TeSys[™] island

Quick Start Guide for EtherNet/IP™ Applications

Instruction Bulletin

This instruction bulletin describes how to quickly integrate TeSys island into the Rockwell Software® Studio 5000® environment.

8536IB1906EN Release date 06/2019





Legal Information

The Schneider Electric brand and any trademarks of Schneider Electric SE and its subsidiaries referred to in this guide are the property of Schneider Electric SE or its subsidiaries. All other brands may be trademarks of their respective owners.

This guide and its content are protected under applicable copyright laws and furnished for informational use only. No part of this guide may be reproduced or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), for any purpose, without the prior written permission of Schneider Electric.

Schneider Electric does not grant any right or license for commercial use of the guide or its content, except for a non-exclusive and personal license to consult it on an "as is" basis. Schneider Electric products and equipment should be installed, operated, serviced, and maintained only by qualified personnel.

As standards, specifications, and designs change from time to time, information contained in this guide may be subject to change without notice.

To the extent permitted by applicable law, no responsibility or liability is assumed by Schneider Electric and its subsidiaries for any errors or omissions in the informational content of this material or consequences arising out of or resulting from the use of the information contained herein.

Schneider Electric, SoMove, and TeSys are trademarks and the property of Schneider Electric SE, its subsidiaries, and affiliated companies. All other trademarks are the property of their respective owners.

Table of Contents

About the Book	5
Scope	5
Validity Note	5
Related Documentation	6
Methodology	7
Obtain the L5X Files	7
Integration Procedure	8
Add an Ethernet Bridge	8
Import the TeSys™ island Module	9
Import the Subroutine	10
Import the Add-on Instructions	11
Create Instances of the AOIs	11
Calling Acyclic Data	13
Data Access Example	13
Accessing Data via AOI	13
Accessing Data via the Acyclic Buffer	15
Frequently Asked Questions (FAQs)	16

Hazard Categories and Special Symbols

Read these instructions carefully and look at the equipment to become familiar with the device before trying to install, operate, service, or maintain it. The following special messages may appear throughout this bulletin or on the equipment to warn of hazards or to call attention to information that clarifies or simplifies a procedure.



The addition of either symbol to a "Danger" or "Warning" safety label indicates that an electrical hazard exists which will result in personal injury if the instructions are not followed.



This is the safety alert symbol. It is used to alert you to personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

A DANGER

DANGER indicates a hazardous situation which, if not avoided, will result in death or serious injury.

WARNING indicates a hazardous situation which, if not avoided, **could result** in death or serious injury.

CAUTION indicates a hazardous situation which, if not avoided, **could result** in minor or moderate injury.

NOTICE

NOTICE is used to address practices not related to physical injury.

NOTE: Provides additional information to clarify or simplify a procedure.

Please Note

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction, installation, and operation of electrical equipment and has received safety training to recognize and avoid the hazards involved.

About the Book

Scope

This user guide provides instructions for configuring a TeSys[™] island device within the Rockwell Software[®] Studio 5000[®] environment. For further information regarding the TeSys island devices, refer to *Related Documentation, page 6*.

Library Compatibility: L5X files exported from SoMove software are compatible with Studio 5000 major version 30 or higher. Refer to the Product Compatibility & Downloads page on the Rockwell Automation website to verify that the PLC firmware is compatible with the Studio 5000 version being used. As of the time of publication, this information can be found at *https://compatibility.rockwellautomation.com*.

Validity Note

This instruction bulletin is valid for all TeSys[™] island configurations. The availability of some functions described in this bulletin depends on the communication protocol used and the physical modules installed on the island.

For product compliance with environmental directives such as RoHS, REACH, PEP, and EOLI, go to *www.se.com/green-premium*.

For technical characteristics of the physical modules described in this bulletin, go to *www.se.com*.

The technical characteristics presented in this bulletin should be the same as those that appear online. We may revise content over time to improve clarity and accuracy. If you see a difference between the information contained in this bulletin and online information, use the online information.

Related Documentation

Table 1 - Related Documen	tation
---------------------------	--------

Document Title	Description Document Number		
TeSys™ island System Guide	Introduces and describes the main functions of TeSys island	8536IB1901	
TeSys™ island Installation Guide	Describes the mechanical installation, wiring, and commissioning of TeSys island	8536IB1902	
TeSys™ island Operating Guide	Describes how to operate and 8536IB1903 maintain TeSys island		
TeSys™ island Functional Safety Guide	Describes the Functional Safety 8536IB1904 features of TeSys island		
TeSys™ island Third Party Function Block Guide	Contains the information needed to create function blocks for third party hardware 8536IB1905		
TeSys™ island EtherNet/IP™ Function Block Library Guide	ctionDescribes the TeSys island library used in the Rockwell Software® Studio 5000® environment8536IB197		
TeSys™ island EtherNet/IP™ Quick Start Guide	Describes how to quickly integrate TeSys island into the Rockwell Software Studio 5000 environment8536IB1906		
TeSys™ island DTM Online Help Guide	Describes how to install and use various functions of TeSys island configuration software and how to configure the parameters of TeSys island	8536IB1907	
TeSys™ island Product Environmental Profile	Image: Describes constituent materials, recyclability potential, and environmental impact information for the TeSys island. ENV-PEP1904009		
TeSys™ island Product End of Life Instructions	Contains end of life instructions for the TeSys island	ENVEO- LI1904009	
TeSys™ island Instruction Sheet, Bus Coupler	Describes how to install the TeSys island bus coupler	MFR44097	
TeSys™ island Instruction Sheet, Starters and Power Interface Modules, Size 1 and 2	Describes how to install size 1 and 2 TeSys island starters and power interface modules	MFR77070	
TeSys™ island Instruction Sheet, Starters and Power Interface Modules, Size 3	Describes how to install size 3 TeSys island starters and power interface modules	MFR77085	
TeSys™ island Instruction Sheet: Input/ Output Modules	Describes how to install the TeSys island analog and digital I/O modules	bes how to install the TeSys MFR44099 analog and digital I/O es	
TeSys™ island Instruction Sheet: SIL Interface and Voltage Interface Modules	Describes how to install the TeSys island voltage interface modules and SIL interface modules	MFR44100	

Methodology

A TeSys[™] island device can be integrated into the Rockwell Software[®] Studio 5000[®] environment using the L5X import feature. Based on the TeSys island configuration, SoMove[™] software provides L5X files that can be imported into the Studio 5000 environment, exposing the device data via add-on instructions.

NOTE: Due to data referencing, the L5X files must be imported in the order listed below. Follow the instructions provided in this manual.

The following files are generated by SoMove software:

- 1. Generic Ethernet Module: This file contains a Generic Ethernet AOP representing the TeSys island configuration. It includes comments in the Controller Tag section indicating what each piece of data is referencing. It uses the following naming convention: {*DeviceName*}_*Module.L5X*.
- 2. Explicit Messages (if acyclic data exists): This file contains the following:
 - subroutines containing explicit messages for calling acyclic data
 - · a data buffer used to hold the acyclic data before and after transmission
 - It uses the following naming convention: {DeviceName}_Acyclic.L5X.

NOTE: If a TeSys island configuration does not have acyclic data, this file is not generated.

 Add-on Instructions (AOI): This file contains the AOIs used for generating function blocks. It must be imported last because the AOIs' reference tags are contained in the other two files. It uses the following naming convention: {DeviceName}_Aoi.L5X.

NOTE: If you alter the naming convention for a TeSys island or for pre-existing avatars, then when you integrate configuration changes into a project, all the imported Studio 5000 software entities (tags, AOIs, and generic Ethernet module) must be deleted and the import process repeated for the new configuration. See *Frequently Asked Questions (FAQs), page 16*.

Obtain the L5X Files

Configure the island in the TeSys[™] island DTM according to the *TeSys[™] island Operating Guide*, 8536IB1903. Then, export the L5X files from the TeSys island DTM according to the following procedure.

- 1. In the TeSys island DTM, open the TeSys island project you wish to export.
- 2. From the drop-down menu, click Device.
- 3. Select Export > EDS to L5X File Format.
- 4. Click Save.

The file is saved as a zip file in the following format: island_name.zip.

5. A notification appears, saying that the L5X files have been created. Click OK.

Integration Procedure

Prerequisites

The integration procedure assumes that you have done the following:

- Set up a TeSys™ island device using SoMove™ software
- Obtained the L5X files needed for integration with the Rockwell Software[®] environment

Example Items

The TeSys island configuration used in this example includes the following items, as shown in the figure below.

- Digital I/O module named TeSysIslandDevice
- Motor Two Directions avatar



Add an Ethernet Bridge

To add an Ethernet bridge:

- 1. Launch the Studio 5000® software.
- 2. Click Create > New Project.
 - A new project opens.
- 3. Select your PLC from the list of available PLCs.
- 4. Enter a Name for the PLC. Click Next.
- 5. Make any changes necessary for the PLC. Click Finish. Your PLC module is added to the I/O Configuration tree.
- 6. Add an Ethernet bridge to its backplane.



Import the TeSys[™] island Module

Import the TeSys island module to the Ethernet bridge. This file uses the following naming convention: {*DeviceName*}_*Module.L5X*.

🖻 🚔 I/O Configuration			
🖨 🛲 1756 Backplane	e, 17	56-A7	
🔁 [0] 1756-L74	Tes	stDevice	
🖻 🖞 [1] 1756-EN	2T B	Bridge	
	IJ	New Module	
		Import Module	
		Discover Modules	
	ß	Paste	Ctrl+V
		Print	•

After import, the device shows up in the I/O configuration tree and the Controller Tags section of the project.

 ■ Controller Tags ■ Controller Tags ■ Controller Fault Handler ■ Power-Up Handler ■ TeSysIslandDevic ((-			
			Name 📰 🗅	Val 🗧	For +	Style	Data Type	Description
TeSysIslandDevi 0 Decimal SINT AvSystem1: TeSysIslandDevi 0 Decimal SINT AvSystem1: TeSysIslandDevi 0 Decimal SINT AvSystem1: TeSysIslandDevi 0 Decimal SINT AvDigitalO: TeSysIslandDevi 0 Decimal SINT AvMotorTwoDirection: TeSysIslandDevi 0 Decimal SINT AvMotorTwoDirection: AvgIRMS Add-On Instructions TeSysIslandDevi 0 Decimal SINT AvMotorTwoDirection: AvgIRMS TeSysIslandDevi 0 Decimal SINT AvMotorTwoDirection: AvgIRMS TeSysIslandDevice: 0 ((ABETHE.) TeSysIslandDevice: 0 ((ABETHE.) ToSe Backplane, 1756-L74 TestDevice TeSysIslandDevice: 0 ((ABETHE.) TeSysIslandDevice: 0 ((ABETHE.) TeSysIslandDevice: 0 ((ABETHE.) TeSpENTER TeSpENTER TeSpENTER TeSPENTER TeSPENTER TeSPENTER TeSPENTER TeSPENTER TeSPENTER TeSPENTER TeSPE	- 🖉 Controller Tags		TeSysIslandDevice:	{	{		AB:ETHE	
Image: Sint Sint AvSystem: Image: Sint AvdSystem: Image: Sint AvdSystem: <t< td=""><td> 🗀 Controller Fault Handler</td><td></td><td>- TeSysIslandDevice</td><td>{</td><td>{</td><td>Decimal :</td><td>SINT[9]</td><td></td></t<>	🗀 Controller Fault Handler		- TeSysIslandDevice	{	{	Decimal :	SINT[9]	
Image: Single	Power-Up Handler		+ TeSysIslandDevi	0		Decimal	SINT	AvSystem1:
Image: Sint Sint Sint Sint Sint AvDigital C Image: Sint Sint Sint Sint Sint Avditor Two Direction: Image: Sint Sint Sint Sint Sint Avditor Two Direction: Image: Sint Sint Sint Sint Sint Avditor Two Direction: Image: Sint Sint Sint Sint Sint Avditor Two Direction: Image: Sint Sint Sint Sint Sint Avditor Two Direction: Image: Sint Sint Sint Sint Sint Avditor Two Direction: Image: Sint Sint Sint Sint Sint Sint Avditor Two Direction: Image: Sint Sint Sint Sint Sint Avditor Two Direction: AvgIRMS Image: Sint Sint Sint Sint Sint Avditor Two Direction: AvgIRMS Image: Sint Sint Sint Sint Sint Avditor Two Direction: AvgIRMS Image: Sint Sint Sint Sint Avditor Two Direction: AvgIRMS Image: Sint Sint Sint Sint Avditor Two Direction: AvgIRMS Image: Sint Sint Sint Sint Sint Avditor Two Direction: AvgIRMS Image: Sint Sint Sint Sint Sint Sint Avditor Two Direction: AvgIRMS Image: Sint Sint Sint Sint Sint Sint Sint Sint	🖻 🚔 Tasks		+ TeSysIslandDevi	0		Decimal	SINT	AvSystem1:
Image: Sint Add: Sint Add	🖨 🚭 MainTask		+ TeSysIslandDevi	0		Decimal	SINT	AvDigitallO:
Inscheduled Image: Sint Add-On Instructions Image: Sint Add-On Instructions Image: Sint Add-On Instructions Image: Sint Add-On Instructions Image: Sint Add-On-Defined <	🕀 🖷 MainProgram		+ TeSvslslandDevi	0		Decimal	SINT	AvMotorTwoDirection:
Motion Groups Motion Mo	Unscheduled	⊫	+ TeSysIslandDevi	0		Decimal	SINT	AvMotorTwoDirection:
Add-On Instructions Add-On-Defined AB:ETHE. TreSysIslandDevic. 0 Decimal SINT AvMotorTwoDirection: AvgIRMS TeSysIslandDevic. 0 Decimal SINT AvMotorTwoDirection: AvgIRMS TeSysIslandDevice: 0 ((AB:ETHE.) TeSysIslandDevice: 0 ((AB:ETHE.) TeSysIslandDevice: 0 ((AB:ETHE.) Trends Trends Trends Trof5e Traft TestPovice Tof5e Backplane, 1756-L74 TestDevice Tof5e Backplane, 1756-L74 TestDevice Tof5e TestPI Bridge TestPovice TestPostIstandDevice TestPostIstandDevice	Motion Groups	⊫	+ TeSysIslandDevi	0		Decimal	SINT	AvMotorTwoDirection: AvaIRMS
Add-On Instructions Add-On Instructions Add-On Detined SINT AvMotorTwoDirection: AvgIRMS SINT AvMotorTwoDirection: AvgIRMS + TeSysIslandDevice: 0 ((ABETHE. + TeSysIslandDevice: 0 (((ABETHE. + TeSysIslandDevice: 0 (((ABETHE. + TeSysIslandDevi	Ungrouped Axes	⊫	+ TeSysIslandDevi	0		Decimal	SINT	AvMotorTwoDirection: AvalRMS
□ Data Types □ Topolation True and the product of	Add-On Instructions	⊩	+ TeSysIslandDevi	0		Decimal	SINT	AvMotorTwoDirection: AvgIRMS
Image: Strings Image: Strings Image: Strings Image: St	🖻 🖼 Data Types	⊩	+ TeSysIslandDevi	0		Decimal	SINT	AvMotorTwoDirection: AvaIRMS
Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings Image: Strings <td< td=""><td>User-Defined</td><td>⊩</td><td>Ta Suelelend Device: 0</td><td></td><td></td><td>Decimal</td><td></td><td>Avmotor r wobliection. Avgir (mo</td></td<>	User-Defined	⊩	Ta Suelelend Device: 0			Decimal		Avmotor r wobliection. Avgir (mo
Image: Model-Defined Image: Model-Defined Image: Modele-Defined Image: Modele-Defined Image: Modele-Defined Image: Modele-Definede	- 🖼 Strings	⊩	TesysisiandDevice.0	1	1	4	AD.ETHE	
	- 🦛 Add-On-Defined	⊫	± TeSysIslandDevice:C	{	{		AB:ETHE	
B	🕀 🖼 Predefined							
Trends Trends Its Logical Model If () Configuration If () 1756-L74 TestDevice If () 1756-EN2T Bridge If () 1756-EN2T Bridge If () 1756-EN2T Bridge If () IT56-EN2T Bridge	🗄 🖼 Module-Defined							
→ In Logical Model → In Zos Backplane, 1756-A7 → In [0] 1756-L74 TestDevice → In [0] 1756-EN2T Bridge → Ethernet → In In In Zos In Internet → In Internet → In Internet → In Internet → Internet	Trends							
	-h. Logical Model							
	⊨ l/O Configuration							
- 節 [0] 1756-L74 TestDevice □ 鄧 [1] 1756-EN2T Bridge □ 鄙 Ethernet □ 1756-EN2T Bridge □ 1756-EN2T Bridge □ ETHERNET-MODULE TeSysIslandDevice	1756 Backplane, 1756-A7							
	1756-L74 TestDevice							
	🖃 🖞 [1] 1756-EN2T Bridge							
ITS6-EN2T Bridge ItERNET-MODULE TeSysIslandDevice	Ethernet							
ETHERNET-MODULE TeSysIslandDevice	1756-EN2T Bridge							
	FTHERNET-MODULE TeSvsIslandDevice							
	s entenater mobile resystandberice							

Import the Subroutine

Import the subroutine L5X file containing the acyclic data, if present. This file uses the following naming convention: {*DeviceName*}_*Acyclic.L5X*.

🚊 🔤 Tasks					
🖨 🗟 MainTask					
🖨 😂 MainProgram					
- 🖉 Parameters		Add			New Routine
🛄 MainRoutir					
Unscheduled	*	Cut	Ctrl+X	Ø	New Local Tag Ctrl+W
🚔 😂 Motion Groups	₿ ₽	Сору	Ctrl+C		New Parameter
Ungrouped Axes	B	Paste	Ctrl+V		1 (D)
Add-On Instructions		Delete	Del		Import Routine
🖻 📾 Data Types					
User-Defined		Verify		L .	
- Strings		Cross Reference	Ctrl+E	L .	
Add-On-Defined				1	
Predefined		Browse Logic	Ctrl+L	L .	
H R Module-Defined		Find in Logical Organizer		L .	
Irenas		Online Edite		1	
Logical Model					
in Configuration		Print	+	L .	
一面 [0] 1756-174 T					
□ ¶ [1] 1756-FN21		Export Program		L .	
는 윦 Ethernet		Properties		1	
1756-EN		Bridge	AITTEILEI		
I FTHERN	FT-I	MODULE TeSysIslandDevice	`		
D ETTERNA		nobole respondition	*		

After import:

- · the subroutines are visible in the Tasks tree
- the acyclic data buffer is visible in the Controller Tags section

i ← Tasks i ← AminTask i ← MainTask i ← MainProgram i ← Ø Parameters and Local Tags	Controller TestDevice	Name	<u>=</u> ∎ ∆ fer	Val + {	For •	Style	Data Type MESSAGE TeSysIsI
MainRoutine AvMotorTwoDirection AvAotorTwoDirection AvAotorTwoDirection AvAotorTwoDirection Device2 Device3 Device4 Device5 Device6	Subroutines	Acyclic Data Buffer					

For more information on accessing the device data, see *Data Access Example, page 13*.

Import the Add-on Instructions

Import the add-on instructions (AOIs) as shown below. This file uses the following naming conventions: {*DeviceName*}_*Module.L5X*

🖨 🖾 Motion Groups			
Ungrouped Axes			
Add-On Instructions			
🖨 🚔 Data Types		New Add-On Instruction	
🖳 🖼 User-Defined		Import Add-On Instruction	
📲 Strings			
📲 Add-On-Defined	X	Cut	Ctrl+X
🕀 🖼 Predefined	Ē	Сору	Ctrl+C
🗄 🖼 Module-Defined	B	Paste	Ctrl+V
Trends		Paste With Configuration	Ctrl+Shift+V
🗄 🗽 Logical Model			
🖨 📾 I/O Configuration		Print	+
🖻 📟 1756 Backplane, 1	756	-A/	

After import, the AOIs are visible in the project tree.

🚔 📾 Add-On Instructions
🖨 🛅 AvDigitallO
🖉 Parameters and Local Tags
🛄 Logic
AvMotorTwoDirection
Parameters and Local Tags
Logic
AvMotorTwoDirection_Control
Parameters and Local Tags
🗈 Logic

Each AOI contains either cyclic or acyclic data, which is indicated by the following naming convention:

- Cyclic data contains only the name of the avatar.
- Acyclic data contains the name of the avatar followed by an underscore (_) and the name of the acyclic data object.

Create Instances of the AOIs

When creating an instance of an AOI, you must reference the module and the acyclic data buffer. Depending on the data present in the AOI (cyclic or acyclic), there are one or two tags to reference.

1. Enter the Main Routine by double-clicking it within the Tasks tree.

Controller Organizer 🝷 🃮	×	Γ	田 雪 昭	1 E B	abed a	b ob.	🔻	<05>		
Controller Tags	^	F							-	
- Controller Fault Handler										
Power-Up Handler										
🖨 📾 Tasks		F		-						
🚊 🚭 MainTask		L	(End)							
🖨 🚭 MainProgram		L								
Parameters and Local Tags		L								
- Di MainRoutine		L								
AvMotorTwoDirection		L								
AvSystem1		L								

2. Add an AOI from the add-on list.



3. Fill in the reference tags for each In/Out parameter. These tags are in the Controller Tags list with names matching the In/Out Parameters.

AvSystem1		?			0
Mylsland_Input ?		~			
Mylsland_Output	-		Ctrl\/ltaE	Ictn \	1
ResetVItgFlctnSt	Enter Name Filter	~ S	how: AB:ETHEF	RNET_MODULE_SINT_1	4E ~
ResetMaxUnbala	Name	B Data Type	Usage	Description	
ResetMaxVRMS	+-Mylsland:I	AB:ETHERNET.	. <controller></controller>		
ResetComErrorC					
ResetMinorEvtCi					
ResetAlarmCntr					
PacatSystem					~

4. Create an instance of the AOI and name it.

AvSystem		
AvSystem1	New Tag	IctnState
Mylsland Output	% Cut Instruction	Ctrl+X
ResetVItoFictnStatus	<u>Copy Instruction</u>	Ctrl+C
ResetMaxUnbalanceVit	Paste	Ctrl+V
ResetMaxVRMS	Delete Instruction	Del
ResetComErrorCntr	Add Ladder Element	Alt+Ins
ResetMinorEvtCntr	Edit Main Operand Description	Ctrl+D
ResetAlarmCntr		₽)
ResetSystem	Save Instruction Defaults	

The AOI is now fully configured and ready for use.

	AvSystem1
AvSystem1	AvSystemInstance
Mylsland_Inpu	MyIslandDeviceIn
MyIsland_Outp	ut MylslandDeviceOut
ResetVItgFlctn	Status 0≮
ResetMaxUnb	llanceVltg 0+
ResetMaxVRM	S 0+
ResetComErro	rCntr 0+
ResetMinorEv	Cntr 0+
ResetAlarmCn	r 0∻
ResetSystem	0∻

Calling Acyclic Data

When calling acyclic data, you must add a Jump To Subroutine operation to the Main Routine. The subroutine needed for the target AOI matches the associated Avatar.



This action pushes or pulls data, depending on the type of data the explicit message is accessing.

- 1. Open the subroutine.
- 2. Toggle the Examine If Open bit on the rung containing the message associated with the target AOI.



Data Access Example

The following section provides an example of how to access data on the TeSys Island. These example AOIs:

- · were added to the project using the same process described above
- are the for cyclic system data (AvSystem1), acyclic system diagnostic (AvSystem1_Diagnostic), and acyclic system asset management (AvSystem1_AssetManagement)

Accessing Data via AOI

After setting up the AOIs you plan to use:

- 1. Download the program.
- 2. Set the PLC to Run mode.



If the device is properly set up, and does not have any trips or other events, the cyclic system AOI should indicate that the system is operational.

AvSystem1		1
AvSystem1 AvSystemInstar Mylsland_Input MylslandDevice0 Mylsland_output MylslandDevice0 ResetVitgFlctnStatus ResetMaxURbalanceVitg ResetComErrorCntr ResetMinorEvtCntr ResetAlarmCntr ResetSystem	nce eln Out O+ O+ O+ O+ O+ O+ O+ O+ O+ O+	- CtrlVitgFictn)- - Preoperational - Operational - TestMode - MinorEvt)- - ForceMode - DegradedMode

To view acyclic data, the appropriate explicit message must be accessed. See *Calling Acyclic Data, page 13*. As shown here, the device currently has one Minor Event logged via the System Minor Event Counter within the acyclic system diagnostic AOI.

AvSystem1	_Diagnostic
AvSystem1_Diagnostic Mylsland_AcyclicBuffer ComErrorCntr AlarmsCntr	AvSystemDiagnostic MylslandDevlicBuffer 0 ← 0 ←
MinorEvtCntr	1+
MinorEvtRegister1_Y	2019*
MinorEvtRegister1_M	6*
MinorEvtRegister1_D	19*
MinorEvtRegister1_n	4*
MinorEvtRegister1_min	2*
MinorEvtRegister1_sec	22€
MinorEvtRegister1_cs	22*
MinorEvtRegister1 Evtlo	dentifier 0 ←

This value can be reset by flipping the Minor Event Reset bit within the cyclic system AOI.

AvSystem1	
AvSystem1 AvSystem	Instance
Mylsland_Output MylslandDe	eviceOut
ResetMaxUnbalanceVitg	0+
ResetMaxVRMS ResetComErrorCntr	0 ← 0 ←
ResetMinorEvtCntr ResetAlarmCntr	1÷ 0€
ResetSystem	0+

After a refresh of the acyclic system diagnostic data (see *Calling Acyclic Data, page 13*), the counter returns to 0.

AvSystem1	I_Diagnostic
AvSystem1_Diagnostic Mylsland_AcyclicBuffer	AvSystemDiagnostic
ComErrorCntr	0*
AlarmsCntr	0←
MinorEvtCntr	0+
MinorEvtRegister1_Y	2019 *
MinorEvtRegister1_M	6*
MinorEvtRegister1_D	19≮
MinorEvtRegister1_h	4≮
MinorEvtRegister1_min	2*
MinorEvtRegister1_sec	22*
MinorEvtRegister1_cs	22 +
MinorEvtRegister1 Evtlo	dentifier 0+

Accessing Data via the Acyclic Buffer

AOIs are capable of exposing only SINT, INT, DINT, REAL, and BOOL data types as Input/Output parameters. Due to this constraint, STRING registers are placed within the acyclic data buffer and can be accessed there.

After you toggle the bit to access the acyclic system asset management data (see *Calling Acyclic Data, page 13*), the STRING data does not appear in the AOI. Instead, it appears within the acyclic data buffer. This buffer can be found in the Controller Tags list with naming convention {*DeviceName*}_AcyclicBuffer.

Controller Load	^	Name == A	Value 🔶	Force Mask +	Style	Data Type
Controller Tags		+ MyIsland:C	{}	{}		AB:ETHERNET
Controller Fault Handler		+ MyIsland:	{}	{}		AB:ETHERNET
Power-Up Handler		+ MyIsland:O	{}	{}		AB:ETHERNET
Tasks		- MyIsland_AcyclicBuffer	{}	{}		MyIsland_Acyclic
Main lask		- Mylsland_AcyclicBuffer.AvSystem1_VendorName	'Schneider Electric'	{}		MyIsland_20
		+ MyIsland_AcyclicBuffer Av System 1_VendorName LEN	18		Decimal	DINT
MainPoutine		MyIsland_AcyclicBuffer.AvSystem1_VendorName.DA	{}	{}	ASCII	SINT[20]
AvMotorTwoSpeeds		MyIsland_AcyclicBuffer.AvSystem1_ProductCode	'TPRBCEIP'	{}		MyIsland_32
AvSwitch		MyIsland_AcyclicBuffer.AvSystem1_AppRevision	'00.0300'	{}		MyIsland_7
AvSystem1		+ MyIsland_AcyclicBuffer AvSystem1_VendorURL	'www.schneider-electric	{}		MyIsland_64
CombinedSystemOutput		MyIsland_AcyclicBuffer.AvSystem1_ProductName		{}		MyIsland_32

The remaining data is accessible within the AOI.

AvSystem1_Asset	/lanagement
AvSystem1_AssetManagem Mylsland AcyclicBuffer My	AvSystemAsset
MacAddress XX	255 +
MacAddress_YY	255 +
MacAddress_ZZ	255 +
MacAddress_UU	255 🗲
MacAddress_VV	255 🕈
MacAddress_WW	255 🕈
TimeDeviceOn	0 ←
DeviceEvtCntr	0 ←

Frequently Asked Questions (FAQs)

Where can STRING data be accessed?

Due to AOI constraints, complex data types cannot be exposed as Input/Output parameters. Refer to *Accessing Data via the Acyclic Buffer, page 15* for instructions on accessing STRING data.

What are the lines attached to the right side of the AOI?

All the data exposed by an AOI exists within the main body, except for data exposed with Output parameters of type BOOL. This data exists on the right side of the AOI, as outlined with the red box in the screenshot below.

AvSystem1		1
AvSystem1 MyIsland_Input MyIsland_Output ResetVitgFlctnStatus ResetMaxUnbalanceVitg ResetMaxVRMS ResetMinorEvtCntr ResetAlarmCntr	tance viceln eOut 0+ 0+ 0+ 0+ 0+ 0+	CtrlVitgFlctn) Preoperational) Operational) TestMode) MinorEvt) ForceMode) OperadedMode

How are TeSys™ Island configuration changes integrated into a project?

- If the naming convention has been altered for a TeSys island or pre-existing Avatars, all the imported Studio 5000[®] software entities (tags, AOIs, and generic Ethernet module) must be deleted and the import process repeated for the new configuration.
- If the naming convention has not changed for pre-existing entities, the import process can be repeated without deleting previously imported entities. Change the import Operation from *Use Existing* to *Overwrite*, as shown in the image below.

∑ Find: Find Within: Final Name	✓ A A → Find/Replace	
port Content:	Configure Tag References	
MainProgram	Import Name Operation 🕞 Final Name 🛆 😭 Usage Al	ias For \land
References	Image: AvMotorTwoDirectip Use Drow AvMotorTwoD Image: Local Image: AvMotorTwoDirectip Overwrite Moto TwoD Image: Local	
Data Types Other Components	AvMotorTwoDirectip Use Existing Moto TwoD Local	
Errors/Alarms	J AvSystem_AssetMb Lise Fxist AvSystem_Ass Image: Construction of the second	
	AvSystem_Energy Use Exist AvSystem_En Local	
	CombinedSystemO Use Exist CombinedSys Local	
	Device2_AssetMan Use Exist Device2_Asse Local	
	Device3_AssetMan Use Exist Device3_Asse Local	
	Device4_AssetMan Use Exist Device4_Asse Local	~

Why do 32-bit unsigned integers have a maximum displayed value of 2,147,483,647?

Rockwell Software[®] Studio 5000 software only handles signed integers. For this reason, the maximum positive value that can be displayed for 32-bit unsigned integers is 2,147,483,647. To enforce this, logic exists within AOIs to max out UDINT registers if the sign bit is used. For these TeSys island registers, a flag exists as an exposed parameter to indicate overflow. These flags are of type BOOL with the naming convention *{TagName}_O*.

Schneider Electric 800 Federal Street Andover, MA 01810 USA

https://www.schneider-electric.com/en/work/support/

www.schneider-electric.us

As standards, specifications, and design change from time to time, please ask for confirmation of the information given in this publication.

© 2019 - Schneider Electric. All rights reserved.

8536IB1906EN