DDS Blockset

Reference

MATLAB&SIMULINK®



R

R2021**a**

How to Contact MathWorks



Latest news:

Phone:

www.mathworks.com

Sales and services: www.mathworks.com/sales_and_services

User community: www.mathworks.com/matlabcentral

Technical support: www.mathworks.com/support/contact_us



 \searrow

508-647-7000

The MathWorks, Inc. 1 Apple Hill Drive Natick, MA 01760-2098

DDS Blockset Reference

© COPYRIGHT 2021 by The MathWorks, Inc.

Online only

The software described in this document is furnished under a license agreement. The software may be used or copied only under the terms of the license agreement. No part of this manual may be photocopied or reproduced in any form without prior written consent from The MathWorks, Inc.

FEDERAL ACQUISITION: This provision applies to all acquisitions of the Program and Documentation by, for, or through the federal government of the United States. By accepting delivery of the Program or Documentation, the government hereby agrees that this software or documentation qualifies as commercial computer software or commercial computer software documentation as such terms are used or defined in FAR 12.212, DFARS Part 227.72, and DFARS 252.227-7014. Accordingly, the terms and conditions of this Agreement and only those rights specified in this Agreement, shall pertain to and govern the use, modification, reproduction, release, performance, display, and disclosure of the Program and Documentation by the federal government (or other entity acquiring for or through the federal government) and shall supersede any conflicting contractual terms or conditions. If this License fails to meet the government's needs or is inconsistent in any respect with federal procurement law, the government agrees to return the Program and Documentation, unused, to The MathWorks, Inc.

Trademarks

MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

Patents

 $MathWorks\ products\ are\ protected\ by\ one\ or\ more\ U.S.\ patents.\ Please\ see\ www.mathworks.com/patents\ for\ more\ information.$

Revision History

March 2021

New for Version 1.0 (Release 2021a)



	Blocks
1	
	Tools
2	

Blocks

Take DDS Sample

Receive Data Samples from DDS network Library: DDS Blockset



Description

The DDS Blockset provides Take DDS Sample and Write DDS Sample blocks to help model DDS applications. The Take DDS Sample block receives samples from the DDS network and converts them to signals that can be used in the Simulink[®] environment. The Write DDS Sample block takes signals from the Simulink environment and converts them to samples to send data into the DDS network. These blocks are preconfigured to be compatible with the DDS network and provide type checking functionality.

Ports

Input

Port_1 — Input DDS sample

scalar | vector | matrix

Receives samples from the DDS network. Samples are converted to signals that can be used within the Simulink environment. The conversion preserves the value and data type of the original DDS sample.

Data Types: bus

Output

Port_S — Show whether a DDS sample was received

scalar

Shows whether a sample was received successfully from the DDS network. If a sample was received successfully the block outputs 1, otherwise it outputs 0.

Dependencies

To enable this port, select the check box labeled Show receive status.

Data Types: double

Port_1 — Output signal

scalar | vector | matrix

Converted signal that can be used in the Simulink environment.

Data Types: bus

Show receive status — Show whether a DDS sample was received

off (default) | on

Select this check box to show whether a sample was received from the DDS network. If a sample was received successfully the block outputs 1, otherwise it outputs 0.

Programmatic Use
Block Parameter: ShowQueueStatus
Type: character vector
Values: 'on' | 'off'
Default: 'off'

Extended Capabilities

C/C++ Code Generation Generate C and C++ code using Simulink® Coder^m.

See Also Write DDS Sample

Write DDS Sample

Send Data Samples to DDS network Library: DDS Blockset



Description

The DDS Blockset Library provides Take DDS Sample and Write DDS Sample blocks to help model DDS applications. The Take DDS Sample block receives samples from the DDS network and converts them to signals that can be used in the Simulink environment. The Write DDS Sample block converts signals from the Simulink environment to samples to send data into the DDS network. These blocks are preconfigured to be compatible with the DDS network and provide type checking functionality.

Ports

Input

Port_Enable — External enable signal

scalar

Enables the block to send a sample into the DDS network. For input values greater than 0 send is enabled. For input values less than or equal to 0 send is disabled.

Dependencies

To display this port, select the check box labeled **Show enable port**.

Data Types: double

Port_1 — Input signal scalar | vector | matrix

Signal in Simulink environment. Signals are converted to samples that applications use to send data into the DDS network. The conversion preserves the value and data type of the original signal.

Data Types: bus

Output

Port_1 - Output DDS sample

scalar | vector | matrix

Sends samples into the DDS network.

Data Types: bus

Show enable port — Display the enable port off (default) | on

Select this check box to display enable port.

Programmatic Use
Block Parameter: ShowEnablePort
Type: character vector
Values: 'on' | 'off'
Default: 'off'

Extended Capabilities

C/C++ Code Generation Generate C and C++ code using Simulink® Coder^m.

See Also

Take DDS Sample

Tools

DDS Dictionary

Manage DDS aspects of an application

Description

To manage the DDS aspects of your applications in the Simulink environment, you can use a section of the Simulink data dictionary that contains DDS properties called the DDS Dictionary. The DDS Dictionary is an intuitive graphical interface that enables you to quickly create and edit the DDS Domains, Topics, Data Samples, and Quality of Service (QoS).

DDS LIE	BRARIES													
Ł	<u></u>					Î	?							
Import	Library	Module	Struct Co	nst Enum	Duplicate	Delete	Help							
FILE	CREATE		ADD		EDIT	г	HELP							
Types	Domains	Qos						De	tails					4
					Filte	r contents	5	Si	nulink.E	Bus: ShapeT	ype			
	N	ame			Туре						Launch Bus Ed	litor		^
🗸 🖾 ту	peLibrary								Design					
	Enum			Enum					-= >	< 主 🛛	Ŧ			
	MAX_CO			Const					lus eleme		Y			
=	ShapeTyp	e		Struct						1				
									Name	DataType		Dimensions	Min	
									x	int32	real	1	0	
									y shap	int32	real real	1	0	
									snap	1111.52	real	1	U	
									<		_		>	
									< .				,	
								0	escriptio	n:				
													_	~
											Revert	Help	Арр	ly
-														

The DDS Dictionary is composed of three management tabs.

Types

The **Types** tab manages the DDS data types that are used to publish and subscribe Data Samples to the DDS network. The tab can also be used to view and edit the Simulink equivalent data types that are used to compute the application logic in the Simulink environment.

DDS LI	BRARIES														1
Import	Library	Module	Struct C	Const	Enum	Duplicate) Delete	? Help							
FILE	CREATE		ADD			EDIT	т	HELP							-
Types	Domains	Qos							Detail	s					ą
						Filte	er contents	;	Simu	link.B	us: ShapeT	ype			
	N	lame				Туре						Launch Bus Ed	itor		^
Y 🔄 Ту	peLibrary								De	sign					
=	Enum			Enu	m				-		(<u>1</u>	Ŧ			
-	MAX_CO	LOR_LEN		Con	ist					eleme		*			
=	ShapeTyp	e		Stru	ict										
									N	lame		Complexity	Dimensions		
									×		int32	real	1	0	
									У		int32	real	1	0	
									sr	пар	int32	real	1	0	
									<	_		_		>	
														-	
									Des	criptio	1:				
												Revert	Help	Арр	ly

Domains

The **Domains** tab manages the Domains, Topics, and Registered Types so that applications can be configured to publish and subscribe to Topics.

DDS L	BRARIES						
lmport	Library	Domain	;;; Topic	Duplicate) Delete	? Help	
FILE	CREATE	ADI	, ,	EDI	т	HELP	p Z
Types	Domains	Qos					Details
							Filter contents Topic: Circle
	Name			Domai	n ID		Topic Type Registered Type: ShapeType 🗸
🗸 🔄 s	hapesDom	ainLibrary					Type: ShapeType
× 💧	ShapesDelayers	omain	0				Topic Qos: <pre></pre>
	💬 Circle						ShapeType
	💬 Square						ShapeType
	💬 Triang	le					ShapeType

QoS

The **QoS** tab manages the Quality of Service (QoS) available for a DDS application. This tab is primarily read-only, but you can import additional QoS profiles and edit the properties of certain QoS policies.

DDS LIE	BRARIES								
Ł	<u>~</u>			Î	?				
Import	Library	Profile	Duplicate [Delete	Help				
FILE	CREATE	ADD	EDIT		HELP				
Types	Domains	Qos					Deta	ils	
						Filter contents	Data	aReaderQos: DataRead	erQoS
		Name				QoS Type		Name	Value
' 🔄 Bu	uiltInQosLi	brary						Base	value
× 🗀	Event							- base	
	🥯 DataR	eaderQoS		Da	ataReade)S		Durability	
	💬 DataV	/riterQoS		Da	taWriter	5		 Deadline 	
> 🗀	RealTime	Event						 LatencyBudget 	
> 🗖	RealTime	Stream						 Liveliness 	
	Stream						>	 Reliability 	
	bacam							 DestinationOr 	
							>	 History 	
							>	 ResourceLimits 	
								 UserData 	
								 Ownership 	
								TimeBasedFilter	
								ReaderDataLif	
								 TopicFilter 	

Open the DDS Dictionary

- On the DDS Blockset app toolstrip, click Code Interface and select DDS Dictionary.
- In a Simulink data dictionary, if DDS definitions are available, a **DDS Libraries** node appears in the dictionary. If you open the **DDS Libraries** section of the Simulink data dictionary, in the **Details** pane, you can click **Open DDS Libraries** to open the graphical interface for the DDS Dictionary.

Examples

Import XML Files

This example shows how to import additional XML files to include more DDS definitions in your DDS Dictionary.

- **1** Open the DDS Dictionary.
- 2 Click Import.
- **3** Select additional XML files and click **Open**.

Configure Types

This example shows how to create and configure new DDS data types and their equivalent Simulink data types.

- **1** Open the DDS Dictionary.
- 2 Create a new DDS data type library. On the DDS Dictionary toolstrip, click Library.
- **3** Create DDS data types. Select the new library and create the following new DDS data types:
 - Create a DDS Struct data type. On the DDS Dictionary toolstrip, click **Struct**. Select the new DDS type and in the **Details** pane, view its Simulink equivalent data type, a Simulink.Bus object.
 - Create a DDS Const data type. On the DDS Dictionary toolstrip, click Const. Select the new DDS type and in the Details pane, view its Simulink equivalent data type, a numeric MATLAB[®] variable.
 - Create a DDS Enum data type. On the DDS Dictionary toolstrip, click **Enum**. Select the new DDS type and in the **Details** pane, view its Simulink equivalent data type, a Simulink Enumerated data type.

Configure Domains and Topics

This example shows how to create and configure a new Domain and Topic.

- **1** Open the DDS Dictionary. Click the **Domain** tab.
- 2 Create a new Domain. On the DDS Dictionary toolstrip, click **Domain**.
 - Edit the Domain name. For the new Domain, in the **Name** column, edit the name directly in the spreadsheet.
 - Edit the Domain ID. For the new Domain, in the **Domain ID** column, edit the Domain ID directly in the spreadsheet.
- **3** Create a Registered Type. To create a new Topic you must first create a Registered Type for your new Domain. Select the new Domain, in the **Details** pane, click the link icon to create a Registered Type.
- **4** Create a new Topic. On the DDS Dictionary toolstrip, click **Topic**.
 - Set the Registered Type for a Topic. Select the new Topic. In the **Details** pane, from the **Registered Type** drop-down, select a Registered Type from the listed options.
 - Set the Quality of Service (QoS) for a Topic. Select the new Topic. In the **Details** pane, from the **Topic QoS** drop-down, select a QoS profile from listed options.

Configure Quality of Service (QoS)

This example shows how to import and view the Quality of Service (QoS) for your DDS application.

- **1** Open the DDS Dictionary.
- **2** Import QoS from XML.
- **3** In the DDS Dictionary, click the **QoS** tab and view QoS profiles and policies. In the **Details** pane, set QoS policy values as needed.

Types

Name — Names of DDS data type libraries and data types

character vector

The names of DDS data type libraries or DDS data types available in the DDS Dictionary.

Example: TypeLibrary, ShapeType

Type — DDS data type

character vector

The DDS data type in the DDS Dictionary. Supported DDS data types are Const, Enum, and Struct.

To view the equivalent Simulink data type for a DDS data type, select a DDS data type to open the **Details** pane and review the corresponding Simulink data type information.

DDS Data Type	Simulink Data Type
Const	numeric MATLAB variable
Enum	Enumeration
Struct	Simulink.Bus object

Example: Const, Enum, Struct

Domains

Name — Names of Domain libraries, modules, Domains, and Topics

character vector

The names of Domain Libraries, Domain Modules, Domains, and Topics available in the DDS Dictionary.

Example: ShapesDomainLibrary, ShapesLibrary, Circle

Domain ID — ID of a Domain

character vector

Domain ID you can set to specify a Domain.

Example: 1

Topic Type — DDS data type specified for a Topic

character vector

The DDS data type that a Topic uses for its Data Samples.

Example: ShapeType

QoS

Name — Names of QoS libraries and profiles character vector

The names of Quality of Service (QoS) libraries and profiles.

Example: BuiltInQosLibrary, Event, DataReaderQoS

QoS Type — Specifies the type of QoS profile character vector

The QoS Type is a read-only field that shows if a QoS profile is specified for DataReaders or DataWriters.

Example: DataReaderQoS

See Also Code Mappings Editor

Code Mappings Editor

Configure DDS application inports and outports as DataReaders and DataWriters

Description

The Code Mappings editor is a graphical interface where you can configure inports and outports as DataReaders and DataWriters in a DDS application.

To configure the inports and outports, use the Code Mappings editor panes:

- Inports
- Outports

DDS Quick C/C++ Code Settings Code for VTPUT Assistance PREPARE Code Settings Code for Shapesdemo			Simulink prerele										
DDS Quick C/C++ Code Setting Code Interface Interface GENERATE CODE Build View Remove Highlighting Shapesdemo TPUT Advisor e shapesdemo Foperty Inspector # 3 Imports: DataReader VALUE Source DataReader Imports: Imports: DataReader Imports: DataReader Imports: Imports: Imports: Imports: Imports: Imports: Imports: Imports: Imports: Imports: Imports: Code Imports: Imports: Imports: Imports: Imports: Imports: Imports: Import	SIMULA	ATION	DEBUG	MO	DELING	FORMAT	APPS	DDS	×) ¢ 🖳 • (2 - (
anglesication Imports: DataReader Imports: DataReader VALUE Imports: DataType Bus: ShapeType Imports: Data Imports: DataReader Imports: Data Imports: DataReader Imports: Data Imports: Data Imports: Data Imports: Imports: Data Imports: Data Imports Imports	DS •	Quick Start	C/C++ Code	Settings	Code		GENER	ATE CODE		_	View 🏡 Rem Code	nove Highlighting	Share SHARE
Imports: DataReader			shapesdemo							Property Ins	pector		₹×
Image: Configuration Mode Topic Reader XML Path	۲		sdemo 🕨						•	Inports: Dat	taReader		
Image: Source Configuration Mode Source Configuration Mode Topic Reader XML Path Reader XML Path Reader XML Path	-	-								NAME		VALUE	
Buttorize Subsorize Dimensions Imports Outports Code Mappings - DDS Imports Outports Filter contents Source Configuration Mode Topic Reader XML Path Reader XML Path	1 1778											DataReader	
Image: Subserie Source Configuration Mode Topic Source Configuration Mode Use Topic Source Configuration Mode Topic Source Configuration Mode <td>K 7</td> <td>1</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>_</td> <td></td> <td></td> <td></td>	K 7	1								_			
Imports Code Mappings - DDS Imports Code Mappings - DDS Imports Outports Filter contents Source Configuration Mode Topic Reader XML Path Reader XML Path Reader QoS	⇒	:									rpe		e
Source Configuration Mode Configuration Mode Topic Reader XML Path Reader XML Path					_				_				
Imports Outports Code Mappings - DDS Code Mappings - DDS Imports Outports Filter contents Source Configuration Mode Topic Reader XML Path Reader XML Path	AE	Sub	scribe		Compute Logic		Publish				ions		
Imports Code Mappings - DDS Code Mappings - DDS Imports Outports Filter contents Source Configuration Mode Topic Reader XML Path Reader XML Path Reader XML Path	\sim				r								
Imports DDS Dustrient Unit innerit Code Mappings - DDS Imports Imports Source Configuration Mode Use Topic and QoS Filter contents Filter contents				≣h I						Sample	Time	0.1	
Lugie Code Code Mappings - DDS Imports Code Mappings - DDS Imports Filter contents Filter contents Source Configuration Mode Use Topic and QoS Topic ShapesDomainLibra Reader XML Path Reader QoS (default)		Dat	iaReader E			Cutput		DS DataWH	bar	Unit		inherit	
Code Mappings - DDS Imports Outports Code Mappings - DDS Imports Reader XML Path Reader QoS (default) Filter contents Filter contents Reader XML Path Imports Im					l	Logic				✓ Code			
Code Mappings - DDS Imports Reader XML Path Reader QOS (default) Inports Outports Filter contents Filter contents Source Configuration Mode Topic Reader XML Path											ration Mode		-
Code Mappings - DDS Reader XML Path Inports Outports Filter contents Source Configuration Mode Topic Reader XML Path	~											ShapesDomain	Libra
Inports Outports Filter contents Source Configuration Mode Topic Reader XML Path								-				(1-5-10)	
Filter contents Source Configuration Mode Topic Reader XML Path	Coc	de Mappings ·	- DDS					C) # ×	Reader	Qos	(default)	
Source Configuration Mode Topic Reader XML Path	In	ports Out	tports										
								Filter contents					
DataReader Use Topic and QoS ShapesDomainLibrary/ShapesDomain/Circle		Source	Configura	ition Mode		Topic		Reader XML Path					
	∞	DataReader	Use Topic and	d QoS	ShapesDom	ainLibrary/ShapesDor	nain/Circle						

Open the Code Mappings Editor

- Open a model in the DDS Blockset app. On the toolstrip, click **Code Interface** and select **Individual Element Code Mappings**.
- Open a model in the DDS Blockset app. The Code Mappings editor appears as a minimized tab directly below the model canvas. Click the tab to expand the editor.

Examples

Configure DataReaders and DataWriters by using Topic and QoS

This example uses the Shapes Demo example model to show how to configure the DataReaders and DataWriters for a DDS application by using Topic and QoS.

Create a DDS application

- **1** Open the Shapes Demo example model, shapesdemo.
- 2 Create DDS definitions. In the DDS Application Quick Start, select **Create and use default dictionary**.

Configure the DDS Interface

1 Open the Code Mappings editor. On the toolstrip, click **Code Interface** and select **Individual Element Code Mappings**.

	apesdem JLATION		DEBUG		DELING	FORMAT	APPS	D)5 ×	:				5	c 🚬 🗸 (? - (5
DDS DDS	Quic Star ASSISTA	k C/	⊘ /C++ Code Advisor ▼	Settings PREPARE	Code Interface 🗸	Code for shapesdemo		NERATE CODE			tiii Build ▼	View Code) 🔒	Open Re	eport 🔻	<	
v <			shapesdemo						E	<u>.</u>	Property Ins	pector				₹×	
	🕀 🎦 s	hapesde	mo 🕨							•	Inports: Dat	aReader					
		Subscrib			Compute Logi	Dage Cutyon	Publah				NAME Source V Design Data Ty Min Max Dimensi Complex Sample Unit V Code Configur Topic	ons tity Time ration Me			VALUE DataReader Bus: ShapeTy D -1 auto 0.1 inherit Use Topic and ShapesDomai	d QoS	
0	Code Map	oings - Di	DS						0 1		Reader		1		(default)		
_	Inports	Outpo						Filter cont		_							
C	Sou DataRe	ader	Configura Use Topic and		ShapesDon	Topic nainLibrary/ShapesDo	main/Circle	Reader XM	L Path								

- 2 Configure the inports. The inports represent DataReaders in a DDS application.
 - Select the configuration mode. On the **Inports** tab, for the **Configuration Mode**, select Use Topic and QoS.
 - Select the Topic. For the **Topic**, select the Circle Topic specified in the drop-down options by its path (ShapesDomainLibrary/ShapesDomain/Circle). The set of drop-down options shows the Topics that are specified with the same DDS data type as the inport.

- Select the Quality of Service (QoS). To specify QoS, in the Property Inspector, expand the **Code** section, and select your QoS profile from the **Reader QoS** drop-down options. For this example, leave the QoS set to the default.
- **3** Configure the outports. The outports represent DataWriters in a DDS application.
 - Select the configuration mode. On the **Outports** tab, for the **Configuration Mode**, select Use Topic and QoS.
 - Select the Topic. For the **Topic**, select the Circle Topic specified in the drop-down options by its path (ShapesDomainLibrary/ShapesDomain/Circle). The set of drop-down options shows the Topics that are specified with the same DDS data type as the outport.
 - Select the Quality of Service (QoS). To specify QoS, in the Property Inspector, expand the **Code** section, and select your QoS profile from the **Reader QoS** drop-down options. For this example, leave the QoS set to the default.
- **4** To generate the DataReaders and DataWriters, press **Ctrl+D**. You can now view the DataReaders and DataWriters in the editor.

Deploy the DDS application

- **1** Set up the build environment for your target vendor.
- 2 On the toolstrip, click **Build**.
- **3** Deploy the application executable on DDS.

Configure DataReaders and DataWriters by using XML Path

This example uses the Shapes Demo example model to show how to configure the DataReaders and DataWriters for a DDS application by using definitions previously specified in imported XML.

Create a DDS application

- **1** Open the Shapes Demo example model, shapesdemo.
- 2 Create DDS definitions. In the DDS Application Quick Start, select Import from XML and select the shapesexample_qos XML file. When you close the Quick Start, the DDS DataReader and DataWriter definitions from the shapesexample_qos XML file become available in the DDS Dictionary for the example application.

Configure the DDS Interface

1 Open the Code Mappings editor. On the toolstrip, click **Code Interface** and select **Individual Element Code Mappings**.

SI	MULA	TION	DEBUG	мо	DELING	FORMAT	APPS	DDS	×				5 ¢ 2	2 - ?) - (•
	·	💋 Quick Start	⊘ C/C++ Code Advisor ▼	کی Settings	Code Interface ▼	Code for shapesdemo				Build	View Code		Open Report Remove High		≪© Share ▼	
OUTP	JT AS	SSISTANCE		PREPARE			GENERAT	E CODE					RESULTS		SHAR	E
Ser	$\langle \neg$		shapesdemo							Property Ins	pector				₹×	Code
Model Browser	۲	hapes	demo 🕨						•	Inports: Da	taReader					e
odel										NAME			VALU	E		
Σ	Q									Source			DataR	Reader		
	K 7									✓ Design						
	⇒									Data Ty	pe			ShapeTyp	e	
										Min						
	AΞ	Subs	cribe		Compute Logic		Publish			Dimensi			-1			
	\sim									Complex			-1 auto			
								_		Sample			0.1			
						put Output	- DDS	DataW	2	Unit	THINE		inheri	it		
	0									✓ Code						
						Logic				Configu	ration Mo	ode	Use R	Reader XM	L Path	
										Topic			Shape	esDomain	Libra	
	~									Reader	XML Path	ı	MyPa	rticipantLi	brar	
	Code	e Mappings -	DDS					6	9 # ×	Reader	QoS		(defa	ult)		
	Inp	orts Out	ports													
			· /				[Filter contents								
	~	Source	Configura	ation Mode		Topic	R	eader XML Path		1						
	o I	DataReader	Use Reader X		ShapesDom	ainLibrary/ShapesDon										
	<u> </u>	Jutaneauel	USE Reduel A	I'IL FOUT	ShapesDom	anciorary/snapespon	num circle myPartic	aponterbrar y/M	yr arc							
Read	v						60%							Fixed	StepAut	to .

- **2** Configure the inports. The inports represent DataReaders in a DDS application.
 - Select the configuration mode. On the **Inports** tab, for the **Configuration Mode**, select Use Reader XML Path.
 - Select the DataReader. For the Reader XML Path, click and select from the drop-down options the MySquareRdr DataReader specified by its path (myParticipant/ MySubscriber/MySquareRdr). The set of drop-down options shows the readers that are specified with the same DDS data type as the inport. The Topic and QoS properties load based on your reader definition from the XML file.
- **3** Configure the outports. The outports represent DataWriters in a DDS application.
 - Select the configuration mode. On the **Outports** tab, for the **Configuration Mode**, select Use Writer XML Path.
 - Select the DataWriter. For the **Writer XML Path**, click and select from the drop-down options the MySquareWtr DataWriter specified by its path (myParticipant/MySubscriber/ MySquareWtr). The set of drop-down options shows the writers that are specified with the same DDS data type as the outport. The **Topic** and **QoS** properties load based on your writer definition from the XML file.

Deploy the DDS application

- **1** Set up the build environment for your target vendor.
- 2 On the toolstrip, click **Build**.
- **3** Deploy the application executable on DDS.

Inports

Source - Root-level Inport block or bus element

character vector

Identifies a root-level Inport block or an element of an In Bus Element block in the model. Inports map to DataReaders in DDS applications.

Configuration Mode — Determines how DataReaders are specified in a DDS application character vector

DataReaders contain Topic and QoS properties. To specify a DataReader with these properties for a DDS application, you can configure the reader by using the configuration mode Use Reader XML Path or Use Topic and QoS.

If you have previously defined DataReaders in XML, you can use the Use Reader XML Path configuration mode. For each inport, in the **Reader XML Path** column, you can select a previously defined DataReader from the drop-down options.

If you do not have previously defined DataReaders, you can use the Use Topic and QoS configuration mode. For each inport, use the **Topic** drop-down options to select a Topic, and then use the Property Inspector to specify the QoS policies.

Topic — Specifies the Topic that a DDS application subscribes to

character vector

Select or generate a Topic for each DataReader.

Reader XML Path — Specifies the DataReader

character vector

Select or generate a DataReader for each inport.

Outports

Source — Name of root Outport block or bus element

character vector

Identifies a root-level DDS Outport block or an element of an Out Bus Element block in the model. Outports map to DataWriters in DDS applications.

Configuration Mode — Determines how DataWriters are specified in a DDS application character vector

DDS DataWriters contain Topic and QoS properties. To specify a DataWriter with these properties for a DDS application, you can configure the writer by using the configuration mode Use Writer XML Path or Use Topic and QoS.

If you have previously defined DataWriters in XML, you can use the Use Writer XML Path configuration mode. For each outport, in the Writer XML Path column, you can select a previously defined DataWriter from the drop-down options.

If you do not have previously defined DataWriters, you can use the Use Topic and QoS configuration mode. For each outport, use the **Topic** drop-down options to select a Topic, and then use the Property Inspector to specify the QoS policies.

Topic — Specifies the Topic that a DDS application publishes

character vector

Select or generate a Topic for each DataWriter.

Writer XML Path — Specifies the DataWriter

character vector

Select or generate a DataWriter for each outport.

See Also DDS Dictionary