this value is always zero and is considered invalid. This is reflected by the fact that the Current Position Valid Status bit (%I9) is OFF.

The Encoder Position continues to be invalid until a Find Home Up (%Q4), Find Home Down (%Q5) or Set Current Position (%Q13) command is executed successfully.

If the SPM30 is properly configured (see chapter 6), Encoder Position will track axis 1's Motor Position. Note that when Encoder Position doesn't match Motor Position exactly, a position validation error has been detected. There are several possible causes for this error (see section 6.4). Some position validation errors can't be avoided, which is why the SPM30 supports an **Encoder Tolerance** configuration parameter.

4.2.3 Command Word Output Register Details (%AQ1 - %AQ14)

Depending on model, the SPM30 uses 6, 7 or 14 %AQ registers. The %AQ command words act as qualifiers for the %Q command bits. Typically, the %AQ command words are set to appropriate values, and then one of the %Q command bits is changed from OFF to ON.

Normally, after the SPM30 starts executing a command, the %AQ command words can be changed to prepare for the next command without affecting the command in progress. The exception to this rule is, that the Destination Position (%AQ1 and %AQ2) must not be disturbed during a Find Home Up or Find Home Down command, until either an error occurs, or the Current Position Valid status bit goes ON.

4.2.3.1 Destination Position (%AQ1 and %AQ2)

The first two command words (%AQ1 and %AQ2) are treated as a single 32-bit signed integer value representing the Destination Position.

For the Find Home Up, Find Home Down and Set Current Position commands, the Destination Position is the value to be loaded into Motor Position and Encoder Position when the command completes successfully.

For the Move Absolute command, the Destination Position is the absolute position to move to.

For the Move Relative and Move Indexed commands, the Destination Position is the relative distance to move above or below the current position.

4.2.3.2 Velocity Divisor (%AQ3)

This command word determines the resolution (multiplier) of the Base Velocity (%AQ4) and the Running Velocity (%AQ5) command words.

The velocity multiplier (M_v), ranges from .01 pulses per second to 30 pulses per second and is calculated as follows:

$$M_v = \frac{600}{\%AQ3} \quad (\text{Pulses per second})$$

Table 4.3 shows some useful %AQ3 settings along with their resulting velocity multipliers and corresponding maximum velocities.

Table 4.3 - Example Velocity Multipliers		
Velocity Divisor (%AQ3)	Velocity Multiplier (M_v)	Maximum Velocity (Pulses Per Second) ($M_v \times 8191$)
20	30.0	245,730.0
60	10.0	81,910.0
120	5.0	40,995.0
300	2.0	16,382.0
600	1.0	8,191.0
1200	0.5	4,095.5

4.2.3.3 Base Velocity (%AQ4)

This command word determines the velocity the SPM30 starts at when executing one of the motion commands (%Q4 through %Q11).

A typical move starts at the Base Velocity and accelerates to the Running Velocity. Then, if the move ends normally, it decelerates from Running Velocity to Base Velocity, and then stops.

Also, near the end of a Find Home Up or Find Home Down command, the motor moves at a constant Base Velocity while searching for the exact Home Position.

Actual Base Velocity (V_B) depends on the Velocity Divisor (%AQ3) and is calculated according to the following formula:

$$V_B = \%AQ4 \times M_V = \%AQ4 \times \frac{600}{\%AQ3} \quad (\text{Pulses per second})$$

4.2.3.4 Running Velocity (%AQ5)

This command word determines the maximum velocity the motor moves when the SPM30 finishes accelerating.

The Running Velocity must be greater than the Base Velocity.

Actual Running Velocity (V_R) depends on the Velocity Divisor (%AQ3) and is calculated according to the following formula:

$$V_R = \%AQ5 \times M_V = \%AQ5 \times \frac{600}{\%AQ3} \quad (\text{Pulses per second})$$

Note that if the Destination Position parameter is too short to accommodate the Acceleration Time and Deceleration Time parameters, the motor never reaches the Running Velocity and the move becomes triangular.

4.2.3.5 Acceleration Time (%AQ6)

This command word determines the maximum time (in mS) spent accelerating from the Base Velocity to the Running Velocity during a move.

Note that if it is time to start decelerating before acceleration to the Running Velocity is complete, the acceleration time is decreased and the velocity profile becomes triangular.

Also note that the maximum useful value for Acceleration Time (%AQ6_{MAX}) is dependent on the Base Velocity (%AQ4) and Running Velocity (%AQ5) according to the following formula:

$$\%AQ6_{MAX} \approx \frac{\%AQ5 - \%AQ4}{0.3} \quad (\text{Milliseconds})$$

4.2.3.6 Deceleration Time (%AQ7)

For SPM30 models, which support programmable deceleration, this command word determines the maximum time (in mS) spent decelerating from the Running Velocity to the Base Velocity at the end of a move. If %AQ7 is zero, deceleration time is the same as the acceleration time set via %AQ6.

Note that if it is time to start decelerating before acceleration to the Running Velocity is complete, the deceleration time is decreased, and the velocity profile becomes triangular.

Also note that the maximum useful value for Deceleration Time (%AQ7_{MAX}) is dependent on the Base Velocity (%AQ4) and Running Velocity (%AQ5) according to the following formula:

$$\%AQ7_{MAX} \approx \frac{\%AQ5 - \%AQ4}{0.3} \quad (\text{Milliseconds})$$

4.2.3.7 Indexed Destination Position (%AQ8 and %AQ9)

For STP113 only, these two command words (%AQ8 and %AQ9) are treated as a single 32-bit unsigned integer value representing the Indexed Destination Position.

The Indexed Destination Position contains the number of steps to continue in the current direction after the SPM30 detects external activation of the Index Input during a Move Indexed command (%Q11).

4.2.3.8 Indexed Deceleration Time (%AQ10)

For STP113 only, this command word determines the time (in mS) spent decelerating from the Running Velocity to the Base Velocity at the end of a Move Indexed command (%Q11) *after* the Indexed Input has been asserted. If %AQ10 is zero, deceleration time is the same as the normal deceleration time set via %AQ7.

4.2.3.9 Indexed Window Open Position (%AQ11 and %AQ12)

For STP113 only, these two command words (%AQ11 and %AQ12) are treated as a single 32-bit unsigned integer value representing the Window Open Position during a Move Indexed command (%Q11).

The Indexed Window Open Position contains the number of steps past the beginning of an Indexed Move when the Index Input Window is opened. While the Index Input Window is open, the SPM30 module monitors the external Index Input. While the Index Input Window is closed, the SPM30 module ignores the external Index Input.

4.2.3.10 Indexed Window Close Position (%AQ13 and %AQ14)

For STP113 only, these two command words (%AQ13 and %AQ14) are treated as a single 32-bit unsigned integer value representing the Window Close Position during a Move Indexed command (%Q11).

The Indexed Window Close Position contains the number of steps past the beginning of an Indexed Move when the Index Input Window is closed. While the Index Input Window is open, the SPM30 module monitors the external Index Input. While the Index Input Window is closed, the SPM30 module ignores the external Index Input.

4.2.4 Command Bit Output Register Details (%Q1 - %Q16)

The SPM30 uses sixteen (16) Digital Output (%Q) registers, which are used as Command bits. These registers are summarized in Table 4.1 above.

Most commands are positive (OFF-to-ON) edge sensitive. This means that the SPM30 obeys the command only when the module sees the command bit transition from OFF to ON. The SPM30 can require more than one PLC logic scan to detect a command bit, but takes no more than 4 mS to start executing the command. The command bit must then be turned OFF for at least 4 mS before the next command (or the same command) can be again issued.

The Jog Up and Jog Down commands are level sensitive. The Jog commands start when the OFF-to-ON transition is recognized and remain running so long as the command bit stays ON. The Jog commands decelerate and stop when the ON-to-OFF transition is recognized.

The Power-Up / Watchdog Error Status bit (%I8) is always ON immediately after Power-On or reset. Therefore, the first command issued *must* be the Clear Errors command (%Q14).

Also, some commands are ignored depending on the state of other status bits. For example, if the Moving Status bit (%I16) is ON, the only legal commands are Decelerate and Stop (%Q15) and Immediate Stop (%Q16).

If more than one command bit goes active during any one CPU Scan Time, the command with the highest bit number takes priority. Note that this gives the Immediate Stop command (%Q16) the highest priority.

NOTES

CHAPTER 5: COMMANDS

5.1 General

Chapter Five describes the commands available for the SPM30 and their operations.

5.2 Issuing Commands

Commands are issued by setting appropriate %Q bits and are subject to the strict conditions as described in the individual command descriptions below.

The SPM30 does not accept commands if any of the Error Status bits are ON. The exception is, of course, the Clear Errors command (%Q14). If an error occurs, the only command that can be issued is the Clear Errors command. Note that immediately following a power-up or reset, the Power-Up / Watchdog Error bit is ON. Thus, the first command issued after a power on or reset *must* be the Clear Errors command. This is a safety feature, which prevents the SPM30 from running until the application program is in control, and specifically issues the Clear Errors command.

5.3 Available Commands

5.3.1 Select Axis Commands

Command Bits:	%Q1, %Q2 and %Q3
Status Bits Required:	%I1 through %I8 and %I16 must be OFF
Status Bits Affected:	%I9 and %I13
Status Words Required:	%AI1 and %AI2
Status Words Affected:	%AI1 and %AI2
Command Words Required:	None

These commands are used to select a new axis to be acted upon by the other commands as follows:

%Q1	Selects axis 1 to be the new current axis
%Q2	Selects axis 2 to be the new current axis
%Q3	Selects axis 3 to be the new current axis

At power-up or after a watchdog timer reset, axis 1 is selected by default. %Q2 and %Q3 are illegal commands for single-axis models (STP1xx).

When a new axis is selected, the following search sequence takes place:

1. If requested new axis is illegal or is the same as the old axis, do nothing.
2. Otherwise, save old axis' Motor Position, Current Position Valid and At Home status.
3. Set axis multiplexer for the new axis.
4. Restore last known Motor Position, Current Position Valid and At Home status for the new axis.
5. If the new axis' At Home status has changed since the last time the axis was selected, its Current Position Valid status is turned OFF.

5.3.2 Find Home Up and Find Home Down Commands

Command Bits:	%Q4 and %Q5
Status Bits Required:	%I1 through %I8 and %I16 must be OFF
Status Bits Affected:	%I1 through %I5, %I9 and %I13 through %I16
Status Words Required:	None
Status Words Affected:	%AI1 through %AI4
Command Words Required:	%AQ1 through %AQ6

These commands are used to search for the current axis' Home Input reference position as follows:

- %Q4 Searches for Home Position in the Up direction
- %Q5 Searches for Home Position in the Down direction

When searching for Home Position, the following search sequence takes place:

1. The Current Position Valid status bit (%I9) is turned OFF (position not valid).
2. The current axis is moved normally (starts at Base Velocity and accelerates to Running Velocity) in the selected direction (up for %Q4 or down for %Q5).
3. When the axis' Home Input becomes active, motion is stopped immediately.
4. If the **Simple Method** was selected during configuration, skip to step 8. **NOTE:** The **Simple Method** variation assumes that the user-programmed Running Velocity is slow enough to prevent overshoot.
5. Otherwise, just in case we shot right past the Home Input, the axis is run at Base Velocity in the opposite direction till the Home Input is active again.
6. Motion then continues in the same direction as in step 5 above, (still at Base Velocity), till the Home Input is inactive for 50 mS.
7. Then the direction is reversed again, and the motor is moved at Base Velocity till the Home Input is active, at which time the motor stops precisely at the Home Position.
8. Destination Position is copied into Motor Position (and into Encoder Position is axis 1 is selected), and the Current Position Valid Status bit (%I9) is turned ON.

If the current axis is 1, and the Encoder configuration parameter is set to Quad, the Find Home command is **Limit-Switch-Qualified**, which changes step 3 above as follows:

3. When the End Limit input for the direction we're moving becomes active, motion is stopped immediately.

NOTE: The **Limit-Switch-Qualified** variation is most useful when the encoder's Marker output is connected to the axis 1's Home Input. Since the Marker on a rotary encoder typically occurs several times during a full stroke move, this technique allows the Marker, which occurs closest to the Limit Switch, to be used as the Home Position.

For best results, the Marker-to-Limit-Switch relationship needs to be mechanically adjusted such that the Marker occurs at approximately one-half of the encoder's revolution away from the Limit Switch.

5.3.3 Jog Up and Jog Down Commands

Command Bits:	%Q6 and %Q7
Status Bits Required:	%I1 through %I8 and %I16 must be OFF
Status Bits Affected:	%I1 through %I5 and %I13 through %I16
Status Words Required:	None
Status Words Affected:	%AI1 through %AI4
Command Words Required:	%AQ3 through %AQ7

These commands are used to perform manual jogging on the current axis as follows:

%Q6	Starts a manual jog move in the Up direction
%Q7	Starts a manual jog move in the Down direction

When one of the Jog Command bits goes ON, the current axis starts in the selected direction at the Base Velocity and accelerates to the Running Velocity.

The motor continues to move at Running Velocity until the Jog Command bit goes OFF again. At that time, the motor decelerates from Running Velocity back down to Base Velocity, and then stops.

If the Jog Command bit transitions from ON to OFF before the motor reaches Running Velocity, the motor acceleration and deceleration times are decreased, and the velocity profile is triangular.

During a Jog Up or Down Move, the velocity can be changed on-the-fly, by setting a new Running Velocity value (%AQ5) and then setting the %Q12 bit ON.

5.3.4 Move Relative and Move Absolute Commands

Command Bits:	%Q8 and %Q9
Status Bits Required:	%I1 through %I8 and %I16 must be OFF; %I9 must be ON for %Q9 command
Status Bits Affected:	%I1 through %I5 and %I13 through %I16
Status Words Required:	%AI1 and %AI2 used by %Q9 command
Status Words Affected:	%AI1 through %AI4
Command Words Required:	%AQ1 through %AQ7

These commands perform a relative or absolute move on the current axis as follows:

%Q8	Performs a relative move
%Q9	Performs an absolute move

These commands perform a programmed move in the up or down direction to a relative target position.

If doing a relative move (%Q8), the relative target position is taken directly from Destination Position (%AQ1 and %AQ2) and it **is not** necessary for the Current Position Valid Status bit (%I9) to be On. The direction of the move is determined by the Destination Position sign (+ for move up or - for move down).

If doing an absolute move (%Q9), the relative target position is calculated as the difference between Destination Position (%AQ1 and %AQ2) and Motor Position (%AI1 and %AI2), and therefore it **is** necessary for the Current Position Valid Status bit (%I9) to be On. The direction of the move depends on whether Destination Position is higher (move up) or lower (move down) than Motor Position.

Normally, the move starts at Base Velocity and accelerates to Running Velocity until it is time to decelerate back down to Base Velocity, and then stop. This type of move is said to have a trapezoidal velocity profile.

If it is time to start decelerating before accelerating to Running Velocity has completed, the motor starts decelerating at that point. In this case, the acceleration and deceleration times are decreased, and the velocity profile becomes triangular.

For some SPM30 models, the End Limit Inputs can be used to start relative and absolute moves. The High Limit Input (HL-) can be used to start moves in the Up direction while the Low Limit Input (LL-) can be used to start moves in the Down direction. To accomplish this, the ladder program needs to set up the %AQ values as for a normal relative or absolute move, and the appropriate End Limit Input needs to be activated before %Q8 or %Q9 is turned ON. In this case, the actual move does not start until the End Limit input is deactivated.

5.3.5 Resume Move Command

Command Bit:	%Q10
Status Bits Required:	%I1 through %I8 and %I16 must be OFF %I10 must be ON
Status Bits Affected:	%I1 through %I5, %I10 and %I13 through %I16
Status Words Required:	None
Status Words Affected:	%AI1 through %AI4
Command Words Required:	None

This command can be used to resume a previously pre-empted Relative Move or Absolute Move.

If a Move Relative or Move Absolute command was previously pre-empted by a Decelerate and Stop command (%Q15), and no other commands have been issued since then, the Pre-empted Move Resumable Status bit (%I10) is ON.

In this case, the Resume Move command can be issued to restart the pre-empted move from where it left off. Of course, this action turns the Pre-empted Move Resumable Status bit back OFF.

Note that the resume logic is such that a move can be pre-empted and resumed any number of times until one of the following occurs:

1. The move reaches its originally programmed relative target position,
2. An error occurs (such as End Limit or Emergency Stop),
3. Some command other than %Q10 is issued after a move is pre-empted.

The Resume Move command is especially useful for manually-assisted programmed moves.

For example, the machine operator can trigger a Move Absolute command by pressing a footswitch. If the footswitch continues to be pressed, the move continues until it reaches its programmed target position.

However, at the operator's option, the footswitch can be released causing a Decelerate and Stop command to be issued. There are a number of reasons why the operator might decide to do this, such as to make a mechanical adjustment or to manually reposition the material being moved.

Then, when the operator is ready, the footswitch can be pressed again sending a Resume Move command to the SPM30 to complete the motion.

5.3.6 Move Indexed Command

Command Bit:	%Q11
Status Bits Required:	%I1 through %I8 and %I16 must be OFF
Status Bits Affected:	%I1 through %I5 and %I13 through %I16
Status Words Required:	None
Status Words Affected:	%AI1 through %AI4
Command Words Required:	%AQ1 through %AQ14

Available only for the STP113 model, this command performs an Indexed Move, which is similar to a Relative Move (%Q8), except that the final destination can be modified if the external Index Input becomes active. This modified behavior is determined by the user-programmed values in %AQ8 through %AQ14. Note that %AQ8 through %AQ14 are ignored by all other STP113 commands.

For details on %AQ8 through %AQ14 values, refer to Table 4.1 and to Section 4.2.3 in Chapter 4.

As stated above, the Indexed Move is similar to the Relative Move. In fact, if the Index Input does not become active during an Indexed Move, it behaves exactly like a Relative Move.

The purpose, of the Move Indexed command, is to provide the ability to synchronize motor movement with an external mechanical reference point. This reference point (or index) is provided to the STP113 module via the Index Input (pin 14 INDEX- signal). The Index Input will only be processed if it occurs during a user-specified window. When the Index Input is accepted, the STP113 then switches to an alternate destination and an alternate deceleration time.

When the %Q11 bit goes ON, the motor starts moving just as it would for a Relative Move. However, during the time the Indexed Window is open (determined by %AQ11 through %AQ14), the STP113 monitors the Index Input. If the Index Input becomes active while the Indexed Window is open, a new end point for the move is determined. This new end point is defined to be N more steps (in the same direction) beyond the point at which the Index Input became active, where N is the Indexed Destination Position (%AQ8 and %AQ9) value. In addition, if %AQ10 is not zero, it sets a new deceleration rate called the Indexed Deceleration Time.

The resulting Indexed Move can be longer or shorter than the default Relative Move. The motor must attain Running Velocity before the window is opened. If the move does not reach Running Velocity the window won't be opened, and the Index Input is not accepted.

If the Relative Move reaches the point to begin decelerating before the Index Input is activated, the window is closed and Index Input is not accepted.

NOTE: A Move Indexed command has one more difference compared to other move commands. In applications where all moves are in the same direction, the **DR+** and **DR-** outputs can be used as **MOVING+** and **MOVING-** outputs, indicating *Move in Progress* to some external device.

At the start of a Relative, Absolute or Indexed move, **DR+** and **DR-** are set according to the direction of the move. At the end of a Relative and Absolute move, the **DR+** and **DR-** outputs do not change state. However, at the end of an Indexed Move, **DR+** and **DR-** are always taken Low and High respectively. Therefore, when an Indexed Move starts in the Up direction, **DR+** and **DR-** start out High and Low respectively, and then switch to Low and High respectively when the move completes.

5.3.7 Repeat Move Command

Command Bit: %Q12

The Repeat Move command can be used to automatically repeat a Relative Move (%Q8) or an Indexed Move (%Q11) command, or to perform an on-the-fly velocity change during a Jog Up Move (%Q6) or Jog Down Move (%Q7).

When a Relative Move or Indexed Move completes normally, the SPM30 checks the %Q12 bit. If the %Q12 bit is ON, the SPM30 automatically repeats the move just completed. As long as the %Q12 bit stays ON, the move is repeated immediately after completion of each move. If the %Q12 bit is turned OFF during a move, the current move will still complete normally, but it will not be repeated.

During a Jog Up or Down Move, the velocity can be changed on-the-fly, by setting a new Running Velocity value (%AQ5) and then setting the %Q12 bit ON.

5.3.8 Set Current Position Command

Command Bit: %Q13

Status Bits Required: %I1 through %I8 and %I16 must be OFF
Status Bits Affected: %I9
Status Words Required: None
Status Words Affected: %AI1 through %AI4
Command Words Required: %AQ1 and %AQ2

This command is used to manually set the current position.

When the Set Current Position command is issued, Destination Position (%AQ1 and %AQ2) is copied into Motor Position (%AI1 and %AI2) and Encoder Position (%AI3 and %AI4), and the Current Position Valid Status bit (%I9) is turned ON.

This command can be used in conjunction with the Jog Up and Jog Down commands as an alternative to the Find Home Up and Find Home Down commands for finding and setting a reference position.

There is no motor movement associated with this command.

5.3.9 *Clear Error(s) Command*

Command Bit:	%Q14
Status Bits Required:	None
Status Bits Affected:	%I1 through %I8
Status Words Required:	None
Status Words Affected:	None
Command Words Required:	None

This command is used to clear errors previously detected by the SPM30. When this command is issued, all Error bits (%I1 through %I8) are cleared (turned OFF).

Note that when an Error bit is ON, the SPM30 does not obey any other commands till the error is cleared via the Clear Error(s) command.

5.3.10 *Decelerate and Stop Command*

Command Bit:	%Q15
Status Bits Required:	None
Status Bits Affected:	%I10, %I13 through %I16
Status Words Required:	None
Status Words Affected:	%AI1 through %AI4
*Command Words Required:	None

This command causes the controller to decelerate and stop the motor.

If the motor is moving when this command is issued, the motor decelerates till it reaches the Base Velocity, and then it stops.

If this command pre-empts a Move Relative or Move Absolute command, the Pre-empted Move Resumable Status bit (%I10) is turned ON, unless an error occurs. In this case, the original move can be resumed from where it left off by issuing a Resume Move command (%Q10).

5.3.11 *Immediate Stop Command*

Command Bit:	%Q16
Status Bits Required:	None
Status Bits Affected:	%I9, %I13 through %I16
Status Word Required:	None
Status Word Affected:	%AI1 through %AI4
Command Words Required:	None

This command is used to cause the motor to stop immediately. When this command is issued, the motor stops as quickly as possible.

If the motor was moving, the Current Position Valid Status bit (%I9) is turned OFF (unless encoder feedback indicates that the position error is less than the Encoder Tolerance configuration parameter).

CHAPTER 6: ENCODERS

6.1 General

Chapter Six details the use of optional encoder feedback with the SPM30.

SPM30 models, which support encoder feedback (STPx1x), can be configured for a variety of Incremental Encoder feedback options for axis 1. Absolute Encoders and Resolvers are not compatible with the SPM30.

The type of encoder used and the relationship of the SPM30 step pulses to the encoder feedback pulses are set via the Encoder Type, Encoder Multiplier, Encoder Divisor and Encoder Tolerance configuration parameters.

6.2 Encoder Type Configuration

The type of encoder used is determined by the Encoder Type configuration parameter as follows:

Encoder Type	Description
None	No encoder feedback
Quad	Incremental quadrature encoder with Marker output
U/D	Incremental up/down encoder
QuadNM	Incremental quadrature encoder without Marker output

If the Encoder Type is configured to **None**, the Encoder Position Status Words (%AI3 and %AI4) can be ignored, and the Motor Stalled Error bit (%I5) never goes ON.

Otherwise, if the Encoder Type is configured for **Quad** (or **QuadNM**) or **U/D**, the Encoder Position Status Words (%AI3 and %AI4) are updated as a result of feedback pulses from the Encoder to the SPM30, thus allowing Motor Position validation and Motor Stall detection.

6.2.1 Quadrature Encoder

Quadrature Encoders (**Quad** or **QuadNM**) are the most common position feedback devices used in motion control. This type of encoder outputs two square wave signals (A and B), which are 90° out of phase with each other. The SPM30 determines the direction of motion based on which signal lags behind the other.

A Quadrature Encoder's resolution can be effectively doubled or quadrupled by the SPM30's quadrature decoding hardware. This occurs automatically if the Encoder Multiplier is a multiple of 2 or 4 respectively.

A Quadrature Encoder can also provide a third signal, called a Marker. This signal is generated once per revolution, can be used by the SPM30 as a reference position when connected to the SPM30's axis 1 Home Input via a level-shifter / adapter such as the HE693MKR100. If the encoder's marker signal is connected in this way, the Encoder Type should be configured to **Quad**, otherwise, use **QuadNM**.

6.2.2 Up / Down Encoder

Up / Down Encoders (**U / D**) also provide A and B signals but not both at the same time. Typically, if the motor is turning clockwise, the A output is active. If the motor is turning counterclockwise, the B output is active. The SPM30 increments the position count when A is active, and decrements the position count when B is active.

An SPM30 configured for an Up / Down Encoder is especially useful for unidirectional motion control, in which some mechanical event provides the feedback, such as a proximity detector monitoring gear teeth.

6.3 Encoder Multiplier and Divisor Configuration

Before encoder feedback can be used for Motor Position validation or Motor Stall detection, the SPM30 must know the ratio of motor step pulses to encoder feedback pulses. This is accomplished by setting the Encoder Multiplier and the Encoder Divisor to values, which are appropriate for the user-supplied hardware.

Step Resolution, specified in micro-steps per revolution, indicates the number of step pulses which must be sent by the SPM30, to move the stepper motor one revolution. This value is determined by the connected translator drive, and is switch or jumper selectable on some drives.

Encoder Resolution, specified in lines per revolution, indicates the number of feedback pulses the encoder sends to the SPM30 during one revolution of motion. This value is determined by the encoder itself.

For a given Step Resolution and Encoder Resolution, the following formula can be used to determine the proper settings for Encoder Multiplier and Encoder Divisor.

$$\frac{\text{EncoderMultiplier}}{\text{EncoderDivisor}} = \frac{\text{Step Resolution}}{\text{Encoder Resolution}}$$

Where: Encoder Multiplier is any value from 1 to 255
Encoder Divisor is any value from 1 to 16

If more than one combination of Encoder Multiplier and Encoder Divisor satisfies the formula, choose the combination with the lowest values for Encoder Multiplier and Encoder Divisor (reduce the fraction).

6.3.1 Encoder Multiplier and Divisor Configuration Example

Assume that an Encoder Resolution of 1000 lines per revolution is used. Further, it is assumed the encoder is mechanically connected to the stepper motor shaft and is electrically connected to the SPM30's Encoder A and Encoder B inputs.

For this example, the following table shows the proper configuration settings for Encoder Multiplier and Encoder Divisor for 16 typical Step Resolutions:

Step Resolution	Encoder Multiplier	Encoder Divisor	Step Resolution	Encoder Multiplier	Encoder Divisor
200	1	5	20,000	20	1
400	2	5	21,600	108	5
1,000	1	1	25,000	25	1
2,000	2	1	25,400	127	5
5,000	5	1	25,600	128	5
10,000	10	1	36,000	36	1
12,800	64	5	50,000	50	1
18,000	18	1	50,800	254	5

6.4 Encoder Tolerance Configuration

When an encoder feedback device is connected to the SPM30, it can be used to verify successful execution of motion commands for axis 1.

Ideally, Encoder Position and Motor Position would always match exactly. However, when Encoder Position and Motor Position become skewed, it means that one or more of the following errors has occurred:

1. A change in motor direction produced a backlash error caused by the mechanical linkage between the motor and the encoder.
2. A resolution error has occurred, because the encoder is less precise than the motor drive's micro-stepping rate. In other words, Step Resolution is greater than Encoder Resolution. This is quite common for Step Resolutions greater than 2000 steps per revolution. For most motors, Step Resolutions greater than 2000 do not afford better positioning accuracy, but merely offer smoother operation. In these cases, it does not make sense to spend the extra money for an extremely fine-pitched encoder.
3. The Step Resolution exceeds the motor's ability to accurately position its rotor. For example, most stepper motors have an accuracy of 1 part in 2000, which means a Step Resolution greater than 2000 has little affect on accuracy, but only contributes to smoother operation.
4. The motor missed some step pulses or stalled because of low or mid-frequency resonance.
5. The motor missed some step pulses or stalled, because the programmed Acceleration Time or Deceleration Time was too low to properly overcome the mechanical inertia of the system.
6. Some external force changed the position of the motor. For example, something caught in the machinery does not let the motor move the load as requested.

The SPM30 can be configured to use encoder feedback pulses, to validate motor position, and to detect motor stalling. This is accomplished when the Encoder Tolerance configuration parameter is non-zero.

6.4.1 Motor Position Validation

If Encoder Tolerance is set to a number from 1 to 255, Encoder Feedback is used to validate current position. As such, the SPM30 compares the Encoder Position value (%AI3 and %AI4) to the Motor Position value (%AI1 and %AI2). If the difference exceeds the configured Encoder Tolerance value, a position validation error has been detected, and the SPM30 turns the Current Position Valid Status bit (%I9) OFF.

During motion, a position validation error does not automatically stop the move. It is up to the ladder program to monitor the Current Position Valid bit, and then to issue the Immediate Stop command, if desired.

6.4.2 Motor Stall Detection

Also, when Encoder Tolerance is non-zero, the SPM30 uses the encoder feedback pulses to verify motor velocity during motion, which allows the SPM30 to detect a stalled motor.

In particular, if the measured move velocity falls to less than 10% of the theoretical move velocity, then a Motor Stalled error has been detected. When this happens, the SPM30 stops sending step pulses (stopping the motor) and turns the Motor Stalled Error Status bit (%I5) ON.

INDEX

Acceleration Time	29	Encoders.....	41
Base Velocity	29	Adding.....	41
Command Bit Output Register	31	Multiplier and Divisor Value Examples	42
Command Word Output Registers	28	Types of Encoders	41
Commands.....	33	Error and Status Input Bit Register.....	26
Available Commands.....	33	External Wiring	13
Clear Errors	40	Features.....	9
Decelerate and Stop.....	40	Indexed Deceleration Time.....	30
Find Home Up / Find Home Down	34	Indexed Window Close Position.....	30
Immediate Stop	40	Indexed Window Open Position	30
Jog Up and Jog Down	35	Indicators	10
Move Indexed	38	Isolated Inputs	15
Move Relative and Move Absolute.....	36	Motion Profiles.....	12
Repeat Move	39	Motor Position.....	27
Resume Move	37	Quadrature Encoder	41
Select Axis.....	33	Register Details	26
Set Current Position	39	Running Velocity.....	29
Deceleration Time	30	Signal Descriptions	15
Destination Position	28	Specifications.....	11
Differential Inputs	16	Up / Down Encoder	41
Differential Outputs	16, 19	Velocity Divisor	28
Encoder Position	27		