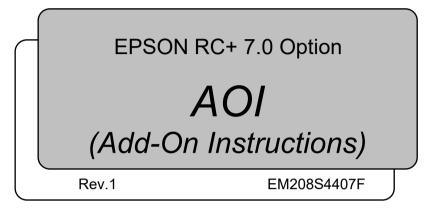
EPSON



EPSON RC+ 7.0 Option AOI (Add-On Instructions) Rev.1

EPSON RC+ 7.0 Option

AOI (Add-On Instructions)

Rev.1

Copyright © 2020 SEIKO EPSON CORPORATION. All rights reserved.

FOREWORD

Thank you for purchasing our robot products.

This manual contains the information necessary for the correct use of the Manipulator. Please carefully read this manual and other related manuals before installing the robot

system.

Keep this manual handy for easy access at all times.

WARRANTY

The robot and its optional parts are shipped to our customers only after being subjected to the strictest quality controls, tests, and inspections to certify its compliance with our high performance standards.

Product malfunctions resulting from normal handling or operation will be repaired free of charge during the normal warranty period. (Please ask your Regional Sales Office for warranty period information.)

However, customers will be charged for repairs in the following cases (even if they occur during the warranty period):

- 1. Damage or malfunction caused by improper use which is not described in the manual, or careless use.
- 2. Malfunctions caused by customers' unauthorized disassembly.
- 3. Damage due to improper adjustments or unauthorized repair attempts.
- 4. Damage caused by natural disasters such as earthquake, flood, etc.

Warnings, Cautions, Usage:

- 1. If the robot or associated equipment is used outside of the usage conditions and product specifications described in the manuals, this warranty is void.
- 2. If you do not follow the WARNINGS and CAUTIONS in this manual, we cannot be responsible for any malfunction or accident, even if the result is injury or death.
- 3. We cannot foresee all possible dangers and consequences. Therefore, this manual cannot warn the user of all possible hazards.

TRADEMARKS

Microsoft, Windows, and Windows logo are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Allen-Bradley and Studio 5000 are registered trademarks of Rockwell Automation, Inc.. Other brand and product names are trademarks or registered trademarks of the respective holders.

TRADEMARK NOTATION IN THIS MANUAL

Microsoft® Windows® 8 Operating system

Microsoft® Windows® 10 Operating system

Throughout this manual, Windows 8, and Windows 10 refer to above respective operating systems. In some cases, Windows refers generically to Windows 8, and Windows 10.

NOTICE

No part of this manual may be copied or reproduced without authorization.

The contents of this manual are subject to change without notice.

Please notify us if you should find any errors in this manual or if you have any comments regarding its contents.

MANUFACTURER

SEIKO EPSON CORPORATION

CONTACT INFORMATION

Contact information is described in "SUPPLIERS" in the first pages of the following manual:

Robot System Safety and Installation Read this manual first

1

2. Operation	1
2.1 Requirements	1
2.2 Robot Controller Preparation	1
2.3 PLC Project Preparation	1
2.4 SPEL AOI Common Inputs and Outputs	1
2.5 SPEL AOI General Operation	2
2 Configuring the Debat Controller	2
3. Configuring the Robot Controller	2

4. Creating a PLC Project using SPEL AOIs

1. Introduction

4

5. SPEL AOI Reference	14
SPEL_Above	14
SPEL_Accel	15
SPEL_AccelS	16
SPEL_Arc	17
SPEL_Arc3	18
SPEL_ArchGet	19
SPEL_ArchSet	20
SPEL_BaseGet	21
SPEL_BaseSet	22
SPEL_Below	23
SPEL_CPOff	24
SPEL_CPOn	25
SPEL_ExecCmd	26
SPEL_FineGet	27
SPEL_FineSet	28
SPEL_Flip	29
SPEL_Go	30
SPEL_In	31
SPEL_InertiaGet	32
SPEL_InertiaSet	33
SPEL_Init	34
SPEL_InW	36
SPEL_Jog	37
SPEL_Jump	38
SPEL_Jump3	39

SPEL_Jump3CP	40
SPEL_Lefty	41
SPEL_LimZ	42
SPEL_LocalGet	43
SPEL_LocalSet	44
SPEL_MemIn	45
SPEL_MemInW	46
SPEL_MemOff	47
SPEL_MemOn	48
SPEL_MemOut	49
SPEL_MemOutW	50
SPEL_MemSw	51
SPEL_MotorOff	52
SPEL_MotorOn	53
SPEL_Move	54
SPEL_NoFlip	55
SPEL_Off	56
SPEL_On	57
SPEL_Out	58
SPEL_OutW	59
SPEL_PowerHigh	60
SPEL_PowerLow	61
SPEL_ResetError	62
SPEL_Righty	63
SPEL_SavePoints	64
SPEL_Speed	65
SPEL_SpeedS	66
SPEL_Sw	67
SPEL_Teach	68
SPEL_ToolGet	69
SPEL_ToolSet	70
SPEL_WeightGet	71
SPEL_WeightSet	72
SPEL_XYLimGet	73
SPEL_XYLimSet	74

75

1. Introduction

This manual contains instructions on how to use EPSON RC+ 7.0 SPEL AOI (Add-On-Instructions). AOI allows Allen-Bradley® PLC users to execute commands in EPSON robot controllers from a PLC ladder logic program. EPSON AOI uses extended remote I/O to communicate with the controller via EtherNet/IP.

2. Operation

2.1 Requirements

To use SPEL Add-On-Instructions, the following are required:

- An EPSON robot controller with an EtherNet/IP Fieldbus slave board installed. The controller must have firmware version 7.5.0.0 or greater installed.
- A Windows PC with EPSON RC+ 7.0 v7.5.0 or greater installed. This is used to configure the robot controller for AOI operation, and to create a project in the controller when a robot point file is used.

2.2 Robot Controller Preparation

To prepare the robot controller for SPEL AOI execution:

- 1. Install an EtherNet/IP Fieldbus slave board in the controller and connect it to the Ethernet network used for communication with the PLC.
- 2. Configure the robot controller for use with SPEL AOI. See chapter 3. *Configuring the Robot Controller*.

2.3 PLC Project Preparation

To prepare the PLC project for SPEL AOI execution:

- 1. Setup the A1 EtherNet module for communication with the robot controller. You can import the EpsonEtherNetIP.L5X file (recommended), or you can manually set it up. See chapter 5, *Creating a PLC Program*.
- 2. Either import all SPEL AOIs into the project by importing SPEL_All.L5x, or import the desired SPEL AOIs separately. You must always import the SPEL_Init AOI.
- 3. Create a rung for execution of the SPEL_Init AOI. This must be executed once before executing other AOIs. SPEL_Init executes SPEL_ResetError and checks robot controller configuration. There are no errors, then AOI execution is allowed.

2.4 SPEL AOI Common Inputs and Outputs

Each SPEL AOI has the following common inputs and outputs:

Inputs:

Name of AOI	A local tag that references the name of the AOI.
ExtInputs	These are the input IO mapping.
ExtOutputs	These are the output IO mapping.
Start	This is the input that starts the AOI.

Outputs:	
InCycle	BOOL output bit that indicates the status of execution of the AOI. If this is high, then the AOI is executing.
Done	BOOL output bit that indicates the status of completion of the AOI. If this is high, then the AOI execution is complete.
Error	BOOL output bit that indicates if an error occurred during execution.
ErrCode1 and ErrCode2	INT error codes from the robot controller. These should be 0 in normal operation, and one or both are greater than 0 when the Error bit is high.

AOIs have additional inputs and/or outputs. These are described separately for each AOI in the chapter 5.*SPEL AOI Reference*.

2.5 SPEL AOI General Operation

General operation of all SPEL AOIs is as follows:

- 1. SPEL_Init AOI must have been executed one time sucessfully before executing other SPEL AOIs.
- 2. Set the Start input from low to high to start execution.
- 3. During execution, the Done and Error output bits are set to low and the InCycle output bit is set to high.
- 4. After execution, the Done output bit is set to high and the InCycle output bit is set to low. If an error occurred during execution, the Error output bit is set to high, and the error code values ErrCode1 and ErrCode2 are set. See the chapter *6. Error Codes* for more information.
- 5. If an error occurs, SPEL AOI execution is prevented until the SPEL_ResetError AOI is executed.

3. Configuring the Robot Controller

In this chapter we will describe how to configure the robot controller Fieldbus slave to work with the PLC when using SPEL AOIs. Perform the following steps:

- 1. Start EPSON RC+ 7.0 on your PC.
- 2. Connect to the robot controller. You may need to configure a connection to the robot controller in [Setup]-[PC to Controller Communications]. See the EPSON RC+ 7.0 User's Guide for instructions.
- 3. From the [Setup] menu, select [System Configuration].

<u>F</u> ile <u>E</u> dit <u>V</u> iew <u>P</u> roject <u>R</u> un <u>T</u> ools	Set <u>up</u> <u>W</u> indow <u>H</u> elp
🗅 🚅 🖉 🎒 👗 🖻 🛍 🔛 🖗	▶ PC to Controller <u>C</u> ommunications
Project Explorer	📖 System Configuration

4. Click [Controller]-[Inputs/Outputs]-[Fieldbus Slave]. Configure the number of input and output bytes to 128 as shown below, then click <Apply>.

📖 System Configuration		?	Х
-Startup -Controller -General	Fieldbus I/O Slave	Close	
ConfigurationPreferencesSimulatorSimulatorDrive Units	Fieldbus Type: EtherNet/IP Input Bytes: 128 v Output Bytes: 128 v	<u>A</u> pply <u>R</u> estore	

- 5. Expand [Fieldbus Slave] in the tree and select [Ethernet/IP].
 - Set the IP address, mask, and gateway that will be used for communication from the A1 EtherNet module in the PLC.

🛤 System Configuration			? 💌
	EtherNet/IP		
- Controller	MAC Address:	00-30-11-29-1C-85	Close
General Configuration	<u>H</u> ost Name:	ETHIP0000	
Preferences Simulator	Domain <u>N</u> ame:	EpsonRobots	Restore
	Primary DNS:	192.168.10.1	<u></u>
iani Inputs / Outputs iani Fieldbus Slave	Secondary DNS:	0.0.0.0	
General EtherNet/IP	Timeout	75 seconds	
Analog I/0 ■- Remote Control ■- RS232	Address <u>C</u> onfiguration:	Static ○ DHCP/BOOTP/ARP Static ○ DHCP/BOOTP/A	
TCP / IP Conveyor Encoders	IP Address:	192.168.10.162	
- Security	IP <u>M</u> ask:	255.255.255.0	
i - Vision	IP <u>G</u> ateway:	192.168.10.1	

6. Select [Remote Control]-[PLC] and select Allen-Bradley® as the PLC Vendor.

📖 System Configuration			? 💌
General Configuration Preferences Simulator Drive Units Robots Inputs / Outputs General EtherNet/IP Analog I/0 Remote Control Inputs Outputs User Outputs Ethernet RS232 RS232 CDP / IP Conveyor Encoders	PLC	PLC Vendor: Allen-Bradley	Close Apply Restore

7. Click "Close" on the [System Configuration] dialog. The controller will restart.

4. Creating a PLC Project using SPEL AOIs

EPSON RC+ 7.0 users are provided with Allen-Bradley® Logix Designer files which are installed on the user PC by the EPSON RC+ v7.5.0 or greater installer. The files are located in \EpsonRC70\Fieldbus\EtherNetIP\AOI on the user PC.

In this chapter, we will show how to create a simple example project to turn robot motors on and off using SPEL AOIs.

To create a new project, make sure you are in offline mode and follow these steps:

1. Start the Studio 5000® software, then click [New Project]. The New Project dialog will be displayed.



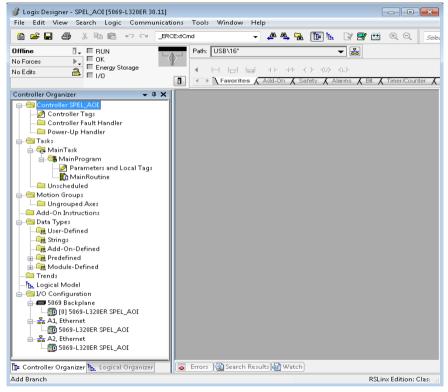
2. Choose your Controller family and PLC controller model number. Enter a project name under [Name], then click <Next>.

🕝 New Project		? 🔀
Project Types	Search	×
💰 Logix	Compact GuardLogix® 5370 Safety Controller	^
🕥 View	CompactLogix [™] 5370 Controller CompactLogix [™] 5380 Controller 5069-L306ER CompactLogix [™] 5380 Controller 5069-L3100ERM CompactLogix [™] 5380 Controller 5069-L310ER CompactLogix [™] 5380 Controller	
	S009-L310ER CompactLogix [™] 5380 Controller 5069-L310ER-NSE CompactLogix [™] 5380 Controller 5069-L320ER CompactLogix [™] 5380 Controller 5069-L320ER CompactLogix [™] 5380 Controller	÷
	Name: SPEL_AO	Browse
	Cancel Back Next	Finish

3. The dialog shown below will be displayed. Leave all choices as default, then click <Finish>.

🕝 New Project		? 💌
5069-L320ER Co SPEL_AOI2	mpactLogix™ 5380 Controller	
Revision:	30 •	
Security Authority:	No Protection 👻	
	$\hfill \hfill $	
Secure With:	○ Logical Name <controller name=""></controller>	
	O Permission Set	
Description:		
	Cancel Back Next	Finish

4. You have just created a new empty PLC project



5. Now you need to add and configure the Ethernet module for communications with the robot controller. There are two methods: Import the file EpsonEtherNetIP.L5X, or perform manual configuration.

Importing the Ethernet configuration

- 1. Right click on [A1, Ethernet], then click [Import Module].
- 2. Navigate to \EpsonRC70\Fieldbus\EtherNetIP\AOI and select the file EpsonEtherNetIP.L5X.

Import Module					Ð
🖉 🗸 🖉 🖉 Windows	7_OS (C:) EpsonRC70 Fieldbus	EtherNetIP + AOI -	✓ Search ≠	.0/	
Organize 👻 New fold	er			≡ • 🔳	2
🔆 Favorites 🕺	Name	Date modified	Туре	Size	
🧮 Desktop	EpsonEtherNetIP.L5X	7/23/2020 4:33 PM	L5X File	23 KB	
📃 Recent Places	SPEL_Above.L5X	7/23/2020 4:25 PM	L5X File	28 KB	
	SPEL_Accel.L5X	7/23/2020 4:25 PM	L5X File	29 KB	
🥽 Libraries	SPEL_AccelS.L5X	7/23/2020 4:25 PM	L5X File	31 KB	
🖻 Documents 🗧	SPEL_AII.L5X	7/23/2020 4:25 PM	L5X File	565 KB	
🌛 Music	SPEL_Arc.L5X	7/23/2020 4:25 PM	L5× File	29 KB	
📔 Pictures	SPEL_Arc3.L5X	7/23/2020 4:25 PM	L5X File	29 KB	
🛃 Videos	SPEL_ArchGet.L5X	7/23/2020 4:25 PM	L5X File	31 KB	
	SPEL_ArchSet.L5X	7/23/2020 4:25 PM	L5X File	31 KB	
🖳 Computer	SPEL_BaseGet.L5X	7/23/2020 4:25 PM	L5X File	50 KB	
🏭 Windows7_OS (C	SPEL_BaseSet.L5X	7/23/2020 4:25 PM	L5X File	49 KB	
🔮 DVD RW Drive (E:	SPEL_Below.L5X	7/26/2020 8:11 AM	L5X File	28 KB	
👝 Lenovo_Recover, 🔻	SPEL_CPOff.L5X	7/23/2020 4:25 PM	L5X File	28 KB	
File n	ame: EpsonEtherNetIP.L5X		✓ Logix Des	igner XML Files (*.L5X	•
			Open	Cancel	_

3. After import, right-click on the module and select Properties. Change the default IP address to be the address of the robot controller's EtherNetIP slave board.

Type: Vendor:	ETHERNET-MODULE Generic Allen-Bradley	Ethern	et Module			
Parent:	Local		Connection Para	ameters		
Name: Description:	Epson		Connection r die	Assembly Instance:	Size:	
		Î	Input:	100	64 🊔 (16-bit)
		Ŧ	Output:	150	64 🊔 (16-bit)
Comm Forma		-	Configuration:	1	0 🚔 (8-bit)
Address / H		5	Status Input:			
🔘 Host Na	ime:		Status Output:			

Manual Ethernet configuration

1. Right click on [A1, Ethernet], then click [New Module].

	ne	ER SPEL_AOI	
5069-L32	1	New Module	
🛓 🚠 🗛 A2, Ethernet		Import Module	
🔁 5069-L32		Discover Modules	
📴 Controller Organizer		Paste	Ctrl+V
Create a module		Properties	Alt+Enter
		Print	•

2. Type in "generic" in the search field. Click "ETHERNET MODULE" under catalog number, then click <Create>.

Select Module Type				
Catalog Module Discovery Favorites	<u>Clear</u> Filters			Show Filters 🗧
Catalog Number	Description	Vendor	Category	
ETHERNET-BRIDGE	Generic EtherNet/IP CIP Bridge	Allen-Bradley	Communication	
ETHERNET-MODULE	Generic Ethernet Module	Allen-Bradley	Communication	
ETHERNET-SAFETY-STANDA	Generic EtherNet/IP Safety and Standard Module	Allen-Bradley	Safety,Other	
3 of 474 Module Types Found				Add to Favorites
Close on Create			Create	Close Help

3. Enter the values as shown, and use the IP address of the robot controller EtherNet/IP slave, then click <OK>.

New Module					.
Type: Vendor: Parent:	ETHERNET-MODULE Generic Ethernel Allen-Bradley Local				
Na <u>m</u> e: Descri <u>p</u> tion:	Epson	Connection Par.	Assembly Instance: 100	Size: 64	🚔 (16-bit)
		O <u>u</u> tput:	150	64	膏 (16-bit)
Comm <u>F</u> ormat Address / H		<u>C</u> onfiguration:	1	0	🚔 (8-bit)
IP Addre	ess: 192 . 168 . 10 . 162	<u>S</u> tatus Input:			-
© <u>H</u> ost Na	me:	Status Output:			
🔽 Open Modu	uļe Properties	OK	Can	cel	Help

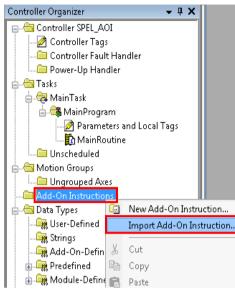
4. Click <OK> on the next window.

Module Properties Report: Local (ETHERNET-MODULE 1.001)
General Connection Module Info
Requested Packet Interval (RPI): 10.0 ms (1.0 - 3200.0 ms)
Major Fault On Controller If Connection Fails While in Run Mode
☑ Use Unicast Connection over EtherNet/IP
Module Fault
Status: Offline OK Cancel Apply Help

Saving your project at this stage is a good idea. When creating a new Ethernet module, please note that connection parameter values should match your robot controller values.

Import SPEL AOIs in to the new project

 Now you need to import AOIs in the new project. For this example, you will import all SPEL AOIs. You can also import individual AOIs. To do this, right click on [Add-On Instructions] folder from [Controller Organizer], click [Import Add-On Instruction].



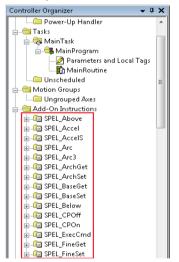
2. Navigate to \EpsonRC70\Fieldbus\EtherNetIP\AOI, then select "SPEL_All.L5X" file and click <Open>.

Import Add-On Instruction	n 7_OS(C:) → EpsonRC70 → Fieldbus → Ether	NetIP + AOI	Search At	21	×
Organize New folde		incur , nor		8≡ ▼ 🔟	0
★ Favorites	Name	Date modified	Туре	Size	
🧮 Desktop	EpsonEtherNetIP.L5X	7/23/2020 4:33 PM	L5X File	23 KB	
📃 Recent Places	SPEL_Above.L5X	7/23/2020 4:25 PM	L5X File	28 KB	
	SPEL_Accel.L5X	7/23/2020 4:25 PM	L5X File	29 KB	
🥽 Libraries	SPEL_AccelS.L5X	7/23/2020 4:25 PM	L5X File	31 KB	
🖹 Documents 🗮	SPEL_AII.L5X	7/23/2020 4:25 PM	L5X File	565 KB	
🌙 Music	SPEL_Arc.L5X	7/23/2020 4:25 PM	L5X File	29 KB	
Pictures	SPEL_Arc3.L5X	7/23/2020 4:25 PM	L5X File	29 KB	
🛃 Videos	SPEL_ArchGet.L5X	7/23/2020 4:25 PM	L5X File	31 KB	
	SPEL_ArchSet.L5X	7/23/2020 4:25 PM	L5X File	31 KB	
📜 Computer	SPEL_BaseGet.L5X	7/23/2020 4:25 PM	L5X File	50 KB	
🏭 Windows7_OS (C	SPEL_BaseSet.L5X	7/23/2020 4:25 PM	L5X File	49 KB	
🍰 DVD RW Drive (E:	SPEL_Below.L5X	7/26/2020 8:11 AM	L5X File	28 KB	
👝 Lenovo_Recovery 🔻	SPEL_CPOff.L5X	7/23/2020 4:25 PM	L5X File	28 KB	
File na	ame: SPEL_AII.L5X		👻 🛛 Logix Desig	gner XML Files (*.L5X	•
			Open	- Cancel	

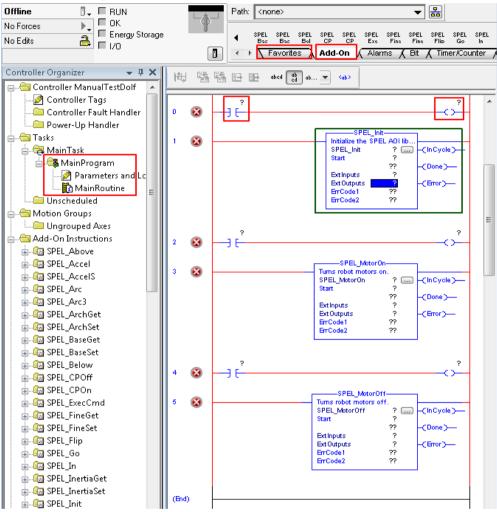
3. The dialog below is displayed. Check to make sure that there are no errors, then click <OK>.

Import Configuration - SPEL_All.L5X		
문 또 Find: Find Within: Final Name	・ 唐 氏 [EndReplace	
Import Content:		
- Add-On Instructions - Add-On Instructions - References - Add-On Instructions - Add-On Instructions - Content of the add of th	Configure Add-On Instructions Imported Instructions: 60 selected, 0 others as references I) Instructions and other references will be imported as configured in the References folders	
leady	Cancel H	ielp

4. Now you should see the list of all SPEL AOIs in the project.



- 5. Now you can create a program.
 - 5-1. Expand [MainProgram], then double click on [MainRoutine].
 - 5-2. Click [Favorites] tab, add 5 extra rungs. Then drag "Examine On" and "Output Energize" to rung 0, 2 and 4.
 - 5-3. Click [Add-On] tab, drag "SPEL_Init" to rung 1, "SPEL_MotorOn" to rung 3, and "SPEL_MotorOff" to rung 5, like shown below.



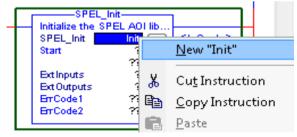
6. In rung 0, double click at [?] of "Examine On", type in the name of the variable. In this case we will use "InitSwitch".

New Paramet	er or Tag	×
Name:	InitSwitch	Create 🗸 🔻
Description:		Cancel
		Help
	-	
Usage:	Local Tag 🔹	
Туре:	Base Connection	
Alias For:		
Data Type:	BOOL	
Parameter Connection:		
Scope:	🕞 MainProgram 👻	
External Access:	Read/Write	
Style:	Decimal 🔹	

- 7. Do the same step as above, in rung 0, double click on [?] of the "Output Energize", and type "InitCoil".
- 8. Right click on [InitSwitch], click on [New "InitSwitch"], then click <Create>, as shown below.



- 9. Create new variable "InitCoil" same method used in "InitSwitch".
- 10. Do same steps in 14 to 17 for rung 2 and 4 to create new variables. Use variable name "MotorOnSwitch", "MotorOnCoil" for rung 2, and "MotorOffSwitch", "MotorOffCoil" for rung 4.
- 11. Now we configure SPEL_Init AOI inputs.
 - 11-1. Inside "SPEL Init" block, click [?] to the right of [SPE Init], and type "Init".
 - 11-2. Right click on [Init], choose [New "Init"]. then click <Create>.



"Init" will be the name of the structure that holds all internal variable of "SPEL_Init" AOI.

11-3. Click [?] next to "Start", type "InitCoil", you do not need to create a new variable.

11-4. Click [?] next to [ExtInputs], type "Ep", it will auto populate, press <Enter>.

Initialize the SPEL AOI library.					
SPEL_Init					
SPEL_Init Init(InCycle) Start InitCoil					
0 ← _(Done)					
Ext Outputs (Error)					
EπCode1 0 ← EπCode2 0 ←					

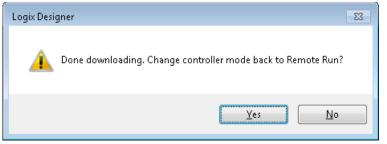
- 11-5. Do same step to [ExtOutputs]. "SPEL_Init" is now configured and the rung lines should change from red to blue.
- 11-6. Do the same steps as in 11-1 to 11-2 for rung 3 and 5. Choose "MotorOn" for rung 3, "MotorOff" for rung 5.
- 11-7. Do the same steps as in 11-3 for rung 3 and 5. Use "MotorOnCoil" for rung 3, "MotorOffCoil" for rung 5.
- 12. The program in now complete. Save the project.
- 13. Click the down arrow right to [Path] to choose communication path with controller.

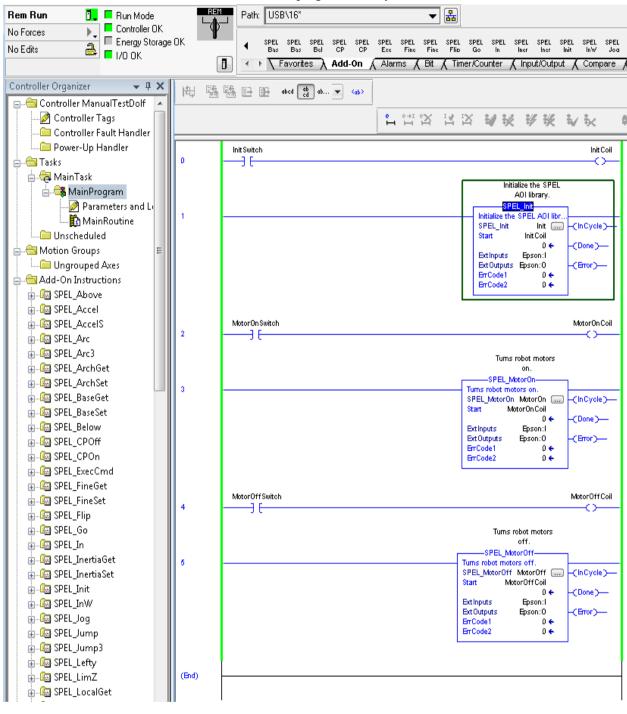
Path: <none></none>		note Console]	
SPEL SPE Bas Bas Bas	unications Path		×
Controller Path			Go Online
JimTest USB\16			Upload
Set Init			Download

- In this example I am using USB to connect my PC to the PLC controller.
- 14. Double click on "USB" to close the window, then click <Download> in the next window to transfer program to PLC controller.

Connected To Go Online	
Options General Date/Time	B Major Faults Minor Faults Project Nonvolatile Memory
Condition: The open project	doesn't match the project in the controller.
Connected Controller: Controller Name:	P. 7
Controller Type: Comm Path: Serial Number:	5069L320ER CompactLogix ⁷¹ 5380 Controller USB\16 60C5C5F7
Security: Offline Project:	No Protection
Controller Name: Controller Type: File: Serial Number: Security:	ManualTestDolf 5069-L32DER CompactLogix ⁷⁷ 5380 Controller SERVDocuments\Studio 5000\Projects\ManualTestDolf.ACD <none> No Protection</none>
	Download Select File

15. Click <Yes> in the next window if prompted to change PLC into "Remote Run" mode, like shown below.





16. PLC now in run mode and program is ready to be executed.

5. SPELAOI Reference

In this chapter each SPEL AOI is described.

For AOI operation in general, refer to section 2.5, SPEL AOI General Operation.

For each AOI in the Operation section, there is also a referal to the corresponding SPEL+ command in the SPEL+ Language Reference manual which has more details about the command.

Each AOI has a simple example.

SPEL_Above

Description

Sets the elbow orientation of the specified point to Above.

Common inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Point INT point number to set its orientation to ABOVE.

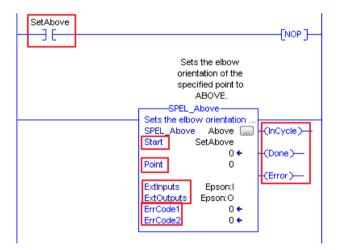
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *Elbow Statement* in the SPEL+ Language Reference manual.

Example

To set P0 orientation to Above, set [Point] to "0", as shown below.



SPEL Accel

Description

Sets the point to point acceleration and deceleration. Specifies the ratio (%) of the maximum acceleration/deceleration using an integer equals to or greater than 1.

Common inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Accel INT value of acceleration as percentage. *Decel* INT value of deceleration as percentage.

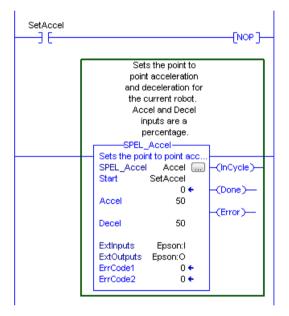
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Accel Statement in the SPEL+ Language Reference manual.

Example

To set acceleration to 50% and deceleration to 50%, set [Accel] to "50" and [Decel] to "50", as shown below.



SPEL_AccelS

Description

Sets acceleration and deceleration. Specifies the value which is the actual acceleration/deceleration in linear or CP motion (Unit: mm/sec2).

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Accel REAL value of acceleration. *Decel* REAL value of deceleration.

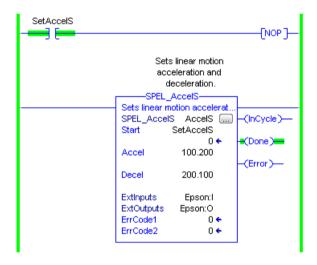
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to AccelS Statement in the SPEL+ Language Reference manual.

Example

To set acceleration to 100.200, deceleration to 200.100, set [Accel] to "100.200", [Decel] to "200.100", as shown below.



SPEL_Arc

Description

Moves the arm from the current position to the specified position in circular interpolation motion on XY plane face.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

midPoint	INT middle point in Arc command.
endPoint	INT end point in Arc command.

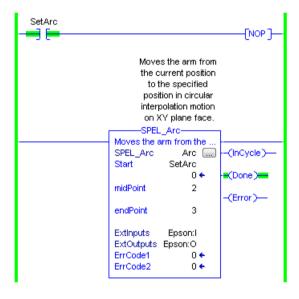
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Arc Statement in the SPEL+ Language Reference manual.

Example

To move from current position passing through P2 and ending at P3, in a circular motion.



5. SPEL AOI Reference

SPEL_Arc3

Description

Moves the arm from the current position to the specified position in circular interpolation in 3 dimensions.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

midPoint INT middle point in Arc3 command. *endPoint* INT end point in Arc3 command.

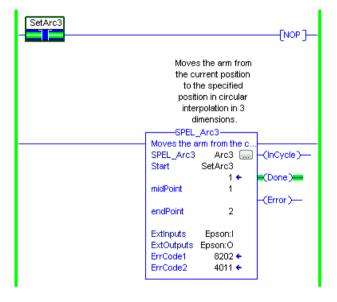
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Arc3 Statement in the SPEL+ Language Reference manual.

Example

To move from current position passing through P1 and ending at P2, in a circular motion.



SPEL_ArchGet

Description

Gets the Arch parameter.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

ArchNum INT desired Arch number.

Outputs

DepartDist	INT departing distance of the given Arch number.
ApproachDist	INT approaching distance of the given Arch number.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Arch Function in the SPEL+ Language Reference manual.

Example

To get the current values of approach and depart distances of given Arch, set the Arch number.

SetArchGet			[NOP]
	SPEL_Arch	meter Get	
	Gets the Arch para SPEL_ArchGet A ArchNum		-(InCycle)
	ExtOutputs E	Epson:I pson:O .rchGet	-(Error)
	DepartDist ApproachDist	1 ← 40.0 ← 40.0 ←	
	ErrCode1 ErrCode2	0 € 0 €	

SPEL_ArchSet

Description

Sets the Arch parameter.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

ArchNumINT desired Arch number.DepartDistREAL departing distance of the given Arch number.ApproachDistREAL approaching distance of the given Arch number.

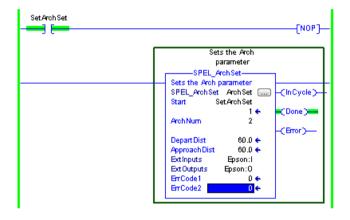
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Arch Statement in the SPEL+ Language Reference manual.

Example

To set 60.0, 60.0 as depart and approach distances respectively of Arch 2, see below.



SPEL BaseGet

Description

Gets the base coordinate system.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

NumAxes INT number of robot axes. For a SCARA robot, use 4. For a 6-axis robot, use 6.

Outputs

BaseX	REAL base value of coordinate X.
BaseY	REAL base value of coordinate Y.
BaseZ	REAL base value of coordinate Z.
BaseU	REAL base value of coordinate U.
BaseV	REAL base value of coordinate V.
BaseW	REAL base value of coordinate W.

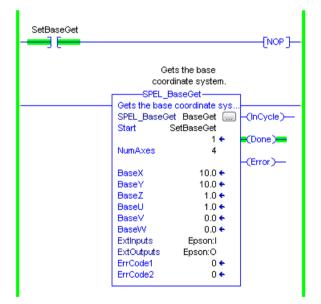
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *Base Statement* in the SPEL+ Language Reference manual.

Example

To get the base values of X through W coordinates for SCARA robot, plug 4 for NumAxes. Base values will update as shown below.



SPEL_BaseSet

Description

Sets the base coordinate system.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

NumAxes	INT number of robot axes. For a SCARA robot, use 4. For a 6-axis robot, use 6.
BaseX	REAL base value of coordinate X.
BaseY	REAL base value of coordinate Y.
BaseZ	REAL base value of coordinate Z.
BaseU	REAL base value of coordinate U.
BaseV	REAL base value of coordinate V.
BaseW	REAL base value of coordinate W.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *Base Statement* in the SPEL+ Language Reference manual.

Example

To set the base value of a SCARA robot, set NumAxes = 4. Enter the base coordinate value for each axis, as shown below.

SetBaseSet			[NOP]
		Sets the base ordinate system.	
		_BaseSet	1
		se coordinate sy Set BaseSet SetBaseSet	-(InCycle)
		1 🗲	(Done)
	NumAxes	4	-(Error)
	BaseX	40	
	BaseY	40	
	BaseZ	4	
	BaseU	4	
	Base∀	0	
	BaseW	0	
	ExtInputs ExtOutputs	Epson:I Epson:O	
	ErrCode1	0 🗲	
	ErrCode2	0 ←	

SPEL Below

Description

Sets the elbow orientation of the specified point to Below.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Point INT desired point number.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *Elbow Statement* in the SPEL+ Language Reference manual.

Example

.

To set orientation of P2 to below, enter 2 as point. As shown below.

SetBelow	[NOP]
50	Sets the elbow orientation of the specified point to Below.
	SPEL_Below
	Sets the elbow orientation SPEL_Below Below(InCycle)
	Point 2
	ExtInputs Epson:I
	ExtOutputs Epson:O ErrCode1 0 +
	ErrCode2 0 +

SPEL_CPOff

Description

Turns off Continuous Path parameter.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

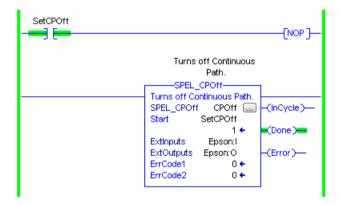
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to CP Statement in the SPEL+ Language Reference manual.

Example

To set CP to off, run the AOI like as shown below.



SPEL_CPOn

Description

Turns on Continuous Path parameter.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

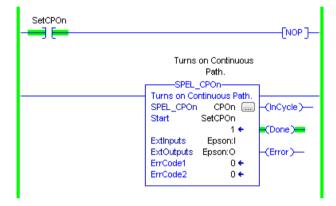
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to CP Statement in the SPEL+ Language Reference manual.

Example

To set CP to On, run the AOI as shown below.



SPEL ExecCmd

Description

The SPEL_ExecCmd AOI is used by other AOIs to execute a command in the robot controller.

SPEL FineGet

Description

Gets the setting of positioning end judgement range for all joints.

Outputs

Axis INT position accuracy for each joint in encoder pulses.

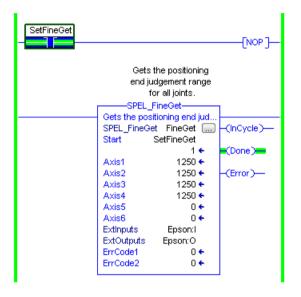
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Fine Function in the SPEL+ Language Reference manual.

Example

To get the position accuracy for the robot, run the AOI as shown below.



5. SPEL AOI Reference

SPEL_FineSet

Description

Sets the positioning end judgement range for all joints.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Axis1..Axis6 INT position accuracy for each joint in encoder pulses.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Fine Statement in the SPEL+ Language Reference manual.

Example

To set the position accuracy for the robot, enter the Axis values and run the AOI as shown below.

SetFineSet			Evon 1
			[NOP]
	end ju f SPEL_	the positioning idgement range or all joints. FineSet	1
	SPEL_FineS(itioning end jud et FineSet	-(InCycle)
	Start	SetFineSet	(110)010)
		1 🗲	(Done)
	Axis1	1300	
	Axis2	1400	-(Error)
	Axis3	1500	
	Axis4	1600	
	Axis5	0	
	Axis6	0	
	ExtInputs	Epson:I	
	ExtOutputs	Epson:O	
	ErrCode1	0 🗲	
	ErrCode2	0 🗲	
	L		1

SPEL_Flip

Description

Sets the wrist orientation of the specified point to Flip.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Point INT desired point number.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Wrist Statement in the SPEL+ Language Reference manual.

Example

To set orientation of robot point P2 to flip, enter 2 as the point number and run the AOI as shown below.

SetFli			
	oriei spec	ts the wrist ntation of the cified point to Flip. _ Flip.	
		ist orientation	
•	SPEL_Flip	Flip	-(InCycle)
	Start	SetFlip	
		1 🗲	(Done)
	Point	2	-(Error)
	ExtInputs	Epson:1	
	ExtOutputs	Epson:O	
• •	ErrCode1	0 🔶	
	ErrCode2	0 🗲	
L _			,

5. SPEL AOI Reference

SPEL_Go

Description

Moves from the current position to the specified position in PTP motion.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Point INT desired point number.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Go Statement in the SPEL+ Language Reference manual.

Example

To move the robot to point 0 using PTP motion, enter "0" as the point and run the AOI, as shown below.

SetGoP0		[NOP]
	Moves from the current position to the specified position in PTP motion.	
	SPEL_Go	ר ר
	Moves from the current p.	
	SPEL_Go GoPO 🛄	-(InCycle)
	Start SetGoP0	
	1 🗲	(Done)
	Point 0	
		-(Error)
	Extinputs Epson: I	
	ExtOutputs Epson:O	
	ErrCode1 0 +	
	ErrCode2 0 +	

SPEL_In

Description

Reads a byte of input.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

PortNum INT desired input byte port number.

Outputs

Value INT value of the desired input port.

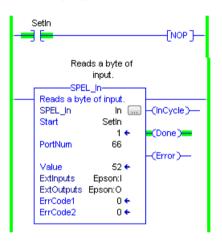
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *In Function* in the SPEL+ Language Reference manual.

Example

To read input port number 66, set [PortNum] to "66"-



SPEL_InertiaGet

Description

Gets the load inertia.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Outputs

Inertia REAL acquired Inertia. *Eccentricity* REAL acquired Eccentricity.

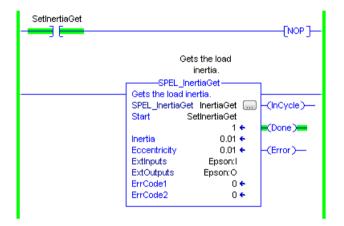
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Inertial Function in the SPEL+ Language Reference manual.

Example

To read load Inertia and Eccentricity, run the AOI, as shown below.



SPEL InertiaSet

Description

Sets the load inertia.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

InertiaREAL desired Inertia.EccentricityREAL desired Eccentricity.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Inertia Statement in the SPEL+ Language Reference manual.

Example

To set load Inertia and Eccentricity to 0.01, 0.01 respectively, enter the values and run the AOI.

SetInertiaSet			[NOP]
	S	ets the load inertia.	
	SPEL_I	nertiaSet	
	SPEL_InertiaS		-(InCycle)
	Inertia	1 + 0.01	(Done)
	Eccentricity	0.01	-(Error)
	ExtInputs	Epson:I	
	ExtOutputs	Epson:O	
	ErrCode1 ErrCode2	0€ 0€	

SPEL_Init

Description

Initializes the PLC program for SPEL AOI execution. It is required to execute SPEL_Init before executing any other AOIs.

Note: If the controller has a system error, then it must be reset before SPEL_Init and other SPEL AOIs can execute successfully.

Common Inputs and Outputs

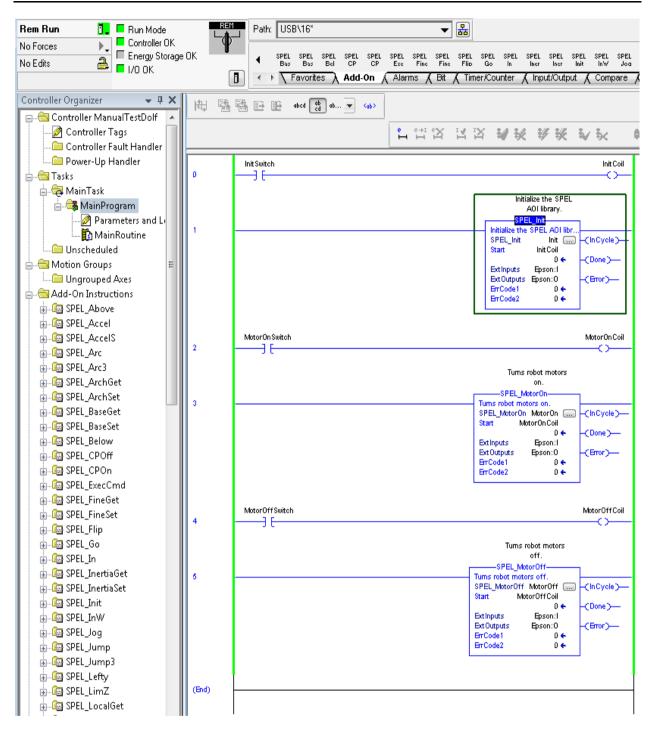
Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Example

As shown below, toggle [Init Switch] to high to start the AOI.



5. SPEL AOI Reference

SPEL_InW

Description

Returns the status if an input word.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

PortNum INT desired port number.

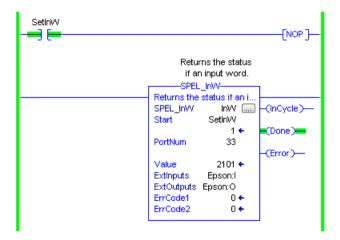
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *InW Function* in the SPEL+ Language Reference manual.

Example

To read content of port number 33, enter the value and run the AOI.



SPEL_Jog

Description

Jogs the robot.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

JogModeINT desired mode. 0=World, 1=Joint.AxisINT desired axis.DistanceREAL value:
World:
X,Y,Z in mm.
U,V,W in deg.
Joint: J1-J6 in deg.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Example

To move robot in -Y direction for 40mm, enter values and run the AOI as shown below.

SetJog	[NOP]
	Linot 1
Jogs the	robot.
SPEL_Jog-	
Jogs the robot. SPEL_Jog	Jog(InCycle)
SPEL_Jug Start Set	
Start Star	1 ← = (Done) =
JoqMode	0
	(Error)
Axes	2
Distance	-40
Extinputs Eps	op:l
Extoutputs Epso	
ErrCode1	0 ←
ErrCode2	0 🗧

5. SPEL AOI Reference

SPEL_Jump

Description

Moves the arm using gate motion for a SCARA robot.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Point INT desired point.

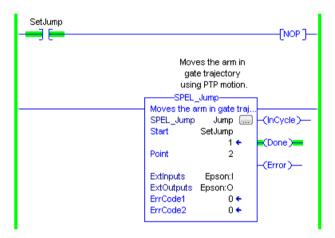
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Jump Statement in the SPEL+ Language Reference manual.

Example

To move the robot to point P2 using gate trajectory, enter the value for Point and run the AOI as shown below.



.

SPEL Jump3

Description

Moves the arm with 3D gate motion for a 6-axis robot. This is a combination of two CP motion and one PTP motion.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

DepartPoint INT desired depart point.

ApproPoint INT desired approach point.

DestPoint INT desired destination point.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Jump3CP Statement in the SPEL+ Language Reference manual.

Example

.

To move the robot to point P2 using gate trajectory, enter the values for the points and run the AOI as shown below.

StartJump3S	StartJump3C
	Moves the arm with 3D gate motion This is a combination of two CP motion and one PTP motion. SPEL_Jump3-
	Moves the arm with 3D gat SPEL_Jump3 Jump3 [] -(InCycle) Start StartJump3C
	1 ← Cone) DepartPoint 1 -(Error)
	ApproPoint 2
	DestPoint 3
	ExtInputs Epson:I
	ExtOutputs Epson:O
	ErrCode1 0 +
	ErrCode2 0 🗧

SPEL_Jump3CP

Description

Moves the arm with 3D gate motion for a 6-axis robot. This is a combination of three CP motions.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

DepartPoint INT desired depart point.

ApproPoint INT desired approach point.

DestPoint INT desired destination point.

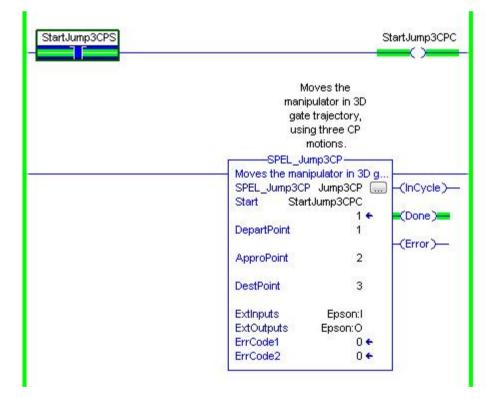
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Jump3CP Statement in the SPEL+ Language Reference manual.

Example

To move the robot to point P2 using gate trajectory, enter the values for the points and run the AOI as shown below.



SPEL_Lefty

Description

Sets the hand orientation of the specified point to Lefty.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Point INT desired point number.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *Hand Statement* in the SPEL+ Language Reference manual.

Example

To change P2's hand orientation to Lefty, enter values and run the AOI as shown below.

SetLefty	[NOP]
	Sets the hand orientation of the specified point to Lefty. SPEL_Lefty
	Sets the hand orientation SPEL_Lefty Lefty(InCycle)- Start SetLefty
	1 ← ■(Done) Point 2
	ErrCode2 0 +

5. SPEL AOI Reference

SPEL_LimZ

Description

Sets the initial Joint #3 height (Z coordinate value) in Jump command.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Height REAL desired Z limit in mm.

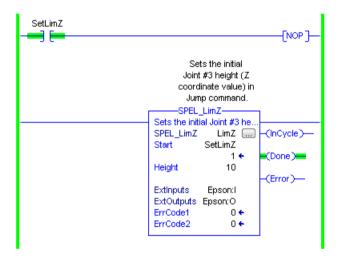
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *LimZ Statement* in the SPEL+ Language Reference manual.

Example

To set LimZ value of 10mm, enter values and run the AOI as shown below.



SPEL LocalGet

Description

Gets data for a given local coordinate system.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

NumAxes	INT number of axes in the robot. For SCARA, use 4, for Articulate robot, use 6.
LocalNum	INT desired local number you want to get.

Outputs

LocalX	REAL the coordinate value of that axis.
LocalY	REAL the coordinate value of that axis.
LocalZ	REAL the coordinate value of that axis.
LocalU	REAL the coordinate value of that axis.
LocalV	REAL the coordinate value of that axis.
LocalW	REAL the coordinate value of that axis.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Local Statement in the SPEL+ Language Reference manual.

Example

To get the coordinate values for local number 3 of a SCARA robot, enter values and run the AOI as shown below.

		[NOP]
	ts the setting	
stat	us of the Local	
C00	rdinate system	
	number.	
SPEL_	LocalGet	
	ng status of the	
	et LocalGet 🛄	-(InCycle)
Start	SetLocalGet	
	1 🗲	(Done)
NumAxes	4	
		-(Error)
LocalNum	3	
LocalX	10.0 🗲	
LocalY	20.0 €	
LocalZ	30.0 🗲	
LocalU	40.0 🗲	
LocalV	0.0	
LocalVV	0.0 €	
ExtInputs	Epson:1	
ExtOutputs	Epson:O	
ErrCode1	0 🗲	
ErrCode2	0 🗲	
		1

SPEL_LocalSet

Description

Sets the local coordinate number.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

NumAxes	INT number of axes in the robot. For SCARA, use 4, for Articulate robot, use 6.
LocalNum	INT desired local number you want to get.
LocalX	REAL the desired coordinate value of X axis.
LocalY	REAL the desired coordinate value of Y axis.
LocalZ	REAL the desired coordinate value of Z axis.
Local U	REAL the desired coordinate value of U axis.
LocalV	REAL the desired coordinate value of V axis.
LocalW	REAL the desired coordinate value of W axis.

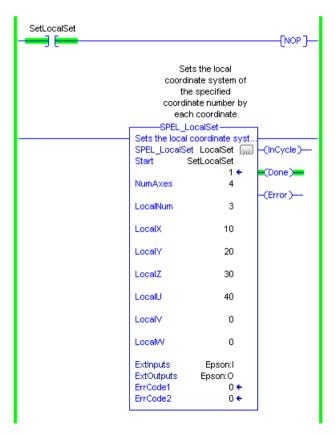
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *Local Statement* in the SPEL+ Language Reference manual.

Example

To set the coordinate values for local number 3 of a SCARA robot, enter values and run the AOI as shown below.



SPEL_MemIn

Description

Reads a byte of memory IO.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

PortNum INT port number to be read. Port number refers to byte number.

Outputs

Value INT value of the port.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to MemIn Function in the SPEL+ Language Reference manual.

Example

To read port number 0 of memory I/O, run the AOI as shown below.

SetMemIn	[NOP]
	Reads a byte of memory IO.
	Reads a byte of memory IO. SPEL_MemIn MemIn
	1 ← ■(Done)== PortNum 0
	-(Error)
	ExtOutputs Epson:O ErrCode1 0 €
	ErrCode2 0 +

SPEL_MemInW

Description

Reads a word of memory IO.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

PortNum INT port number to be read.

Outputs

Value INT value of the port.

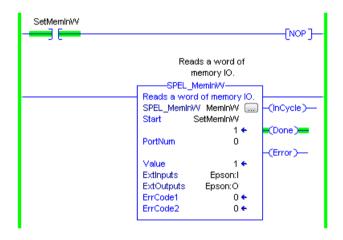
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *MemInW Function* in the SPEL+ Language Reference manual.

Example

To read port number 0 as word, run the AOI as shown below.



SPEL_MemOff

Description

Turns a memory IO bit off.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Bit INT bit number to be turned off.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *MemOff Statement* in the SPEL+ Language Reference manual.

Example

To turn off memory bit number 3, run the AOI as shown below.

SetMemOff	[NOP]
50	с э
	Turns a memory IO
	bit off.
	SPEL_MemOff
	Turns a memory IO bit off.
	SPEL_MemOff MemOff(InCycle)
	Start SetMemOff
	1 ← = (Done) ==
	Bit 3
	-(Error)
	Extinputs Epson:I
	ExtOutputs Epson:O
	ErrCode1 0 +
	ErrCode2 0 🕈

SPEL_MemOn

Description

Turns a memory IO bit on.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Bit INT bit number to be turned on.

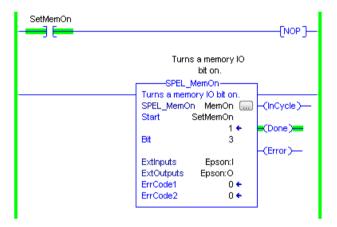
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *MemOn Statement* in the SPEL+ Language Reference manual.

Example

To turn on memory bit number 3, run the AOI as shown below.



SPEL_MemOut

Description

Sets a byte of memory IO.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

PortNum	INT desired output port number.
OutData	INT value of the data to be sent to output port.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *MemOut Statement* in the SPEL+ Language Reference manual.

Example

To send 99 to port number 4, run the AOI as shown below.

SetMemOut	[иор]
	Sets a byte of memory IO. ————————————————————————————————————
	1 ← (Done) PortNum 4 -(Error)
	outData 99 ExtInputs Epson:1 ExtOutputs Epson:0 ErrCode1 0 ← ErrCode2 0 ←

SPEL_MemOutW

Description

Sets a word of memory IO.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

PortNumINT desired output port number.OutDataINT value of the data need to be sent to output port.

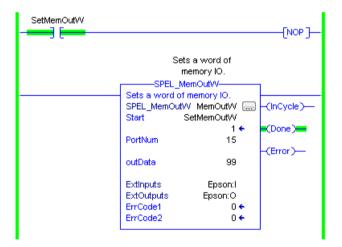
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to MemOutW Statement in the SPEL+ Language Reference manual.

Example

To send 99 to port number 15, run the AOI as shown below.



SPEL_MemSw

Description

Reads a single bit of memory IO.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Bit INT desired memory bit number.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *MemSw Function* in the SPEL+ Language Reference manual.

Example

To read memory bit number 5, run the AOI as shown below.

SetMemSw			
			0.001
	Reads	bit of memory IO.	
		/lemSvv	1
	Reads bit of r	nemory IO.	
	SPEL_MemSV	v MemSw 🛄	-(InCycle)
	Start	SetMemSvv	
		1 🗲	(Done)
	Bit	5	
			-(Error)
	Value	0 🗲	
	ExtInputs	Epson:I	
	ExtOutputs	Epson:O	
	ErrCode1	0 🗲	
	ErrCode2	0 🗲	
		0 ←	

SPEL_MotorOff

Description

Turns robot motors off.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

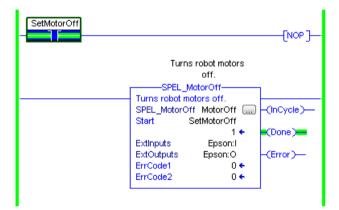
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *Motor Statement* in the SPEL+ Language Reference manual.

Example

To turn off motors, run the AOI as shown below.



SPEL_MotorOn

Description

Turns robot motors on.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *Motor Statement* in the SPEL+ Language Reference manual.

Example

To turn on motors, run the AOI as shown below.

SetMotorOn			[NOP]
	Turn	s robot motors	
		on.	
	SPEL_I	MotorOn	
	Turns robot n	notors on.	
	SPEL_MotorC	🔊 MotorOn [(InCycle)
	Start	SetMotorOn	
		1 🗲	(Done)
	ExtInputs	Epson:I	
	ExtOutputs	Epson:O	(Error)
	ErrCode1	0 🗲	
	ErrCode2	0 🗲	

5. SPEL AOI Reference

SPEL_Move

Description

Moves the arm from the current position to the specified position in a linear interpolation motion.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Point INT desired point number.

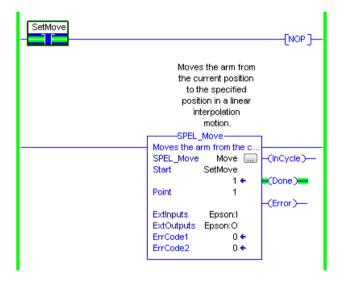
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *Move Statement* in the SPEL+ Language Reference manual.

Example

To move the end effector to point P1, run the AOI as shown below.



SPEL NoFlip

Description

Sets the wrist orientation of the specified point to NOFLIP.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Point INT desired point number.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Wrist Statement in the SPEL+ Language Reference manual

Example

To set point P2 orientation to NoFlip, run the AOI as shown below.

SetNoFlip	
50	
	Sets the wrist orientation of the specified point to NOFLIP.
	SPEL_NoFlip
	Sets the wrist orientation SPEL_NoFlip NoFlip(InCycle)
	1 ← = (Done) ==
	Point 2
	-(Error)
	Extinputs Epson:I
	ExtOutputs Epson:O
	ErrCode1 0 +
	ErrCode2 0 🗲

SPEL_Off

Description

Turns an output bit off.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Bit INT desired output bit number.

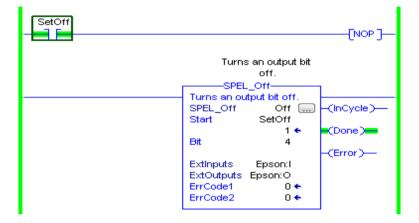
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Off Statement in the SPEL+ Language Reference manual.

Example

To turn off bit number 4, run the AOI as shown below.



SPEL_On

Description

Turns an output bit on.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Bit INT desired output bit number.

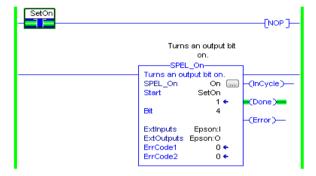
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *On Statement* in the SPEL+ Language Reference manual.

Example

To turn on bit number 4, run the AOI as shown below.



SPEL_Out

Description

Sets an output byte to a given value.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

PortNumINT desired output port number.outDataINT desired output port value.

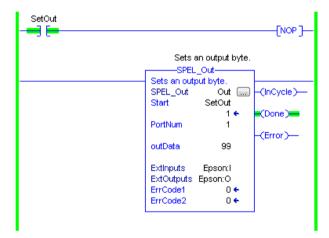
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *Out Statement* in the SPEL+ Language Reference manual.

Example

To set port number 1 with value of 99, run the AOI as shown below.



SPEL_OutW

Description

Sets an output word to a given value.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

PortNum	INT desired output port number.
outData	INT desired output port value.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *OutWStatement* in the SPEL+ Language Reference manual.

Example

To set port number 0 with value of 99, run the AOI as shown below.

SetOutW	[NOP]
	(*** J
	Sets an output word.
	SPEL_OutVV
	Sets an output word.
	SPEL_OutW OutW (InCycle)-
	Start SetOutW
	1 ← <mark>=(</mark> Done) =
	PortNum 0
	(Error)-
	outData 99
	Extinputs Epson:I
	ExtOutputs Epson:O
	ErrCode1 0 +
	ErrCode2 0 +

SPEL_PowerHigh

Description

Sets the power level of robot to high.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

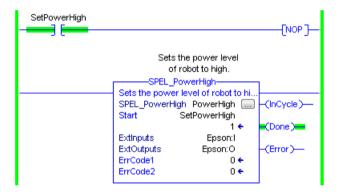
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Power Statement in the SPEL+ Language Reference manual.

Example

To set power high to the robot, run the AOI as shown below.



SPEL PowerLow

Description

Sets the power level of robot to low.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *Power Statement* in the SPEL+ Language Reference manual.

Example

To set power low to the robot, run the AOI as shown below..

SetPowerLow			[NOP]
		s the power level of robot to low.	
	SPEL_	PowerLow er level of robot to lo	w
	SPEL_PowerL Start	.ow PowerLow SetPowerLow	
	ExtInputs ExtOutputs	1 < Epson:I Epson:O	(Done)
	ErrCode1 ErrCode2	0 • 0 •	

SPEL_ResetError

Description

Reset the robot controller AOI error state. After an error has occurred while executing a SPEL AOI, you must execute SPEL ResetError successfully before you can execute another AOI.

Note: If the controller has a system error, then it must be reset before SPEL_Init and other SPEL AOIs can execute successfully.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

SPEL Righty

Description

Sets the hand orientation of the specified point to Righty.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Point INT desired point.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Hand Statement in the SPEL+ Language Reference manual

Example

To set orientation of P2 to Righty, run the AOI as shown below.

SetRighty	[NOP]
	Sets the hand orientation of the specified point to Righty.
	SPEL_Righty
	SPEL_Righty Righty(InCycle)
	Start SetRighty 1 ← ■(Done)■■
	Point 2
	-(Error)
	Extinputs Epson:I ExtOutputs Epson:O
	ExtOutputs Epson:O ErrCode1 0 €
	ErrCode2 0 €

SPEL_SavePoints

Description

Saves the current point data in robot controller memory to the default point file for robot 1 (robot1.pts) in the robot controller. To use this command, a valid RC+ project must exist in the controller. Typically, SavePoints is used to save points taught using the SPEL_Teach AOI. When the controller starts up, it loads the project and the default point file, so the saved points are in memory.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

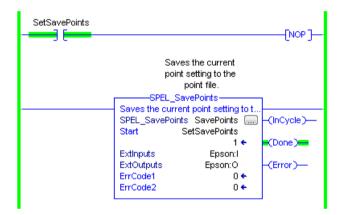
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to SavePoints Statement in the SPEL+ Language Reference manual

Example

To save all points in robot controller memory to the file robot1.pts in the robot controller, run the AOI as shown below.



SPEL Speed

Description

Sets the arm speed setting for PTP motion.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

SpeedINT desired speed.ApproSpeedINT desired approach speed, units are %.DepartSpeedINT desired depart speed, units are %.

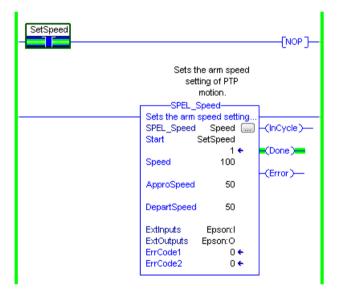
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Speed Statement in the SPEL+ Language Reference manual.

Example

To set Speed to 100%, Approach, Depart Speed to 50%, run the AOI as shown below.



SPEL_SpeedS

Description

Sets the arm speed setting of CP motion. This will set the depart, and approach speed as well.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

SpeedINT desired speed.ApproSpeedINT desired approach speed.DepartSpeedINT desired depart speed.

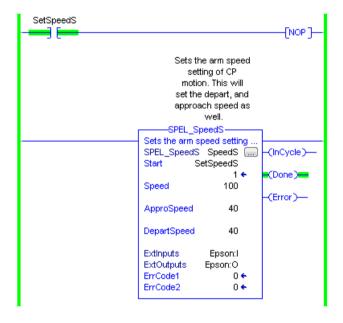
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to SpeedS Statement in the SPEL+ Language Reference manual.

Example

To set Speed to 100, Approach, Depart Speed to 40, run the AOI as shown below.



SPEL_Sw

Description

Reads the status of an input bit.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Bit INT desired input bit.

Outputs

Value INT the value of the input bit.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Sw Function in the SPEL+ Language Reference manual.

Example

To read the value of input bit number 514, run the AOI as shown below.

SetSw			[NOP]
	8	ls the status of an input bit. _ Sw	
		status of an in Sw	-(InCycle)
	Start	SetSw	(Done)
	Bit Value	514 1 •	-(Error)
	ExtInputs ExtOutputs	Epson:I	
	ErrCode1 ErrCode2	0 • 0 •	

5. SPEL AOI Reference

SPEL_Teach

Description

Teaches specified robot point in the robot controller to the current robot position.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

Point INT desired point.

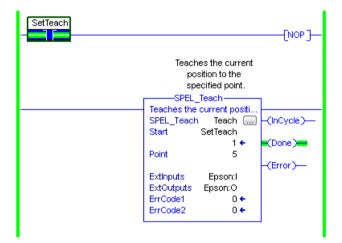
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Here Statement in the SPEL+ Language Reference manual.

Example

To teach current robot position for robot point P5, run the AOI as shown below.



SPEL_ToolGet

Description

Gets the tool selection status.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Outputs

ToolNum INT the tool selected.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *Tool Function* in the SPEL+ Language Reference manual.

Example

To read the selected tool by the robot, run the AOI as shown below.

SetToolGet		[NOP]
	aquires the too selection statu SPEL_ToolGet	IS
	aquires the tool selection SPEL_ToolGet ToolGet Start SetToolGet	:(InCycle)
		← ■(Done)■■

SPEL_ToolSet

Description

Sets the tool.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

ToolNum INT the tool to be set.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *Tool Statement* in the SPEL+ Language Reference manual.

Example

To set current tool to 3, run the AOI as shown below.

SetToolSet			[NOP]
			Urion D
	G	ets the tool	
	-	ection status.	
		ToolSet	
		selection status.	
		t ToolSet	-(InCycle)
	Start	SetToolSet	
		1 🗲	(Done)
	ToolNum	3	
			-(Error)
	ExtInputs	Epson:I	
	ExtOutputs	Epson:O	
	ErrCode1	0 🔶	
	ErrCode2	0 🗲	
			J

SPEL_WeightGet

Description

Gets the hand weight and arm length parameters.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

HandWeight	REAL weight of the hand.
ArmLength	REAL length of the arm.

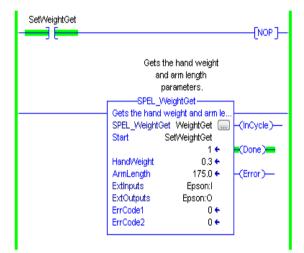
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to Weight Function in the SPEL+ Language Reference manual.

Example

To get the current hand weight and arm length, run the AOI as shown below.



SPEL_WeightSet

Description

Sets the weight parameter.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

HandWeight	REAL weight of the hand.
ArmLength	REAL length of the arm.

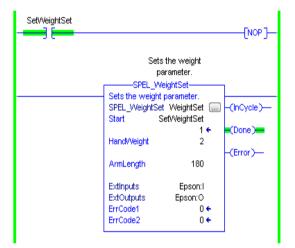
Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to *Wait Statement* in the SPEL+ Language Reference manual.

Example

To set the hand weight and arm length, run the AOI as shown below.



SPEL_XYLimGet

Description

Gets the value of the allowable motion area by specifying the lower and upper limit positions.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Outputs

XLower	REAL X lower limit.
Xupper	REAL X upper limit.
YLower	REAL Y lower limit.
Yupper	REAL Y upper limit.
ZLower	REAL Z lower limit.
Zupper	REAL Z upper limit.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to XYLim Function in the SPEL+ Language Reference manual.

Example

To get the upper and lower limits of X,Y and Z, run the AOI as shown below.

SetXYLimGet			[NOP]
	Gets	the setting	
	val	ue of the	
	allow	able motion	
	area b	y specifying	
	the low	er and upper	
		limit	
	р	ositions.	
	SPEL_XY	LimGet ———	
	Gets the setting	value of the allo	
		XYLimGet 🛄	-(InCycle)
	Start S	etXYLimGet	
		1 🗲	(Done)
	XLower	10.0 🗲	
	XUpper	100.0 🗲	-(Error)
	YLower	20.0 🗲	
	YUpper	200.0 🗲	
	ZLower	30.0 🗲	
	ZUpper	300.0 🗲	
	ExtInputs	Epson:I	
	ExtOutputs ErrCode1	Epson:O	
	ErrCode1	0	
	ErrCode2	0 🕈	

SPEL_XYLimSet

Description

Sets the allowable motion area by specifying the lower and upper limit positions.

Common Inputs and Outputs

Refer to section 2.4 SPEL AOI Common Inputs and Outputs.

Inputs

XLower	REAL X lower limit.
Xupper	REAL X upper limit.
YLower	REAL Y lower limit.
Yupper	REAL Y upper limit.
ZLower	REAL Z lower limit.
Zupper	REAL Z upper limit.

Operation

Refer to section 2.5 SPEL AOI General Operation.

Refer to XYLim Statement in the SPEL+ Language Reference manual.

Example

To set the upper and lower limits of X,Y and Z, run the AOI as shown below.

SetXYLimSet			[NOP]
	ma sr Iow Iin	the allowable tion area by becifying the er and upper hit positions. (YLimSet	
	Sets the allow SPEL_XYLimS	able motion area	-(InCycle)
	XLower	1 • 10	=(Done)== -(Error)
	XUpper YLower	100 20	
	YUpper	200	
	ZLower ZUpper	30 300	
	ExtInputs ExtOutputs	Epson:I Epson:O	
	ErrCode1 ErrCode2	0 • 0 •	

6. Error Codes

Each AOI has an Error output bit and two INT error codes: ErrCode1 and ErrCode2. The error output is set to high when an error has occurred, and ErrCode1, ErrCode2 indicate which error has occurred as described in the table below.

ErrCode1	ErrCode2	Description	Cause/Remedy
0x200A (8202)	1 -9999	A robot controller error has occurred. ErrCode2 is the robot controller error.	See the section <i>SPEL</i> + <i>Error Messages</i> in the SPEL+ Language Reference manual.
0x200B (8203)	0	Command not accepted by the controller	The controller is in a state where it cannot accept commands. Power cycle the controller.
0x3000 (12288)	0x280A (10250)	AOI execution timeout	Ethernet communications was lost during command execution, or the command took too long to execute.
0x3000 (12288)	0x280B (10251)	Cannot execute instruction because of previous error or ExtReset input in the controller is low.	After any error has occurred, SPEL_ResetError must be executed.
0x3000 (12288)	0x280C (10252)	Cannot execute instruction because of invalid robot controller configuration.	Check that Remote I/O and PLC Vendor settings are correct in the robot controller.
0x3000 (12288)	1 -9999	A robot controller error has occurred. ErrCode2 is the robot controller error.	See the section SPEL+ Error Messages in the SPEL+ Language Reference manual.