

A Sierra Monitor Company

SlotServer User Manual

FS-RA-CLX-LON-000 SlotServer LonWorks Open Interface

APPLICABILITY & EFFECTIVITY

Effective for all systems manufactured after May 1, 2001

Firmware Version:0.00Document Revision:6

TABLE OF CONTENTS

INTRODUCTION. 1.1. About this product. 1.2. About LonWorks and Profiling. 1.3. Connection Capability.	.3 .3 .3
 PRE-STARTUP CHECKLIST 2.1. Accessories supplied with the SlotServer 2.2. Third party components (not supplied by FieldServer Technologies) 2.3. Required configuration for the SlotServer 	.4 .4 .4 .4
SLOTSERVER TOPOLOGY 3.1. Hardware Connection Tips / Hints	.5 .5
 4. SLOTSERVER INSTALLATION 4.1. SlotServer Ports and Displays 4.1.1. Scrolling Display 4.1.2. Card Status Indicators: 4.1.3. Service Pin 4.1.4. Communications Status LED's 4.1.5. TP/FT10 LonWorks connection 4.1.6. Trade Port 4.2. Inside Cover Door Label 4.3. Technical Specifications 	.6 .6 .6 .7 .7 .7 .7
 5. SLOTSERVER COMMUNICATIONS QUICK START EXAMPLE PROJECT 5.1. Step 1: Create or use an existing RSLogix project 5.2. Step 2: Add and configure the SlotServer as an IO Module 5.3. Step 3: Bind LonWorks Variables 5.4. Step 4: Download the RSLogix Program and Run 5.5. Step 5: Interpret the Network Variable Data 5.6. Step 6: (Optional) Arrange Data into User Defined Tags 	.8 .9 11 11 12
6. COMPLETE SLOTSERVER COMMUNICATIONS PROJECT 1 6.1. Step 1: Add the SlotServer Module 2 6.2. Step 2: Create two User Defined Data Types 2 6.3. Step 3: Add and Configure a Read(In) Message Block 2 6.4. Step 4: Add and Configure a Write(Out) Message Block 2 6.5. Step 5: Assign Data Array Names 2 6.6. Step 6: Create logic to control the Message Blocks 2 6.7. Step 7: Save and download the RSLogix program 2 6.8. Step 8: Commission the LonWorks NetWork 2	15 16 17 22 23 24 24 25
APPENDIX A. ADVANCED TOPICS	26 26
APPENDIX B. TROUBLESHOOTING TIPS	27 27 27
APPENDIX C. USING LONMAKER TO COMMISSION A SLOTSERVER	29 E
30 Appendix D.1. LonWorks Network Variables Summary	30

1. Introduction

1.1. About this product

The SlotServer LonWorks Open Interface allows an Allen Bradley ControlLogix platform to pass data between a ControlLogix CPU and a standard LonWorks communications network. The SlotServer Module provides a standard generic profile in LonWorks that allows most of the more common Standard Network Variable Types (SNVT's) to be bound to the SlotServer for data transfer.

The SlotServer Module is configured in a ControlLogix Rack using standard RSLogix 5000 I/O configuration combined with "CIP Generic" message block calls to allow for efficient transfer of LonWorks data.

1.2. About LonWorks and Profiling

Inherent to the design of LonWorks is the intention that all nodes residing on a LonWorks network possess a profile which accurately describes the network variables available to the LonWorks network. This profile is declared to the LonWorks network by means of an External Interface File (XIF). This XIF remains unchanged for any particular profile. In other words, a change in the XIF is by definition a new profile.

The Open Interface profile for SlotServer provides a generic list of SNVT's that can be used to bind the most common LonWorks data types to the SlotServer for data transfer. This document describes the fixed Open Interface profile for the SlotServer. As a LonMark compatible product, the SlotServer platform is also capable of supporting other profiles. For further information, please contact FieldServer Technologies and enquire about available Specific Profiles and other SlotServer Profiling Services.

SLOTSERVER MODE	LONWORKS NODES	COMMENTS
CLIENT (SCANNER)	15	As an implicitly bound object in LonWorks, the LonWorks protocol supports the binding of this device to no more than 15 other nodes.
SERVER (ADAPTER)	N/A	The ControlLogix CPU is always configured as a scanner to communicate with the SlotServer adapter.

1.3. Connection Capability

2. **Pre-Startup checklist**

2.1. Accessories supplied with the SlotServer

Please verify that the following components were supplied with the SlotServer module:

FieldServer Technologies PART #	Description	
	2-way combi-comb connector for LonWorks port	
	Software CD	
	CAT-5 Direct Ethernet Cable	
	Documentation Binder	

2.2. Third party components (not supplied by FieldServer Technologies)

The following components will need to be procured as a minimum for achieving a LonWorks/ControlLogix integration. Ensure that these are available before commencing with commissioning

Part #	Description	
AB1756-Ax	ControlLogix Rack	
AB1756-L55/L61	ControlLogix CPU	
AB1756-PA7x	ControlLogix PSU	
9324-RLD300ENE	RSLOGIX 5000	
	LonWorks NetWork Manager (e.g.: LonMaker)	
	LonWorks Network components.	

2.3. Required configuration for the SlotServer

In order to achieve data transfer between CPU tags and LonWorks network variables, the SlotServer will need to be commissioned into a LonWorks network using a LonWorks Network Manager. Once commissioned and bound, a subset of the Data from the LonWorks network is accessed by I/O image transfer or the full amount of data by programming MSG blocks in the ControlLogix CPU. The ControlLogix CPU uses CIP Generic Custom read and write operations to achieve data transfer from the SlotServer Module to the CPU tags.

3. SlotServer Topology

The diagram below shows a typical implementation of the SlotServer LonWorks Open Interface Module



3.1. Hardware Connection Tips / Hints

- The LonWorks TP/FT-10 medium is polarity insensitive and therefore does not require attention to connection orientation on the pins.
- Follow the Rockwell documentation for guidelines on inserting the SlotServer module into the ControlLogix backplane / rack.

4. SlotServer Installation

The SlotServer plugs directly into the AB1756 ControlLogix rack. Take note of the slot number used for the SlotServer as this will be needed when configuring the I/O in the RSLogix software.

4.1. SlotServer Ports and Displays

Note that there are several hardware ports available on the SlotServer card, but that this SlotServer model only makes use of a select few. The ports and displays of interest are as shown in the diagram below:



4.1.1. Scrolling Display

This will Display an OK status when the SlotServer has powered up successfully.

4.1.2. Card Status Indicators:

The Comm and OK lights will indicate green when the SlotServer has powered up successfully. The Sys indicator will indicate green if the Lon Nodes are all communicating, and red otherwise.

4.1.3. Service Pin

To be used for commissioning the SlotServer into a LonWorks network

4.1.4. Communications Status LED's

Run	When the SlotServer is powered up, this light will start flashing after approximately 2 minutes. If this light does not flash, the firmware is not
	running, and that the SlotServer will not be operating.
LonSvc	This light will flash when the card's Neuron chip is unconfigured, and will extinguish when the SlotServer has been commissioned into a Network.
NodeOff	This light indicates that a device bound to the SlotServer on the LonWorks
	network is not communicating. Use the LonWorks Network Manager to
	troubleshoot the problem.
Conf	This light indicates an error in the configuration of the card. Since this model
Err	is shipped pre-configured, this light should never come on unless the card
	configuration is tampered with
Sys Err	This light indicates that an unexpected operation state occurred. The light
	can be reset by cycling power to the card. If the error re-occurs persistently,
	then contact FieldServer Technical support for assistance with
	troubleshooting after setting up the trade port as shown in Appendix A.
Com Err	This light indicates that a communication error occurred. Use the LonWorks
	NetWork Manager to troubleshoot the problem, or refer to Appendix A.1.

4.1.5. TP/FT10 LonWorks connection

This two wire connection is polarity insensitive and can be multi-dropped into a LonWorks NetWork.

4.1.6. Trade Port

The Trade port is an Ethernet port that can be used for advanced troubleshooting (See Appendix A.1). Use standard Ethernet cables for this connection.

4.2. Inside Cover Door Label

Note that a label has been provided on the inside of the cover door for the User to document important or useful configuration parameters used. In particular, special provision has been made to document the Trade port IP parameters, as well as the LonWorks DSN (Domain/Subnet/Node) allocated by the Network Manager. These fields are there to be utilized at the User's discretion.

Slot width	1
BackPlane Current Load	0.6A
Operating Temperature	0-60 Deg C (32-140 Deg F)
Storage Temperature	-40 – 85 Deg C (-40 – 185 Deg F)
Humidity	5-95% (Non-Condensing)
Allen Bradley Platform:	ControlLogix 1756 Rack
LonWorks Connection type:	TP/FT-10
LonWorks Connection	As an implicitly bound object in LonWorks, the LonWorks protocol
Capability ¹	supports the binding of this device to no more than 15 other nodes.
LonWorks Point Capability	See Appendix D

4.3. Technical Specifications

¹ A LonWorks Network Management Tool (e.g.: LonMaker,Rover, LonWatcher etc) is required to commission a LonWorks network. This tool is used for creating the variable bindings on the network.

FieldServer Technologies 1991 Tarob Court Milpitas, California 95035 USA Web:www.fieldserver.com Tel: (408) 262-2299 Fax: (408) 262-9042 Toll_Free: 888-509-1970 email: support@fieldserver.com

5. SlotServer Communications Quick Start Example Project

The discussion that follows describes the basic steps to set up and test the system for transferring data between CPU tags and the SlotServer using the I/O image method. 84 Network Variables of the full 2,600 points are available using this easy to get going method. Refer to the Advanced Project if you wish to access more than 84 Network Variables.

5.1. Step 1: Create or use an existing RSLogix project

- Use File, New to create a new project or File, Open to open an existing project.
- Make sure to choose the correct Slot number in the rack where your Controller resides.

New Control	er	
Vendor:	Allen-Bradley	
Туре:	1756-L61 ControlLogix5561 Controller	OK
Revision:	13 💌	Cancel
	Redundancy Enabled	Help
Name:	QuickStart	
Description:	SlotServer QuickStart Project	
Chassis Type:	1756-A7 7-Slot ControlLogix Chassis	
Slot:	0 .	
Create In:	C:\RSLogix 5000\Projects	Browse

Note: Your Controller may be of a different type to the one shown in the example.

5.2. Step 2: Add and configure the SlotServer as an IO Module

- Right-click on I/O Configuration and select "New Module".
- Choose the 1756-MODULE

	1.223		
no RSLogix 2000 - QuickStart [1726	-L61]		
File Edit View Search Logic Communi	ations Tools Window Help		
	CM	- 1888 1: 19	
Offline	Path: <none></none>	_ 品	
	The second secon		
No Edits		FAL FSC COP FLL AVE SRT STD	
Redundancy 👦	Compute/Math	n 🔏 Move/Logical 👌 File/Misc. 🦨 File/Shift 🔏 Se	
🕞 🚔 Gasharlar Quid-Chart			
Controller QuickStart			
Controller Fault Handler	Select Module Type		
Power-Up Handler	THE 175C MODULE	_	
🖻 🖯 Tasks	Type: TYSOMODOLL		
🖻 🧔 MainTask	Type Desci	ription	
⊕ La set adulad Des serves a	1756-M02AE 2 Axis	Analog/Encoder Servo	<u>^</u>
Motion Groups	1756-MUZAS Z AXIS	: Analog/551 Servo	
	1756-M035E 3 Axis	SEBCOS Interface	
Trends	1756-M16SE 16 Ax	is SERCOS Interface	
🚊 📇 Data Types	1756-MODULE Gener	ric 1756 Module	
🖳 🔤 User-Defined	1756-0A16 16 Po	int 74V-265V AC Output	
庄 🗔 Strings	1756-0A16l 16 Po	int 74V-265V AC Isolated Output	
	1756-0A8 8 Poir	ht 74V-265V AC Output	
	1756-0A8D 8 Poir	nt 74V-132V AC Diagnostic Output	
I/O Configuration	1756-UA8E 8 Poir	nt /4V-132V AU Electronically Fused Uutput	
	1756-0B16E 16 Po	int 16/231 2V DC Electronically Eused Output	
	1756-0B16I 16 Po	int 10V-30V DC Isolated Output, Sink/Source	~
	- Show	• •	
	Vendor: All	▼ IV Other IV Specialty I/O Select	All
	i origon [i iii		
	🔽 Analog 🔽 Digital 🔽	Communication 🔽 Motion 🔽 Controller Clear	All
		OK Cancel Help	>

- Configure the SlotServer Module using the settings as shown below:
- Be sure to choose the correct Slot number in the rack where your SlotServer resides. The example below uses Slot number 5.

🖁 RSLogix 5000 - QuickStart [1756-L61]		
File Edit View Search Logic Communications Tools	is Window Help	
	- 38 1 29	
Diffine I + RUN No Forces C + OK No Edits A HAT Redundancy M	Path: <none> FAL FSC COP FLL AVE SRT STD Compute/Math & Move/Logical File/Misc. & File/Shift & Se</none>	
Controller QuickStart Controller Tags Controller Fault Handler Power-Up Handler Tasks MainTask MainProgram Unscheduled Programs Unscheduled Progra	Module Properties - Local:1 (1756-MODULE 1.1) Type: 1756-MODULE Generic 1756 Module Parent: Local Name: SlotServer Description: Imput: 1 63 Comm Format: Data - REAL Slot: 5 Cancel Mext > Finish >> Help)

 Click Next and choose a RPI of 750 ms. This is the rate at which the I/O image data will be transferred.

🔀 RSLogix 5000 - QuickStart [1756-L61]	
File Edit View Search Logic Communications Tools	Window Help
	- <u>&&&</u> <u>F</u> <u>9</u>
Offline Image: Constraint of the second se	Path: Image: Constant of the sector of t
Controller QuickStart Controller Tags Controller Fault Handler Power-Up Handler Tasks Gamma Start MainTask Gamma Start Motion Groups Motion Groups	Module Properties - Local:5 (1756-MODULE 1.1) Requested Packet Interval (RPI): 750.0 + ms Inhibit Module Inhibit Module Major Fault On Controller If Connection Fails While in Run Mode Module Fault Cancel < Back

• Click Finish to complete the Module Properties setup.

5.3. Step 3: Bind LonWorks Variables

The SlotServer's LonWorks Network Variables must be bound to other devices using a Network Management Tool such as LonMaker. See Appendix C for more details on how to use a Network Management Tool to make the bindings.

5.4. Step 4: Download the RSLogix Program and Run

Use the Who Active or Communications Path directly to Download and Run the Program on the Controller/CPU.



5.5. Step 5: Interpret the Network Variable Data

LonWorks Input Variable Values can be found under the Local: 5:I.Data Tag and LonWorks Output Variable Values must be written into the Local: 5:O.Data

The :5: denotes the Slot number in the Rack that contains the SlotServer - your setup may use a different number.

The Variables are arranged as follows for the Quick-Start example project:

		NV name	
Input Data Tag	Output Data Tag	(i) for Input	LonWorks SNVT Type
1		(o) for Output	
			SNVT chlr status:
Local:5:1.Data[0]	Local:5:0.Data[0]		chlr_run_mode
Local:5:1 Data[1]			SNVT_chlr_status:
	Local.5.0.Data[1]		chlr_op_mode
Local:5:1 Data[2]	Local:5:0 Data[2]		SNVT_chlr_status:
Local.o.i.Data[2]	Local.o.O.Data[2]		chlr_state
Local:5:I.Data[3]	Local:5:O.Data[3]	nv(i)(o)Count1_01	SNVT_count_f
Local:5:I.Data[4]	Local:5:O.Data[4]	nv(i)(o)Count2_01	SNVT_count_f
Local:5:I.Data[5]	Local:5:O.Data[5]	nv(i)(o)CtInc1_01	SNVT_count_inc_f
Local:5:I.Data[6]	Local:5:O.Data[6]	nv(i)(o)CtInc2_01	SNVT_count_inc_f
Local:5:I.Data[7]	Local:5:O.Data[7]	nv(i)(o)Flow1_01	SNVT_flow_f
Local:5:I.Data[8]	Local:5:O.Data[8]	nv(i)(o)HVMod1_01	SNVT_hvac_mode
Local:5:1 Data[9]	Local:5:0 Data[9]		SNVT_hvac_status:
LoodioinData[0]	Loodilo.O.Data[0]	-	hvac_sts_mode
Local:5:I.Data[10]	Local:5:O.Data[10]		SNVT_hvac_status:
			hvac_heat_out_pri
Local:5:I.Data[11]	Local:5:O.Data[11]		SNVI_hvac_status:
		-	NVac_neat_out_sec
Local:5:I.Data[12]	Local:5:O.Data[12]	nv(i)(o)HVSts1_01	byac cool out
		-	SNVT byac status:
Local:5:I.Data[13]	Local:5:O.Data[13]		hvac econ out
			SNVT hvac status:
Local:5:I.Data[14]	Local:5:O.Data[14]		hvac fan out
			SNVT hvac status:
Local:5:1.Data[15]	Local:5:0.Data[15]		hvac_in_alarm
Local:5:I.Data[16]	Local:5:O.Data[16]	nv(i)(o)Lev1_01	SNVT_lev_cont
Local:5:I.Data[17]	Local:5:O.Data[17]	nv(i)(o)Occ1_01	SNVT_occupancy
Local:5:I.Data[18]	Local:5:O.Data[18]	nv(i)(o)Occ2_01	SNVT_occupancy
Local:5:I.Data[19]	Local:5:O.Data[19]	nv(i)(o)Perc1_01	SNVT_lev_percent
Local:5:I.Data[20]	Local:5:O.Data[20]	nv(i)(o)Perc2_01	SNVT_lev_percent
Local:5:I.Data[21]	Local:5:O.Data[21]	nv(i)(o)Perc3_01	SNVT_lev_percent
Local:5:I.Data[22]	Local:5:O.Data[22]	nv(i)(o)Perc4_01	SNVT_lev_percent
Local:5:I.Data[23]	Local:5:O.Data[23]	nv(i)(o)Perc5_01	SNVT_lev_percent
Local:5:I.Data[24]	Local:5:O.Data[24]	nv(i)(o)Perc6_01	SNVT_lev_percent
Local:5:I.Data[25]	Local:5:O.Data[25]	nv(i)(o)Perc7_01	SNVT_lev_percent
Local:5:I.Data[26]	Local:5:O.Data[26]	nv(i)(o)Perc8_01	SNVT_lev_percent
Local:5:I.Data[27]	Local:5:O.Data[27]	nv(i)(o)PPM1_01	SNVT_ppm

FieldServer Technologies 1991 Tarob Court Milpitas, California 95035 USA Web:www.fieldserver.com Tel: (408) 262-2299 Fax: (408) 262-9042 Toll_Free: 888-509-1970 email: support@fieldserver.com

Input Data Tag Output Data Tag		NV name (i) for Input	I onWorks SNVT Type
		(o) for Output	
Local:5:I.Data[28]	Local:5:O.Data[28]	nv(i)(o)Press1_01	SNVT_press
Local:5:I.Data[29]	Local:5:O.Data[29]	nv(i)(o)Press2_01	SNVT_press
Local:5:1 Data[30]			SNVT_temp_setpt:
LUCALS.I.DAIA[30]	Local.5.0.Data[50]		Occupied_cool
Local:5:I.Data[31]	Local:5:O.Data[31]		SNVT_temp_setpt: Standby_cool
Local:5:I.Data[32]	Local:5:O.Data[32]	$n_{i}(i)(a)$ Satat1 01	SNVT_temp_setpt: Unoccupied_cool
Local:5:I.Data[33]	Local:5:O.Data[33]		SNVT_temp_setpt: Occupied_heat
Local:5:I.Data[34]	Local:5:O.Data[34]		SNVT_temp_setpt: Standby_heat
Local:5:I.Data[35]	Local:5:O.Data[35]		SNVT_temp_setpt: Unoccupied_heat
Local:5:I.Data[36]	Local:5:O.Data[36]	nv(i)(o)State1_01	SNVT_state
Local:5:I.Data[37]	Local:5:O.Data[37]	nv(i)(o)State2_01	SNVT_state
Local:5:I.Data[38]	Local:5:O.Data[38]	nv(i)(o)State3_01	SNVT_state
Local:5:I.Data[39]	Local:5:O.Data[39]	nv(i)(o)State4_01	SNVT_state
Local:5:I.Data[40]	Local:5:O.Data[40]	$p_{V}(i)(o)S_{W}(1) = 01$	SNVT_switch
Local:5:I.Data[41]	Local:5:O.Data[41]	110(1)(0)301_01	SNVT_switch
Local:5:I.Data[42]	Local:5:O.Data[42]	$p_{V}(i)(a)S_{W}^{2}$ 01	SNVT_switch
Local:5:I.Data[43]	Local:5:O.Data[43]	110(1)(0)3W2_01	SNVT_switch
Local:5:I.Data[44]	Local:5:O.Data[44]	$p_{V}(i)(a)S_{W}^{2}$ 01	SNVT_switch
Local:5:I.Data[45]	Local:5:O.Data[45]	110(1)(0)3003_01	SNVT_switch
Local:5:I.Data[46]	Local:5:O.Data[46]	$p_{V}(i)(o)S_{W}(1,0)$	SNVT_switch
Local:5:I.Data[47]	Local:5:O.Data[47]	110(1)(0)3004_01	SNVT_switch
Local:5:I.Data[48]	Local:5:O.Data[48]	$n_{1}(i)(0)Sw5 01$	SNVT_switch
Local:5:I.Data[49]	Local:5:O.Data[49]	110(1)(0)5005_01	SNVT_switch
Local:5:I.Data[50]	Local:5:O.Data[50]	$n_{V}(i)(a)S_{W}6$ 01	SNVT_switch
Local:5:I.Data[51]	Local:5:O.Data[51]	110(1)(0)300_01	SNVT_switch
Local:5:I.Data[52]	Local:5:O.Data[52]	$p_{V}(i)(o)S_{W}Z_{0}$	SNVT_switch
Local:5:I.Data[53]	Local:5:O.Data[53]	110(1)(0)3W7_01	SNVT_switch
Local:5:I.Data[54]	Local:5:O.Data[54]	$p_{i}(i)(a) S_{i}(a) = 01$	SNVT_switch
Local:5:I.Data[55]	Local:5:O.Data[55]	110(1)(0)3wo_01	SNVT_switch
Local:5:I.Data[56]	Local:5:O.Data[56]	nv(i)(o)Temp1_01	SNVT_temp_p
Local:5:I.Data[57]	Local:5:O.Data[57]	nv(i)(o)Temp2_01	SNVT_temp_p
Local:5:I.Data[58]	Local:5:O.Data[58]	nv(i)(o)Temp3_01	SNVT_temp_p
Local:5:I.Data[59]	Local:5:O.Data[59]	nv(i)(o)Temp4_01	SNVT_temp_p
Local:5:I.Data[60]	Local:5:O.Data[60]	nv(i)(o)Temp5_01	SNVT_temp_p
Local:5:I.Data[61]	Local:5:O.Data[61]	nv(i)(o)Temp6_01	SNVT_temp_p
Local:5:I.Data[62]	Local:5:O.Data[62]	nv(i)(o)Temp7_01	SNVT_temp_p

5.6. Step 6: (Optional) Arrange Data into User Defined Tags

You can make a User Defined Data Type and create a Tag for each LonWorks Data In and LonWorks Data Out. Use a ladder COPY instruction to copy between the User Defined Data Type Tag and the Local Tag.



6. Complete SlotServer Communications Project

The Complete Project uses Message Blocks to transfer 25 Input and 25 Output Blocks of Lonworks Data. See Appendix D for the Variable List.

6.1. Step 1: Add the SlotServer Module

Follow Steps 1 & 2 of the Quick Start example to add the SlotServer to the RSLogix Project.

6.2. Step 2: Create two User Defined Data Types

- Right-click on Data Types, User-Defined, and select New Data Type.
- Create a new Data Type as shown below to be used with Read(In) type Message Blocks:

🕷 RSLogix 5000 - Project_01 [1756-L61]		
File Edit View Search Logic Communications	Tools Window Help	
	<u></u>	
Dffline 0 → □ RUN	Path: <none></none>	
No Edits BAT	Image: Second	
Controller Project_01 Controller Tags	🔠 Data Type: New UDT1*	
Controller Fault Handler	Name: SlotServer_Data_In	
	Description:	
Motion Groups Ungrouped Axes		
Data Types		
En - Grings ⊕ - Grings ⊕ - Gring Predefined	Members: Data	a Type Size: ?? byte(s)
Module-Defined	Name Data Type Style Description	
I/O Configuration	DA_Name STRING	
	* Paka_m PEAL(70) Float	

• Repeat the procedure to create a new Data Type to be used with Write(Out) type Message Blocks.

🕅 RSLogix 5000 - Project_01 [1756-L61]*					
File Edit View Search Logic Communications	Tools Window Help				
125 5 <u>x 66</u> 20		• & & &	1 📴 🗾	P Q Q	
Offline Image: Controller Project_01 Image: Controller Tags	Path: <none></none>	UDT2*	+ -(L)- Input/Output	▼ 器	
Controller Fault Handler Power-Up Handler Tasks MainTask GMainProgram Unscheduled Programs Unscheduled Programs Unscheduled Axes Trends Data Types User-Defined User-Defined	Name: Description: Members:	SlotServer_Data_Out	_	Data	Tvoe Size: ?? bvte(s)
±- 🙀 Strings ±- 🙀 Predefined	Name	Data Type	Style	Description	
Module-Defined	⊞ DA_Name	STRING			
I/O Configuration	Data_Out	REAL[75]	Float		
	*				

6.3. Step 3: Add and Configure a Read(In) Message Block

- Add an Examine On element and Message Block to the first rung of the ladder logic.
- Right Click on the Message Control ? bar and Select New Tag.
- Configure the Tag by naming it Msg_In_01 as shown below:

🖁 RSLogix 5000 - Project_01 [1756-	L61]*	
File Edit View Search Logic Communica	tions Tools W	indow Help
		<u>_ &&&&</u> <u>_</u> <u>_</u>
Offline 📴 🗸 🗖 BUN	🔯 P	ath: <none> 🖌 🛃</none>
	Y i	
	<u> </u>	
edundancy 👧		Favorites / Bit / Timer/Counter / Input/Output / Compare
Controller Project_01		itroller Tags - Project_UT(controller)
Controller Fault Handler	New Tag	ojt: Base Tag 🔍
Power-Up Handler	Manag	
🖃 📇 Tasks	iname:	
🖻 🤕 MainTask	Description:	AB:1756_MODULE_INT_2
🖻 📲 MainProgram	Description.	
Program Tags		Help
Onscheduled Programs		
Indian Gloups		
Trends	Tag Type:	F Base
🖻 📇 Data Types		C áliae
🖻 🦏 User-Defined		O Produced 1 - consumers
SlotServer_Data_Out		
	Data Type:	MESSAGE Configure
	Scope:	Project_01(controller)
☐ 1/0 Connigatation ☐ [5] 1756-MODULE SlotServer	Stule:	
5.0	0.910.	
		e Message Control ? DN)-
		e
	(End)	

• Click on Configure and use the parameters as shown below for a Read Message block:

🖁 RSLogix 5000 - Project_	_01 [1756-L61]*				
File Edit View Search Logic	Communications	Tools Window Help			
		<u>ه</u> ۲	x & & T= I = Q	<u>.</u>	
Dffline U RUN No Forces C RUN No Edits A RAT Redundancy M		Path: <none></none>	v ssu IOT /Counter Ainput/Output & Comps	Sare	
Controller Project_01 Controller Tags Controller Tags Controller Fault Hang Controller Tags Controller Fault Hang Controller Tasks Controller	Message Config Configuration* (Message Tune:	Controller Tags - Project uration - Msg_In_01 ommunication Tag CIP Generic	_01 (controller)	Z Type AB:1756_MODULE_IN AB:1756_MODULE_IN AB:1756_MODULE_IN	
Program Tag MainRoutine MainRoutine Unscheduled Program Ungrouped Axes Trends Otta Types Guero-Defined Misolocerver_Data Misolocerver_Data Misolocerver_Data Misolocerver_Data Misolocerver_Data Misolocerver_Data Misolocerver_Data Misolocerver_Data	Service Custo Type: Custo Service 4c Code: 1	(Hex) Class: 4 (Hex) Attribute: 8 (Hex)	Source Element: Source Length: 20 (Destination New Tag	(Bytes)	
Generation G	Enable Error Code: Error Path: Error Text:	Enable Waiting O Start (Extended Error Code:	Done Done Length: 0 Timed Out Cancel Apply		
		(End)			_ []

- Click in the Source Element field and then click on New Tag.
- Name the Tag as shown below and select the Read User defined data type "SlotServer_Data_In" you created in the previous Step.



• Select the DA_Name element as the Source Element and the Data_In element as the Destination Element of the Message Block.



• Complete the Message Block Configuration by Browsing to the SlotServer device in the rack on the Communication Tab.



6.4. Step 4: Add and Configure a Write(Out) Message Block

- Add an Examine On element and Message Block to the first rung of the ladder logic.
- Right Click on the Message Control ? bar and Select New Tag.
- Configure the Tag by naming it Msg_Out_01 and Configure the Message Data Type by choosing the SlotServer_Data_Out User Defined Data Type you created in Step 4.
- Configure the MESSAGE Data Type for Write as shown below:

🔞 RSLogix 5000 - Project	01 [1756-L61]*	
File Edit View Search Log	Communications Tools Window Help	
2 2 3 4	E ∽ ∽ _ &&&& E ₽ ₽ Q Q	
Offline 📴 🗖 RUN	Path: <none> ╂</none>	
No Forces		
No Edits		
Redundancy	Favorites & Bit & Timer/Counter & Input/Output & Compare	_
Controller Project_01	Controller Tags - Project_01(controller)	
Power-Up Handler	Aessage Configuration - Msg_Out_01	-
Tasks Tasks MainTask MainProgram Motion Groups Ungrouped Axes Trends Data Types Storserver_Dat Motion SlotServer_Dat Module-Defined Module-Defined Module-Defined Module-Defined T/O Configuration [5] 1756-MODULE :	Configuration* Communication* Tag Message Type: CIP Generic Service Custom Type: Source Element: Out_01 AB:1756_MODULE_INI AB	
	Error Path: Isg_In_01	
	1 e EN) e e e e e e e e e e e e e e e e e e e	>
<	MainRoutine* /	
Ready		1

• Finally, click on the Communication tab and browse to the SlotServer to set the communications path.

6.5. Step 5: Assign Data Array Names

Each Message Block's Source Element Tag holds an element that addresses a data array on the SlotServer.

 Open the In_01 CPU Tag and edit the DA_Name element adding the name of the data array as shown below: The format is DA_Name[offset][number of elements]. Note that the LonMark certified configuration supplied with the SlotServer is already configured for 75 elements per data array, so use the syntax as shown below:



• Do the same for the Out_01 CPU Tag:

🔀 RSLogix 5000 - Project_01 [1756-L61]*					
File Edit View Search Logic Communications	Tools Window Help				
		• <u>&</u> &&			
	Path: AB_ETHIP-1\1	92.168.1.34\Back	plane\0*	→ 器	
No Edits BAT Redundancy NO		a asv ssv II Timer/Counter	זד] ∖Input/Output ג	Compare	
Controller Project_01	🖉 Controller Tags - Pr	oject_01(cont	roller)		
Controller Fault Handler	Scope: Project_01(contro	ller V Show: Sh	iow All	▼ Sort: Bas	æTag 🗾
Power-Up Handler	Tag Name	Value 🗲 Fo	orce Mask 🛛 🗲	Style	Туре
	+ Local:5:1	{}	{}		AB:1756_MODULE_INT_252
😑 🚭 MainProgram	+ Local:5:C	{}	{}		AB:1756_MODULE:C:0
Program Tags	+ Msg_In_01	{}	{}		MESSAGE
MainRoutine		{}	{}		SlotServer_Data_In
Unscheduled Programs	+ In_01.DA_Name	'In_0	{}		STRING
	+ In_01.Data_In	{}	{}	Float	REAL[75]
Trends	+ Msg_Out_01	{}	{}		MESSAGE
🖻 🔠 Data Types	Out_01	{}	{}		SlotServer_Data_Out
E 🤤 User-Defined			{}		STRING
SlotServer_Data_In	String Browner Out 01	DA Namo*			<u>▶</u>
+ Strings	string browser - Out_OT	.DA_Maine			
Predefined	Out 01[0][75]		~	\$\$	
🛨 🚂 Module-Defined				\$°	
E 🔄 I/O Configuration				\$L	
[5] 1756-MODULE SlotServer				MSG-	A
				and and	Msg In 01
1				\$P	-œR)-
				\$R	
			19	\$T	
P	osition: 13 Count: 13 of 82		< Errors ≥	c	
		-	3.2		Msg_Out_01 0N)
	OK Cancel	Apply	Help		
Enter a tao value					

6.6. Step 6: Create logic to control the Message Blocks

A simple method for testing is to create a Bool Tag that is examined by the two "Examine On" elements of the Message Blocks. The Tag value can be manually set to 1 and 0 to trigger and test for data transfer.

6.7. Step 7: Save and download the RSLogix program.

After Saving and Downloading, be sure to Set the Controller to Run mode.

6.8. Step 8: Commission the LonWorks NetWork

To effect actual transfer of LonWorks data, the next step is to present the SlotServer profile to the LonWorks network by commissioning the SlotServer into the network with a LonWorks Network Manager. The precise method for doing this varies depending on the Network Manager used, but Appendix C shows how this can be achieved using LonMaker for Windows. Other Network managers follow similar philosophies to achieve the same goal.

Once the SlotServer is commissioned, it can be bound to the variables of interest.

Communications should now be enabled between the LonWorks variables and the CPU tags. If the data is not being transferred, refer to Appendix B for hints and tips on troubleshooting the application.

The detailed tag list available for data transfer can be found in Appendix D. Should more advanced operation or functionality be required than described above, then refer to Appendix A for documentation of advanced topics.

Appendix A. Advanced Topics

Appendix A.1. Using the FieldServer RUI for troubleshooting.

As a FieldServer enabled product, the SlotServer supports the use of the FieldServer Utilities for diagnostic purposes. It should not be necessary to connect these utilities to the SlotServer. However, if the need arises, then the following steps are advised:

- Download the utilities software (or just RuiNet) from www.fieldserver.com
- Download the FieldServer Utilities manual
- Get familiar with the utilities by reading the Utilities manual
- Connect to the Ethernet port on the SlotServer as prescribed by the FieldServer Utilities manual.
- Follow directions on each of the features of the Utilities to achieve the desired result

<u>Note:</u> It is possible to change the profile of the SlotServer by loading a new configuration with these utilities. However, doing so for this particular product will void the Warranty on the product and doing so without consulting FieldServer Technologies is strongly discouraged.

Appendix B. Troubleshooting tips

Appendix B.1. Things to check when communications has failed.

- Check for loose cabling on the LonWorks network
- Verify that the bindings in the LonWorks network have not been broken by checking the variable status' with the LonWorks Network Manager.
- If the LonWorks Network Manager shows the communications to be healthy, then check the tag names used in the MSG blocks in the RSLogix program.
- Verify that the correct program is loaded to the CPU
- Verify that Custom Generic functions are being used in the MSG blocks
- Check the communications tab in the MSG block. Make sure that the path is correct.
- Verify that the correct data types for the tags have been used.

Appendix B.2. Use the FieldServer RUI utility to check data transfer

The screenshot below shows the Read data transfer for the In_01 Tag.



The screenshot below shows the Write data transfer for the Out_01 Tag:

🔞 RSLogix 5000 - Project_01 [1756-L61]									- 0	×
File Edit View Search Logic Communications	Tools Window Help		🗪 Command Pro	ompt - ruinet					- 🗆 ×	
	<u>-</u> &	& 🔈 🖪 🖉			Da	ata Array				i l
Rem Run 📴 🔲 Run Mode	Path: AB_ETHIP-1\192.168.1.3	4\Backplane\0*	Data Array Data_Array_Na	ame		1 / 5 Out_0	5 91			
No Forces			Data_Hrray_Le	ength 			/5			
No Edits		SSV IOT	Data format: Bytes Per Ite	Float em: 4						
Redundancy Ag	Favorites & Bit & Timer/Con	nter Alnput/Output 🖌	Data age Oldest	31:07:05	10.785s 5:37.789s					
E-G Controller Project 01	Controller Tags - Project 01	(controller)	0: 98	76.5 0.	.0 0.0	0.0 0.	.0 0.0	0.0 0	.0	
Controller Tags		Show All	8: 16:	0.0 0.	.0 0.0 .0 0.0	0.0 0. 0.0 0.	.0 0.0 .0 0.0	0.0 0 0.0 0	.0 .0	F
Controller Fault Handler	Scope: Project_UT(controller V Sr		24: 32:	0.0 0. 0.0 0.	.0 0.0 .0 0.0	0.0 0. 0.0 0.	.0 0.0 .0 0.0	0.0 0 0.0 0	-0 -0	
E-G Tasks	I ag Name	Value •	40:	0.0 0.	0.0	0.0 0.	0 0.0	0.0 0	.0	Ĥ
🖻 🤕 MainTask	+ Local5:0	()	56:	0.0 0.	.0 0.0	0.0 0.		0.0 0	.0	
E- A MainProgram	T Local 5 C	()	64: 72:	0.0 0.	.0 0.0	0.0 0.	.0 0.0	0.0 0	-0	\square
MainRoutine	+ Msg In 01	()	1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 -							
Unscheduled Programs	- in_01	()	Vouet (at) New	st /Press //	New Local Sectors		IIIIN Mente	(Chata	(FRC)	
E Motion Groups	+ In_01.DA_Name	'In_01[0][75]'	KH>ex (D)ec (l	J>nsigned H	Fl <o>at B<y>ti</y></o>	e inary <	S>tring <m>o</m>	dify <+->	Offset	
Trends	+ In_01.Data_In	{}								4
🖃 📇 Data Types	+ Msg_Out_01	{}	()		MESSAGE					
🖃 📆 User-Defined	Out_01	{}	{}		SlotServer_Data_Ou	t		1		
SlotServer_Data_In	+ Out_01.DA_Name	'Out_01[0][75]'	()		STRING			-		
E Strings	- Uut_UI.Data_Uut	{}	{} Flo	at	REAL[75]					
Predefined	Out_01_Data_Out[0]	9876.5	FIO	at	REAL					
Module-Defined	-Out 01 Data Out[2]	0.0	Flo	at	REAL			1		
[5] 1756-MODULE SlotServer	-Out 01.Data Out[3]	0.0	Flo	at	REAL					
	Out_01.Data_Out[4]	0.0	Flo	at	REAL					-
1	Monitor Tags / Edit Tags /	·		•					•	
	📕 MainProgram - MainRoutine									X
		8 8 V V	ab l							
			<u></u>							
										_
	Msg_Trigger					1	M.	sc	-tho-	^
	·						Message Control	Msg_In_01		
									-(ER)	
										=
	Msg_Trigger					_	MS	;c	_	
	1					!	Type - CIP Generic	Mar Out 01		
							wessage control	msg_out_or		_
						L				
	(End)									
	· ·								1	
										~
	MainRoutine /			•						١Ī
Enter a tag value									2	_

Appendix C. Using LonMaker to Commission a SlotServer

- Ensure that the SlotServer and the LonMaker PC are on the same LonWorks network.
- Open the existing Network in LonMaker, or create a new Network
- Click on "Create New Network" and follow the network wizard, making the following selections:
 - o Network Interface: Choose Network Attached
 - Management Mode: Choose Onnet unless you are working offline
 - o Register Plug-ins required. None
- Once Visio is open with the Network showing, drag a new device onto the drawing from the toolbox.
- Follow the Device Network, making the following selections
 - Enter Device Name: Choose commission device
 - Specify Device Template: Choose upload from device
 - Specify Device Channel: Choose Auto Detect
 - Specify Device Properties: Leave as is (Ping is optional)
 - Identify Device: Choose service pin
 - Device Application Image: Leave unchecked
 - o Initial State: Online
- Press the service pin on the SlotServer when asked to do so, and the SlotServer will be commissioned.
- Drag a new function block onto the drawing from the toolbox. Give the function block a name and ensure that it is allocated to the SlotServer device.
- Once the function block is on the drawing, you can drag input and output variables onto the function block. When you do this, LonMaker will show you the variables available for binding. Click on the variables you require (or use the select all option), and they will be commissioned onto the function block.
- You are now ready to connect these variables to other devices by dragging connections from the toolbox and connecting the variables.

Appendix D. Network Variables List – LonWorks Open Interface Profile

Appendix D.1. LonWorks Network Variables Summary

The SlotServer LonWorks Open Interface Profile contains 50 Functional Blocks containing a total of 2,600 Network Variables of different SNVT Types.

Input Data Function Blocks: In_01 .. In_25

Output Data Function Blocks: Out_01 .. Out_25

Each Function Block contains the following Network Variables:

NV name	
(i) for Input	LonWorks SNVT Type
(o) for Output	
nv(i)(o)Amp1	SNVT_amp
nv(i)(o)Amp2	SNVT_amp
nv(i)(o)Chlr1	SNVT_chlr_status
nv(i)(o)Count1	SNVT_count_f
nv(i)(o)Count2	SNVT_count_f
nv(i)(o)CtInc1	SNVT_count_inc_f
nv(i)(o)CtInc2	SNVT_count_inc_f
nv(i)(o)Flow1	SNVT_flow_f
nv(i)(o)Freq1	SNVT_freq_hz
nv(i)(o)HVMod1	SNVT_hvac_mode
nv(i)(o)HVSts1	SNVT_hvac_status
nv(i)(o)Lev1	SNVT_lev_cont
nv(i)(o)Obj1	SNVT_obj_status
nv(i)(o)Occ1	SNVT_occupancy
nv(i)(o)Occ2	SNVT_occupancy
nv(i)(o)Perc1	SNVT_lev_percent
nv(i)(o)Perc2	SNVT_lev_percent
nv(i)(o)Perc3	SNVT_lev_percent
nv(i)(o)Perc4	SNVT_lev_percent
nv(i)(o)Perc5	SNVT_lev_percent
nv(i)(o)Perc6	SNVT_lev_percent
nv(i)(o)Perc7	SNVT_lev_percent
nv(i)(o)Perc8	SNVT_lev_percent
nv(i)(o)PPM1	SNVT_ppm
nv(i)(o)Press1	SNVT_press
nv(i)(o)Press2	SNVT_press
nv(i)(o)Setpt1	SNVT_temp_setpt
nv(i)(o)State1	SNVT_state
nv(i)(o)State2	SNVT_state
nv(i)(o)State3	SNVT_state
nv(i)(o)State4	SNVT_state

NV name	
(i) for Input	LonWorks SNVT Type
(o) for Output	
nv(i)(o)Sw1	SNVT_switch
nv(i)(o)Sw2	SNVT_switch
nv(i)(o)Sw3	SNVT_switch
nv(i)(o)Sw4	SNVT_switch
nv(i)(o)Sw5	SNVT_switch
nv(i)(o)Sw6	SNVT_switch
nv(i)(o)Sw7	SNVT_switch
nv(i)(o)Sw8	SNVT_switch
nv(i)(o)Temp1	SNVT_temp_p
nv(i)(o)Temp2	SNVT_temp_p
nv(i)(o)Temp3	SNVT_temp_p
nv(i)(o)Temp4	SNVT_temp_p
nv(i)(o)Temp5	SNVT_temp_p
nv(i)(o)Temp6	SNVT_temp_p
nv(i)(o)Temp7	SNVT_temp_p
nv(i)(o)Temp8	SNVT_temp_p
nv(i)(o)TodEv1	SNVT_tod_event
nv(i)(o)VItAC1	SNVT_volt_ac
nv(i)(o)VItAC2	SNVT_volt_ac
nv(i)(o)Volt1	SNVT_volt
nv(i)(o)Volt2	SNVT_volt

THIS PAGE INTENTIONALLY LEFT BLANK