MicroLabBox

RTI Reference

Release 2021-A – May 2021

dSPACE

How to Contact dSPACE

Mail:	dSPACE GmbH Rathenaustraße 26 33102 Paderborn Germany
Tel.:	+49 5251 1638-0
Fax:	+49 5251 16198-0
E-mail:	info@dspace.de
Web:	http://www.dspace.com

How to Contact dSPACE Support

If you encounter a problem when using dSPACE products, contact your local dSPACE representative:

- Local dSPACE companies and distributors: http://www.dspace.com/go/locations
- For countries not listed, contact dSPACE GmbH in Paderborn, Germany.
  Tel.: +49 5251 1638-941 or e-mail: support@dspace.de

You can also use the support request form: http://www.dspace.com/go/supportrequest. If you are logged on to mydSPACE, you are automatically identified and do not need to add your contact details manually.

If possible, always provide the relevant dSPACE License ID or the serial number of the CmContainer in your support request.

Software Updates and Patches

dSPACE strongly recommends that you download and install the most recent patches for your current dSPACE installation. Visit http://www.dspace.com/go/patches for software updates and patches.

Important Notice

Contents

About This Reference

Content

This document provides you with detailed information about the Real-Time Interface (RTI) of your MicroLabBox.

Symbols

dSPACE user documentation uses the following symbols:

Symbol	Description
DANGER	Indicates a hazardous situation that, if not avoided, will result in death or serious injury.
WARNING	Indicates a hazardous situation that, if not avoided, could result in death or serious injury.
CAUTION	Indicates a hazardous situation that, if not avoided, could result in minor or moderate injury.
NOTICE	Indicates a hazard that, if not avoided, could result in property damage.
Note	Indicates important information that you should take into account to avoid malfunctions.
Tip	Indicates tips that can make your work easier.
	Indicates a link that refers to a definition in the glossary, which you can find at the end of the document unless stated otherwise.
	Precedes the document title in a link that refers to another document.

Naming conventions

dSPACE user documentation uses the following naming conventions:

%name% Names enclosed in percent signs refer to environment variables for file and path names.

< > Angle brackets contain wildcard characters or placeholders for variable file and path names, etc.

Examples:

- Where you find terms such as `rti<XXXX>` replace them by the RTI platform support you are using, for example, `rti1007`.
- Where you find terms such as `<model>` or `<submodel>` in this document, replace them by the actual name of your model or submodel. For example, if the name of your Simulink model is `smd_1007_sl.slx` and you are asked to edit the `<model>_usr.c` file, you actually have to edit the `smd_1007_sl_usr.c` file.

RTI block name conventions All I/O blocks have default names based on dSPACE's board naming conventions:

- Most RTI block names start with the board name.
- A short description of functionality is added.
- Most RTI block names also have a suffix.

Suffix	Meaning
B	Board number (for PHS-bus-based systems)
M	Module number (for MicroAutoBox II)
C	Channel number
G	Group number
CON	Converter number
BL	Block number
P	Port number
I	Interrupt number

A suffix is followed by the appropriate number. For example, DS2201IN_B2_C14 represents a digital input block located on a DS2201 board. The suffix indicates board number 2 and channel number 14 of the block. For more general block naming, the numbers are replaced by variables (for example, DS2201IN_Bx_Cy).

Special folders

Some software products use the following special folders:

Common Program Data folder A standard folder for application-specific configuration data that is used by all users.

```
%PROGRAMDATA%\dSPACE\<InstallationGUID>\<ProductName>
```

or

```
%PROGRAMDATA%\dSPACE\<ProductName>\<VersionNumber>
```

Documents folder A standard folder for user-specific documents.

```
%USERPROFILE%\Documents\dSPACE\<ProductName>\
<VersionNumber>
```

Local Program Data folder A standard folder for application-specific configuration data that is used by the current, non-roaming user.

```
%USERPROFILE%\AppData\Local\dSPACE\<InstallationGUID>\
<ProductName>
```

Accessing dSPACE Help and PDF Files

After you install and decrypt dSPACE software, the documentation for the installed products is available in dSPACE Help and as PDF files.

dSPACE Help (local) You can open your local installation of dSPACE Help:

- On its home page via Windows Start Menu
- On specific content using context-sensitive help via **F1**

dSPACE Help (Web) You can access the Web version of dSPACE Help at www.dspace.com/go/help.

To access the Web version, you must have a *mydSPACE* account.

PDF files You can access PDF files via the icon in dSPACE Help. The PDF opens on the first page.

General Information on the RTI Blockset of the MicroLabBox

Introduction

To give you basic information on the RTI Blockset of MicroLabBox (RTI1202).

Where to go from here

Information in this section

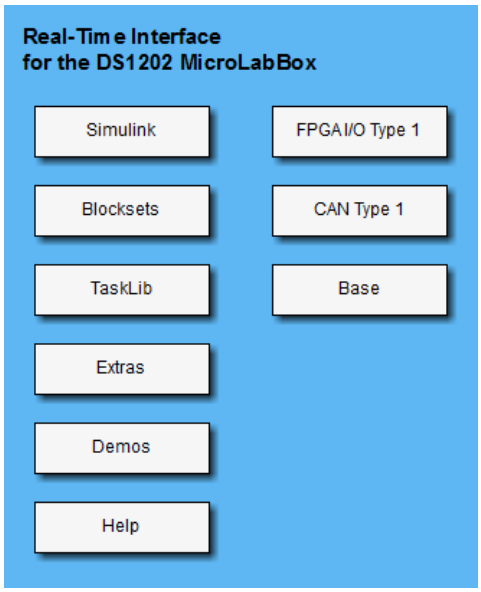
Overview of RTI1202

Introduction

The Real-Time Interface (RTI) **rtilib1202** provides the RTI blocks that implement the functionality and I/O capabilities of MicroLabBox in Simulink models. These RTI blocks are designed to specify the hardware setup for real-time applications. Furthermore, rtilib1202 provides access to demo models and additional RTI blocksets.

RTI blockset

After you enter **rti1202** in the MATLAB Command Window, the RTI board library for MicroLabBox is displayed.



The following rtilib1202 components are available in the **Library: rtilib1202** window:

Simulink Opens the Simulink Library Browser providing the standard Simulink block library.

Blocksets Offers access to optional RTI blocksets that provide specific functionalities for various dSPACE hardware devices. Specific license keys are required for some of these blocksets.

For example, MicroLabBox supports the following blocksets:

- RTI CAN Blockset
- RTI CAN MultiMessage Blockset
- RTI USB Flight Recorder Blockset
- RTI Ethernet Blockset

For detailed information, see Overview of the Optional RTI Blocksets on page 21.

TaskLib Offers RTI blocks for modeling interrupts in Simulink. For detailed information, see TaskLib Block Reference (RTI and RTI-MP Implementation Reference 🕮).

Extras Offers access to RTI blocks for special purposes, such as the service code for dSPACE experiment software. For detailed information, see Extras Block Reference (RTI and RTI-MP Implementation Reference 🕮).

Demos Opens the **rti1202demolib** that provides example models.

Base Offers access to RTI blocks that access functions of MicroLabBox's base board DS1202. For detailed information, see Overview of the Base Library on page 15.

FPGA I/O Type 1 Offers access to the RTI blocks providing functions of MicroLabBox's FPGA I/O, for example, for A/D conversion and PWM signal generation. For detailed information, see Overview of the FPGA I/O Type 1 Library on page 16.

CAN Type 1 Offers access to the dSPACE blockset for implementing CAN bus communication. For detailed information, see Basics on the RTI CAN Blockset (RTI CAN Blockset Reference 🕮).

Help Opens dSPACE Help and displays an overview of the available user documentation for implementing real-time applications with MicroLabBox.

Tip

You can open rtilib1202 components by double-clicking the appropriate button in the **Library: rtilib1202** window.

Related topics

Basics

Basics on the RTI CAN Blockset (RTI CAN Blockset Reference 🕮)

References

Extras Block Reference (RTI and RTI-MP Implementation Reference 🕮)
Overview of the Base Library....15
Overview of the FPGA I/O Type 1 Library....16
TaskLib Block Reference (RTI and RTI-MP Implementation Reference 🕮)

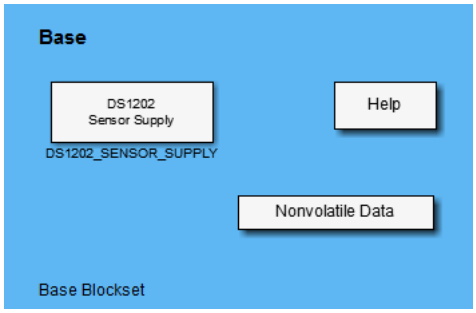
Overview of the Base Library

Introduction

To provide access to specific capabilities of the MicroLabBox's base board DS1202.

Base blockset

Click the Base button in the RTI board library window to display the Base Blockset window.



The Base blockset provides the following RTI blocks:

- Power supply for a connected sensor
  - DS1202_SENSOR_SUPPLY on page 26
- Access to the NVDATA blockset, refer to Nonvolatile Data Handling (NVDATA) on page 105.

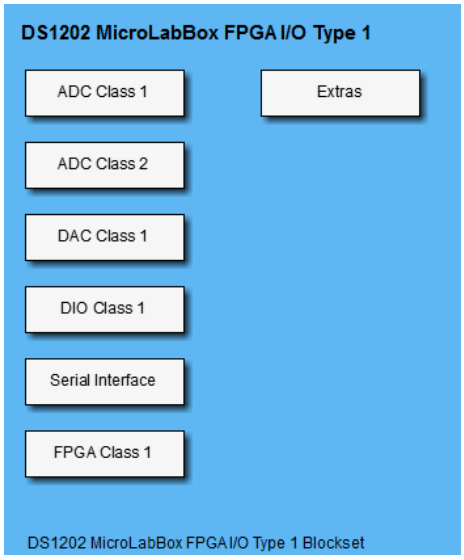
Overview of the FPGA I/O Type 1 Library

Introduction

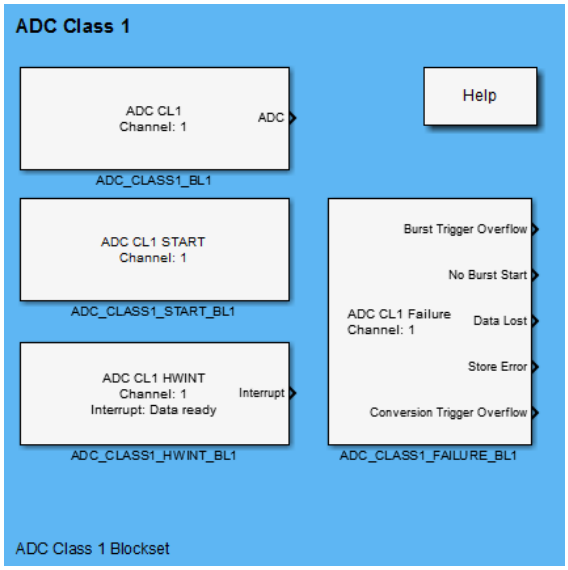
To provide sublibraries to implement MicroLabBox's FPGA I/O in your Simulink model.

DS1202 MicroLabBox FPGA I/O Type 1

After you click the FPGA I/O Type 1 button in the RTI board library window, the following sublibraries are displayed:



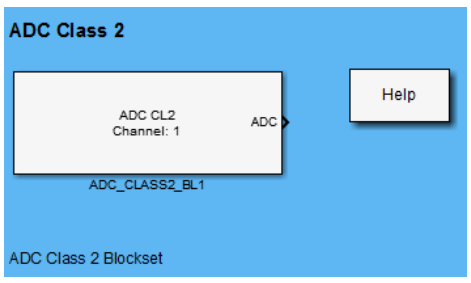
ADC Class 1 The sublibrary contains the blocks that provide access to the Class 1 analog input channels.



The ADC Class 1 blockset provides the following RTI blocks:

- A/D Conversion
  - ADC_CLASS1_BLx on page 46
  - ADC_CLASS1_START_BLx on page 56
- Failure Handling
  - ADC_CLASS1_FAILURE_BLx on page 58
- Interrupt Handling
  - ADC_CLASS1_HWINT_BLx on page 38

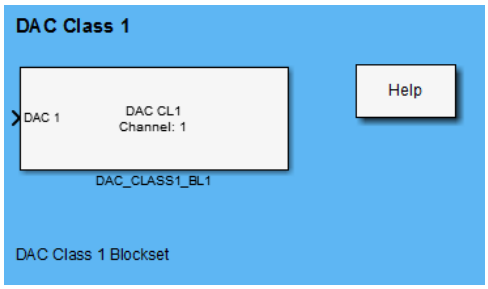
ADC Class 2 The sublibrary contains the blocks that provide access to the Class 2 analog input channels.



The ADC Class 2 blockset provides the following RTI block:

- ADC_CLASS2_BLx on page 61

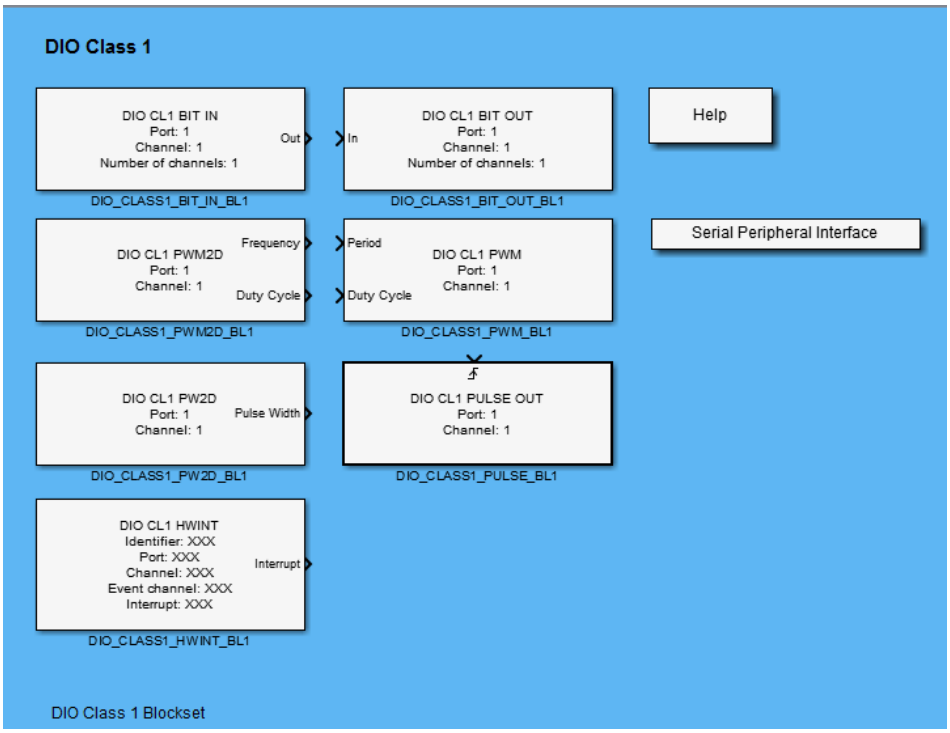
DAC Class 1 The sublibrary contains the blocks that provide access to the Class 1 analog output channels.



The DAC Class 1 blockset provides the following RTI block:

- DAC_CLASS1_BLx on page 64

DIO Class 1 The sublibrary contains the blocks that provide access to the Class 1 digital I/O channels.

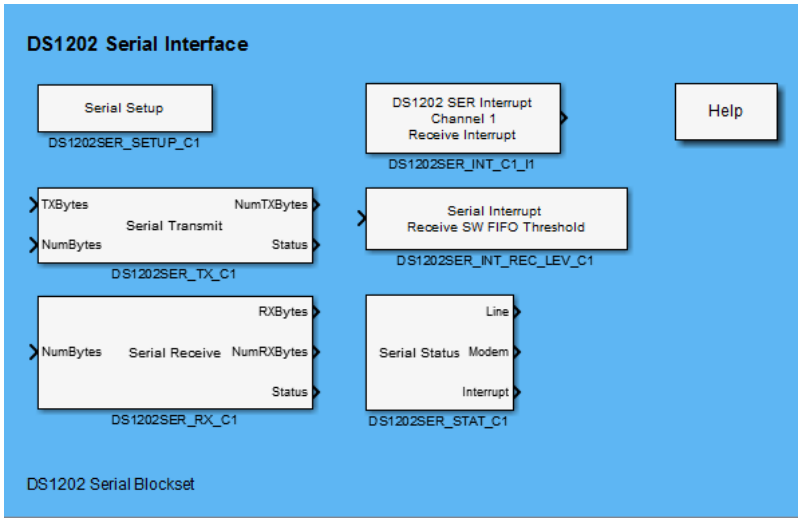


The DIO Class 1 blockset provides the following RTI blocks:

- Bit I/O
  - DIO_CLASS1_BIT_IN_BLx on page 70
  - DIO_CLASS1_BIT_OUT_BLx on page 75
- PWM Signal Generation
  - DIO_CLASS1_PWM_BLx on page 82

- PWM Signal Measurement
  - DIO_CLASS1_PWM2D_BLx on page 90
- Pulse Generation
  - DIO_CLASS1_PULSE_BLx on page 95
- Pulse Width Measurement
  - DIO_CLASS1_PW2D_BLx on page 100
- Interrupts
  - DIO_CLASS1_HWINT_BLx on page 41
- Serial Peripheral Interface (separate library)
  - DIO_CLASS1_SPI_SETUP_BLx on page 142
  - DIO_CLASS1_SPI_CYCLE_SETUP_BLx on page 146
  - DIO_CLASS1_SPI_RX_BLx on page 152
  - DIO_CLASS1_SPI_TX_BLx on page 155

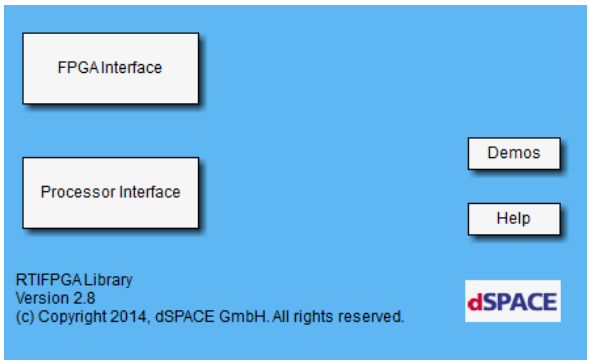
Serial Interface The sublibrary contains the blocks that provide access to the serial interface of the board.



The Serial Interface blockset provides the following RTI blocks:

- DS1202SER_SETUP_Cx on page 118
- DS1202SER_STAT_Cx on page 123
- DS1202SER_TX_Cx on page 127
- DS1202SER_RX_Cx on page 131
- DS1202SER_INT_Cx_Iy on page 135
- DS1202SER_INT_REC_LEV_Cx on page 138

FPGA Class 1 Click the FPGA Class 1 button to display the RTIFPGA Library window.

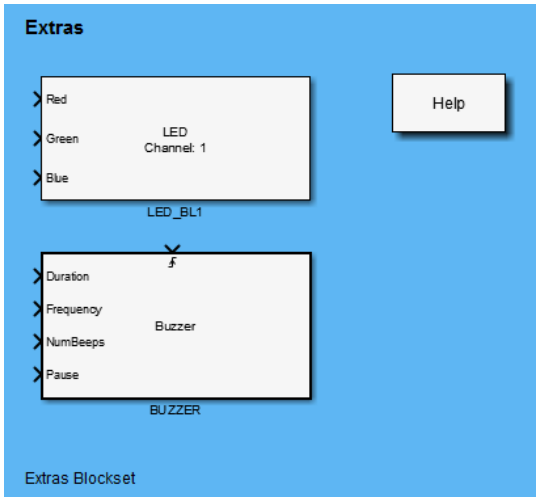


The RTI FPGA Class 1 blockset provides access to the programmable FPGA Class 1 module of MicroLabBox. To use this module, the optional RTI FPGA Programming Blockset is required. Refer to RTI FPGA Programming Blockset Guide 🕮.

Note

The FPGA Interface sublibrary provides two different frameworks for MicroLabBox. You can either implement an FPGA application that only has custom I/O features, or an FPGA application with a combination of custom I/O features and standard I/O features. For more information, refer to General Information on FPGA Support (MicroLabBox Features 🕮).

Extras The sublibrary contains the blocks that provide access to the buzzer and the customizable LEDs of the board.



The Extras blockset provides the following RTI blocks:

- LED_BLx on page 32
- BUZZER on page 28

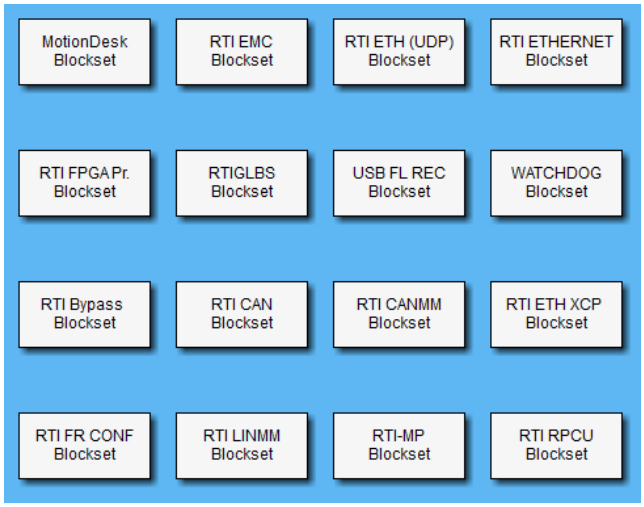
Overview of the Optional RTI Blocksets

Introduction

To provide access to optional RTI blocksets. The blockset buttons are only enabled if the blocksets were installed according to the required licenses.

Blocksets

Click the Blocksets button in the RTI board library window to display the library with the optional blocksets as follows:



An optional blockset is only enabled if it supports the active platform and, where required, an appropriate license is installed. For example, MicroLabBox supports the following optional blocksets:

- RTI CAN Blockset

  For further information, refer to Basics on the RTI CAN Blockset (RTI CAN Blockset Reference 🕮).
- RTI FPGA Programming Blockset

  For further information, refer to RTI FPGA Programming Blockset - FPGA Interface Reference 🕮 and RTI FPGA Programming Blockset - Processor Interface Reference 🕮.
- RTI Multiprocessor Systems Blockset

  For further information, refer to RTI-MP Blockset Reference (RTI and RTI-MP Implementation Reference 🕮).
- RTI USB Flight Recorder Blockset

  For further information, refer to RTI USB Flight Recorder Blockset Reference 🕮.
- RTI Ethernet Blockset

  For further information, refer to RTI Ethernet Blockset Reference 🕮.
- RTI Electric Motor Control Blockset

  For further information, refer to RTI Electric Motor Control Blockset Reference 🕮.

How to Create a Real-Time Application for MicroLabBox

Introduction

Provides instructions for using RTI to create a real-time application for MicroLabBox.

Preconditions

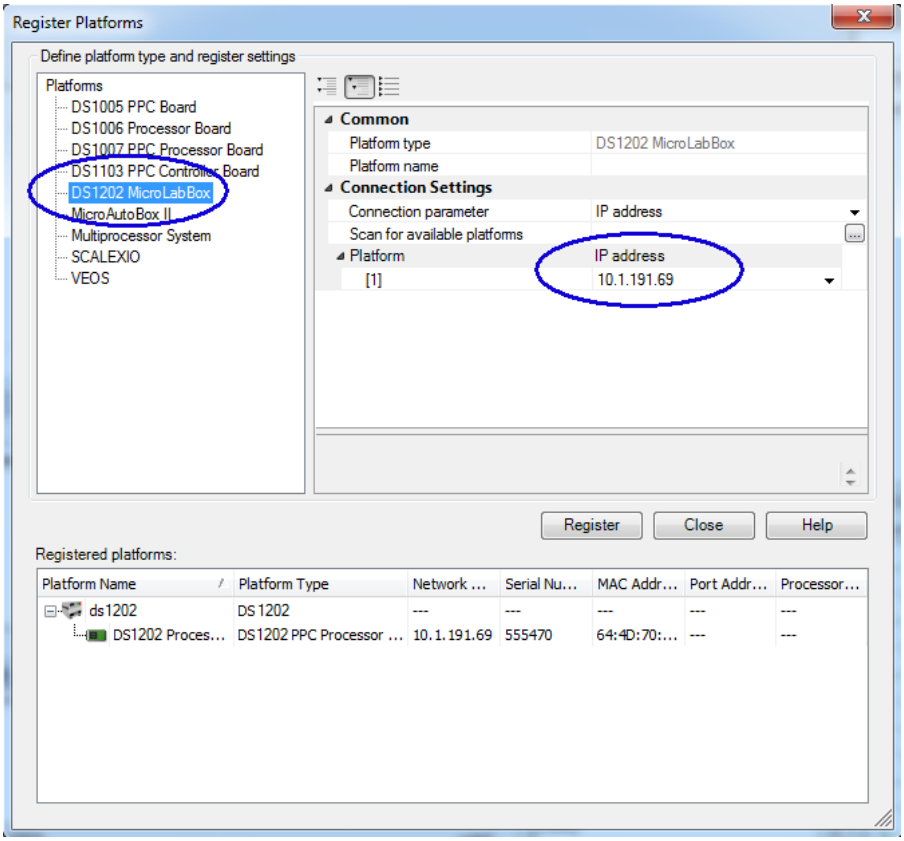
Your host PC and your MicroLabBox must be connected to your TCP/IP network and switched on.

For further information on the installation and configuration of MicroLabBox, refer to MicroLabBox Hardware Installation and Configuration 🕮.

Registering the platform

Before you can access your MicroLabBox from your host PC via RTI it must be registered.

You can use ControlDesk to do this. In the **Register Platforms** dialog, select the board type **DS1202 MicroLabBox**. Enter the IP address of your MicroLabBox. Then click **Register**.



You can access MicroLabBox via your host PC. For more details on registering platforms, refer to Register Platforms (ControlDesk Platform Management 🕮).

Add MicroLabBox RTI blocks to your Simulink model

You can integrate MicroLabBox to your model using MATLAB® and Simulink®.

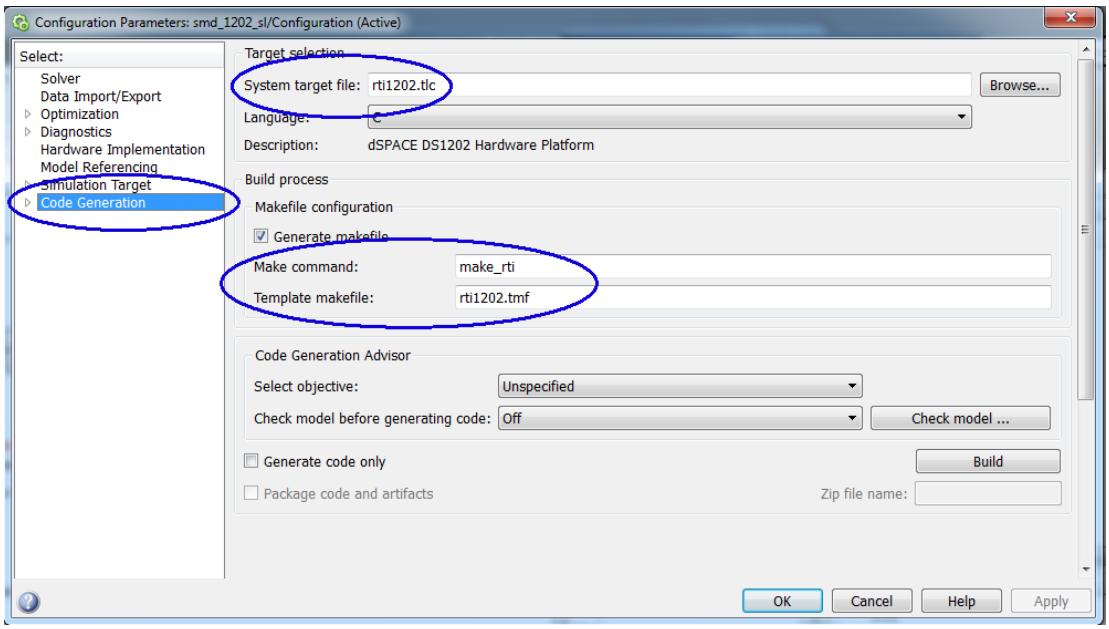
Open your Simulink model and add blocks from the MicroLabBox blockset to it. You can do so by dragging them from the opened library window.

As a result, your model is able to access the input and output pins provided by MicroLabBox.

Building, downloading and starting your application

To run your model on MicroLabBox, you must build and download the appropriate application.

You can do this using the Simulink® Coder™. On the **Code Generation** page of the model's **Configuration Parameters** dialog, the **System target file**, the **Make command**, and the **Template makefile** must be specified correctly. Then click **Build**.



Your application is built, downloaded to MicroLabBox, and started. Depending on your MATLAB version, the single steps of the build process are reported in the MATLAB command window or in the MATLAB Diagnostic Viewer.

Result

You have used RTI to create a real-time application on your MicroLabBox.

More information and examples

For further information on starting to work with your MicroLabBox, refer to DS100x, DS110x, MicroAutoBox II, MicroLabBox – Software Getting Started 🕮.

For further information on working with RTI, refer to RTI and RTI-MP Implementation Guide 🕮.

The **Demos** sublibrary of the DS1202 blockset provides examples for models that access MicroLabBox.

Basic Functions

Introduction

The RTI Blockset of MicroLabBox (RTI1202) provides access to specific capabilities of the MicroLabBox's base board DS1202 and to the board's optical and acoustic signals.

You find:

- DS1202_SENSOR_SUPPLY in the Base library.
- BUZZER and LED_BLx in the I/O FPGA Type 1 - Extras library.

Where to go from here

Information in this section

DS1202_SENSOR_SUPPLY

Purpose

To provide an adjustable power supply for a connected sensor.

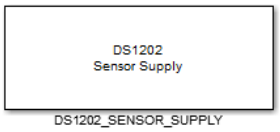
Where to go from here

Information in this section

Block Description (DS1202_SENSOR_SUPPLY)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To provide an adjustable power supply for a connected sensor.

Description

You can add at most one **DS1202_SENSOR_SUPPLY** block to your model to specify the output voltage of the adjustable power supply for sensors.

I/O mapping

For information on connecting the sensor to the adjustable power supply port of MicroLabBox, refer to Connector Pinouts (MicroLabBox Hardware Installation and Configuration 🕮) .

I/O characteristics

MicroLabBox provides the voltage level that you specified on the Unit page (refer to Unit Page (DS1202_SENSOR_SUPPLY) on page 27) in the range 2 V … 20 V at the sensor supply channel. The maximum current must not exceed 0.2 A.

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (DS1202_SENSOR_SUPPLY) on page 27)

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions:

- SensorSupply_create
- SensorSupply_setVoltage
- SensorSupply_setSensorState
- SensorSupply_apply
- SensorSupply_start

Unit Page (DS1202_SENSOR_SUPPLY)

Purpose

To specify the voltage of the sensor power supply channel.

Dialog settings

Voltage Lets you enter the voltage level of the sensor supply channel in the range 2 V … 20 V.

BUZZER

Purpose

To generate an acoustic signal.

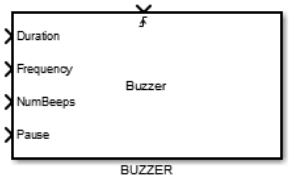
Where to go from here

Information in this section

Block Description (BUZZER)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To generate an acoustic signal.

Description

You can add at most one BUZZER block to your model to access the board's buzzer.

I/O characteristics

When the block is triggered via its trigger input, the configured values are updated and the acoustic signal is output.

The following table describes the ports of the block:

Port	Description
Input	
Duration	Specifies the duration of one beep of the acoustic signal. If you specify a value greater than 2.54 s, the beep is generated permanently.

Port	Description
	Data type: Double Range: 0.01 … 2.54 s Resolution: 10 ms
Frequency	Specifies the frequency of the acoustic signal. Data type: Double Range: 98 … 25,000 Hz Resolution: 40e-6 Hz
NumBeeps	Specifies the number of beeps to be generated after triggering. Data type: UInt32 Range: 0 … 255 If you specify 0, no acoustic signal will be output. If you specify 255, the number of beeps is infinite.
Pause	Specifies the duration of a pause between two beeps of the buzzer. Data type: Double Range: 0.01 … 2.55 s Resolution: 10 ms

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (BUZZER) on page 30)
- Parameters page (refer to Parameters Page (BUZZER) on page 30)

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- Buzzer_create
- Buzzer_setDuration
- Buzzer_setFrequency
- Buzzer_setNumberOfBeeps
- Buzzer_setPause
- Buzzer_apply
- Buzzer_write
- Buzzer_start

Unit Page (BUZZER)

Purpose

To specify the trigger for generating the acoustic signal.

Dialog settings

Trigger type Lets you specify how triggering via the block's trigger port is activated.

Trigger Type	Meaning
Rising	The generation of the acoustic signal is triggered at each low-to-high transition.
Falling	The generation of the acoustic signal is triggered at each high-to-low transition.
Either	The generation of the acoustic signal is triggered at each transition.
Function-call	The generation of the acoustic signal is triggered by a function call.

Related topics

References

Block Description (BUZZER) 28
Parameters Page (BUZZER) 30

Parameters Page (BUZZER)

Purpose

To specify initialization and termination values for the acoustic signal.

Description

The buzzer generates an acoustic signal with the initial values specified on this page. If you trigger the block, you can use the block inports to modify the buzzer configuration during run time. The specified termination values are used for the buzzer configuration when the simulation terminates.

Dialog settings

Initial duration Lets you specify the initialization value for the duration of one beep in the range 0.01 … 2.54 s in steps of 10 ms.

Initial frequency Lets you specify the initialization value for the frequency of the acoustic signal in the range 98 … 25.000 Hz in steps of 40e-6 Hz.

Initial number of beeps Lets you specify the initialization value for the number of beeps in the range 0 … 255.

Initial pause Lets you specify the initialization value for the duration between two beeps in the range 0.01 … 2.55 s in steps of 10 ms.

Last buzzer values Lets you keep the current buzzer configuration when the simulation terminates.

Specific buzzer values Lets you set specific values for the buzzer configuration when the simulation terminates.

If this option is set, the following settings are enabled:

- Duration on termination
- Frequency on termination
- Number of beeps on termination
- Pause on termination

Duration on termination Lets you specify the termination value for the duration of one beep in the range 0.01 … 2.54 s in steps of 10 ms.

Frequency on termination Lets you specify the termination value for the frequency of the acoustic signal in the range 98 … 25.000 Hz in steps of 40e-6 Hz.

Number of beeps on termination Lets you specify the termination value for the number of beeps in the range 0 … 255.

Pause on termination Lets you specify the termination value for the duration between two beeps in the range 0.01 … 2.55 s in steps of 10 ms.

Related topics

References

Block Description (BUZZER)..... 28
Unit Page (BUZZER)..... 30

LED_BLx

Purpose

To control the four customizable LEDs.

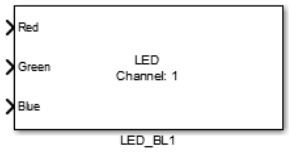
Where to go from here

Information in this section

Block Description (LED_BLx)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To control the four customizable LEDs.

Description

You can add at most four LED_BLx blocks to your model, each controlling one of the four customizable LEDs. You can configure the color of an LED by specifying the three components of the related RGB value.

For example, if you want the LED to light yellow with the RGB value of (255,255,0), you have to set red to 255, green to 255 and blue to 0. The scaled values at the block inputs are (1,1,0).

I/O characteristics

This table shows the scaling between the input of the block and the color intensity as RGB value. The scaling is identical for each color inport.

Block Input	RGB Value
0 … 1.0	0 … 255

You can calculate the block input for a specific RGB value as:
$Value_{Scaled} = 1 / 255 \cdot Value_{RGB}$

The following table describes the ports of the block:

Port	Description
Input	
Red	Specifies the red content of the LED's RGB value. Data type: Double Range: 0 … 1.0
Green	Specifies the green content of the LED's RGB value. Data type: Double Range: 0 … 1.0
Blue	Specifies the blue content of the LED's RGB value. Data type: Double Range: 0 … 1.0

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (LED_BLx) on page 33)
- Parameters page (refer to Parameters Page (LED_BLx) on page 34)

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- Led_create
- Led_setBlueColor
- Led_setGreenColor
- Led_setRedColor
- Led_apply
- Led_write
- Led_start

Unit Page (LED_BLx)

Purpose

To specify the customizable LED to be controlled.

Dialog settings

Channel number Lets you specify the channel number in the range 1 … 4 to control the customizable LEDs 1 … 4.

Related topics

References

Block Description (LED_BLx)..... 32
Parameters Page (LED_BLx)..... 34

Parameters Page (LED_BLx)

Purpose

To specify initialization and termination values for the specified customizable LED.

Description

A customizable LED lights with the color specified by the block's initial values specified on this page. You can use the block inports to modify the color during run time. The specified termination values are used for the LED color when the simulation terminates.

Dialog settings

Initial red intensity Lets you specify the initialization value for the red content of the RGB value used for the LED color in the range 0 … 1.0.

Initial green intensity Lets you specify the initialization value for the green content of the RGB value used for the LED color in the range 0 … 1.0.

Initial blue intensity Lets you specify the initialization value for the blue content of the RGB value used for the LED color in the range 0 … 1.0.

Last color values Lets you keep the current color values when the simulation terminates.

Specific color values Lets you set specific values for the color when the simulation terminates.

If this option is set, the following settings are enabled:

- Red intensity on termination
- Green intensity on termination
- Blue intensity on termination

Red intensity on termination Lets you specify the termination value for the red content of the RGB value used for the LED color in the range 0 … 1.0.

Green intensity on termination Lets you specify the termination value for the green content of the RGB value used for the LED color in the range 0 … 1.0.

Blue intensity on termination Lets you specify the termination value for the blue content of the RGB value used for the LED color in the range 0 … 1.0.

Related topics

References

Interrupts

Introduction

The RTI Blockset of MicroLabBox (RTI1202) provides blocks that you can use to receive hardware interrupts as trigger sources in a Simulink model. For further information, refer to Interrupt Handling (MicroLabBox Features 🕮).

Where to go from here

Information in this section

ADC_CLASS1_HWINT_BLx

Purpose

To make the hardware interrupts of the MicroLabBox ADC Class 1 module available as trigger sources in a Simulink model.

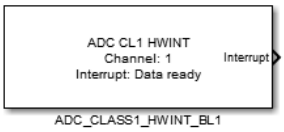
Where to go from here

Information in this section

Block Description (ADC_CLASS1_HWINT_BLx)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To make the hardware interrupts of the MicroLabBox ADC Class 1 module available as trigger sources in a Simulink model.

Description

You can add ADC_CLASS1_HWINT_BLx blocks to your Simulink model to receive hardware interrupts that are generated for an A/D conversion channel via an ADC_CLASS1_BLx block.

The interrupt-generating block must reside in the same model as the ADC_CLASS1_HWINT_BLx blocks.

You can access interrupts of different types for the same channel by adding separate ADC_CLASS1_HWINT_BLx blocks to your model.

Within your Simulink model, each ADC_CLASS1_HWINT_BLx block must be unique regarding the channel and the interrupt type.

For more information on the MicroLabBox interrupts, refer to Interrupt Handling (MicroLabBox Features 🕮).

I/O characteristics

The following table describes the port of the block:

Port	Description
Output	
Interrupt	Trigger output. Data type: Function call

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (ADC_CLASS1_HWINT_BLx) on page 39)

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- AdcCl1AnalogIn_setInterruptMode()

Unit Page (ADC_CLASS1_HWINT_BLx)

Purpose

To specify the ADC Class 1 channel and the type of the interrupts to be received.

Description

Via an ADC_CLASS1_HWINT_BLx block you can specify the interrupt to be received by:

- The *channel number* of the interrupt-generating ADC_CLASS1_BLx block, that resides in the same model.
- The A/D conversion *interrupt type*.

It is not allowed to use ADC_CLASS1_HWINT_BLx blocks in the same model which are identically configured regarding the channel number and the interrupt type. In this case, the build process aborts with an error message.

Dialog settings

Channel number Lets you select a channel number in the range 1 … 24.

Type Lets you select the interrupt type:

Interrupt Type	Meaning
Burst start	The interrupt is generated at the start of an A/D conversion burst. In single conversion mode, the interrupt is generated at the start of an A/D conversion.
Data ready	The interrupt is generated at the completion of an A/D conversion burst. In single conversion mode the interrupt is generated at the completion of an A/D conversion. The interrupt indicates that a buffer with new conversion results is ready to be read.

Interrupt Type	Meaning
Failure	The interrupt is generated if a failure occurred during the conversion. To get more specific information on failures, use the ADC_CLASS1_FAILURE_BLx (refer to ADC_CLASS1_FAILURE_BLx on page 58) block.

DIO_CLASS1_HWINT_BLx

Purpose

To make the hardware interrupts of the MicroLabBox DIO Class 1 module available as trigger sources in a Simulink model.

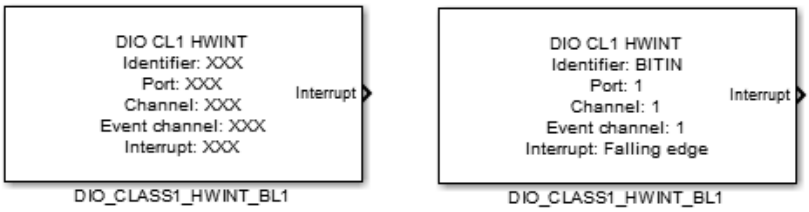
Where to go from here

Information in this section

Block Description (DIO_CLASS1_HWINT_BLx)

Block

Gives you information about the appearance and purpose of the block. The second figure shows a block configured with an event generated by a DIO_CLASS1_BIT_IN_BLx block.



Purpose

To make the hardware interrupts of the MicroLabBox DIO Class 1 module available as trigger sources in a Simulink model.

Description

You can add DIO_CLASS1_HWINT_BLx blocks to your Simulink model to receive hardware interrupts. These interrupts can be generated by various RTI blocks. The following block types can be configured to generate hardware interrupts:

- DIO_CLASS1_BIT_IN_BLx on page 70
- DIO_CLASS1_BIT_OUT_BLx on page 75
- DIO_CLASS1_PWM_BLx on page 82
- DIO_CLASS1_PWM2D_BLx on page 90
- DIO_CLASS1_PW2D_BLx on page 100

- **DIO_CLASS1_SPI_CYCLE_SETUP_BLx** on page 146
- **EMC_MOTOR_SETUP_BLx** (RTI Electric Motor Control Blockset Reference 🕮)
- **EMC_BC_PWM_BLx** (RTI Electric Motor Control Blockset Reference 🕮)
- **EMC_MC_PWM_BLx** (RTI Electric Motor Control Blockset Reference 🕮)

Which interrupt types these blocks can generate depends on their block type. The interrupt-generating block must reside in the same model as the DIO_CLASS1_HWINT_BLx blocks. Whether an interrupt-generating block is identified via the port and channel number or via the unit number depends on the block type.

You can receive different interrupt types from the same generating block by adding separate DIO_CLASS1_HWINT_BLx blocks to your model.

Within your Simulink model, each DIO_CLASS1_HWINT_BLx block must be unique regarding the interrupt-generating RTI block and the interrupt type.

For more information on the MicroLabBox interrupts, refer to Interrupt Handling (MicroLabBox Features 🕮).

I/O characteristics

The following table describes the port of the block:

Port	Description
Output	
Interrupt	Trigger output Data type: Function call

Dialog pages

The dialog settings can be specified on the following page:

- Unit page (refer to Unit Page (DIO_CLASS1_HWINT_BLx) on page 42)

Related RTLib functions

None

Unit Page (DIO_CLASS1_HWINT_BLx)

Purpose

To specify the interrupt-generating block and the type of the interrupts to be received.

Description

Via a DIO_CLASS1_HWINT_BLx block you can specify the interrupt to be received by:

- The interrupt-generating RTI block, that resides in the same model.
- The block-specific interrupt type.

It is not allowed to use DIO_CLASS1_HWINT_BLx blocks in the same model which are identically configured regarding the interrupt-generating block and the interrupt type. In this case, the build process stops with an error message.

Dialog settings

Interrupt source block Lets you select one of the existing interrupt-generating blocks of your model.

How these blocks are characterized depends on the block type:

- DIO Class 1 blocks are characterized by the block identifier, the port number and the channel number, for example, [BITIN]: Port 1, Channel 1.
- The EMC_MOTOR_SETUP_BLx block of the optional *RTI Electric Motor Control* blockset is characterized by the block identifier and the unit number.

The values that characterize the interrupt source block are displayed in the DIO_CLASS1_HWINT_BLx block.

The value of this setting is set to *NOT SELECTED*, when this block is initially added to the model, or afterwards, if the specified interrupt source block has been deleted or modified, so that the specified parameters do not match anymore.

Type Lets you select the interrupt-causing event to be used for generating an interrupt. The list of available interrupt types depends on the specified source block. The selected interrupt type is displayed in the DIO_CLASS1_HWINT_BLx block.

The value of this setting is set to *NOT SELECTED* when this block is initially added to the model, or afterwards, if the specified interrupt type has been modified in the corresponding interrupt source block or this block is deleted.

A/D Conversion

Introduction

The RTI Blockset of MicroLabBox (RTI1202) provides blocks that you can use for converting analog signals. For an overview of which RTI block is appropriate for your use case, refer to A/D Conversion (MicroLabBox Features 🕮).

Where to go from here

Information in this section

Information in other sections

ADC_CLASS1_BLx

Purpose

To provide read access to the ADC Class 1 converters of MicroLabBox.

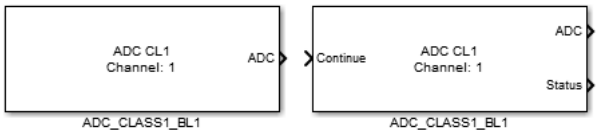
Where to go from here

Information in this section

Block Description (ADC_CLASS1_BLx)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To provide read access to one of the ADC Class 1 converters.

Description

You can use an ADC_CLASS1_BLx block to control one of up to 24 ADC Class 1 analog input channels with a separate A/D converter each.

Each A/D converter can be configured separately for either single conversion mode or burst conversion mode.

You can configure the generation of trigger events. The generation of interrupts is activated, when you add an **ADC_CLASS1_HWINT_BLx** on page 38 block to your model.

I/O mapping

For information on the mapping of converter and channel numbers to the related I/O pins of the MicroLabBox I/O connector, refer to Connector Pinouts (MicroLabBox Hardware Installation and Configuration 🕮).

I/O characteristics

The scaling between the analog input voltage and the output of the block is:

Input Voltage Range	Simulink Output
-10 V … +10 V	-1.0 … +1.0

The following table describes the ports of the block:

Port	Description
Input	
Continue	Controls A/D conversion. Available only if the sample mode of the burst conversion mode is set to continuous. Data type: Boolean: ▪ 0: Successive A/D conversion bursts abort immediately. The conversion results of the aborted burst are not available. The conversion results of the preceding burst are read again instead. ▪ 1: Successive A/D conversion bursts are activated.
Output	
ADC	Outputs the result of the last A/D conversion on the current channel. If the conversion settings are set to burst conversion mode, the output comprises the A/D conversion results of the last burst of A/D conversions on the selected channel. This is a vector of 1 … 8192 results depending on the buffer settings. The earliest value is stored in the first element of the vector. If the conversion settings are set to single conversion mode, the output is the (one) result of the last A/D conversion on the current channel. Data type: Double Range: -1.0 … +1.0
Status	Represents the current status of the output. Available only if the status outport is enabled on the **Advanced** page. Data type: Int16 Range: 0, 1 ▪ 0: A buffer was read repeatedly. New conversion results are not yet available. ▪ 1: A buffer with new conversion results was read.

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (ADC_CLASS1_BLx) on page 48)
- Single Conv page (refer to Single Conv Page (ADC_CLASS1_BLx) on page 49)
- Burst Conv page (refer to Burst Conv Page (ADC_CLASS1_BLx) on page 50)

- Event page (refer to Event Page (ADC_CLASS1_BLx) on page 53)
- Advanced page (refer to Advanced Page (ADC_CLASS1_BLx) on page 54)

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- AdcCl1AnalogIn_create()
- AdcCl1AnalogIn_setConversionMode()
- AdcCl1AnalogIn_setConversTrigger()
- AdcCl1AnalogIn_setBurstTrigger()
- AdcCl1AnalogIn_setBurstSize()
- AdcCl1AnalogIn_setBurstMode()
- AdcCl1AnalogIn_setTriggerLineIn()
- AdcCl1AnalogIn_setTimerPeriod()
- AdcCl1AnalogIn_setTriggerMode()
- AdcCl1AnalogIn_setTriggerLineOut()
- AdcCl1AnalogIn_setConversSwTrigger()
- AdcCl1AnalogIn_setBurstSwTrigger()
- AdcCl1AnalogIn_apply()
- AdcCl1AnalogIn_setBurstAbort()
- AdcCl1AnalogIn_write()
- AdcCl1AnalogIn_start()
- AdcCl1AnalogIn_isDataReady()
- AdcCl1AnalogIn_read()
- AdcCl1AnalogIn_getSingleValue()
- AdcCl1AnalogIn_getBurstCurrent()

Unit Page (ADC_CLASS1_BLx)

Purpose

To provide access to an A/D converter by specifying the related channel number and conversion mode.

Dialog settings

Channel number Lets you select a channel number in the range 1 … 24.

Note

Concurrent access to the same ADC Class 1 channel by multiple blocks or functions is not allowed.

Signal connector pin Displays the signal connector pin for the selected channel.

Reference connector pin Displays the reference connector pin for the selected channel.

Conversion mode Lets you select whether to use the specified A/D converter in single or burst conversion mode.

Conversion Mode	Meaning
Single Conversion	One A/D conversion will be executed. You can configure this conversion on the Single Conv page.
Burst Conversion	A burst of A/D conversions of a user-specified size will be executed. You can configure this conversion on the Burst Conv page.

Related topics

References

Advanced Page (ADC_CLASS1_BLx) 54
Block Description (ADC_CLASS1_BLx) 46
Burst Conv Page (ADC_CLASS1_BLx) 50
Event Page (ADC_CLASS1_BLx) 53
Single Conv Page (ADC_CLASS1_BLx) 49

Single Conv Page (ADC_CLASS1_BLx)

Purpose

To configure the single conversion.

Dialog settings

The settings of this page are only enabled if the Conversion mode on the Unit page is set to Single conversion.

Channel Displays the channel number of the A/D converter that you specified on the Unit page.

Trigger source Lets you select how the conversion is triggered.

Trigger Source	Meaning
Sample base rate	The start of the A/D conversion is triggered with the model step size. You can additionally set the sample time. For details, refer to Sample Time (refer to Advanced Page (ADC_CLASS1_BLx) on page 54).
Trigger line	The start of the A/D conversion is triggered by the trigger line specified in the **Trigger line** setting.

Note

If the sample base rate is selected as the A/D conversion trigger source, the block uses the polling functions for reading the conversion results. If a trigger line is selected as the A/D conversion trigger source, the current read buffer is read immediately without using the polling functions. You can evaluate the read buffer state for reading the conversion results by enabling the status outport of the ADC_CLASS1_BLx block. Alternatively, you can evaluate the data ready interrupt for reading the conversion. For details on the reading methods, refer to ADC Class 1 (MicroLabBox Features 🕮).

Trigger line Lets you select a trigger line number in the range 1 … 16.
This setting is only enabled if the **Trigger source** is set to **Trigger line**.

Related topics

References

Burst Conv Page (ADC_CLASS1_BLx)

Purpose

To configure the burst conversion.

Dialog settings

The settings of this page are only enabled if the **Conversion mode** on the **Unit** page is set to **Burst conversion**.

Channel Displays the channel number of the A/D converter that you specified on the **Unit** page.

Sample mode Lets you select how the start of a burst is triggered.

Sample Mode	Meaning
Triggered	A burst of A/D conversions will be executed after a burst trigger event occurred. In this case, you can configure the burst trigger and the conversion trigger.
Continuous	A burst of A/D conversions will be executed continuously. In this case, you can configure only the conversion trigger.

Trigger source (Burst trigger settings) Lets you select the trigger source for a new burst.

This setting is only enabled if the sample mode is set to **Triggered**.

Trigger Source	Meaning
Sample base rate	The burst A/D conversion start is triggered with the model step size. You can additionally set the sample time. For details, refer to Sample Time (refer to Advanced Page (ADC_CLASS1_BLx) on page 54).
Trigger line	The A/D conversion start is triggered by the line that is specified in **Trigger line (Burst trigger settings)**.

Note

The burst and conversion trigger sources must not be identical on a channel at the same time.

Trigger line (Burst trigger settings) Lets you select a trigger line number in the range 1 … 16 for the burst trigger.

This setting is only enabled if the **Trigger source (Burst trigger settings)** has been set to **Trigger line**.

Trigger source (Conversion trigger settings) Lets you select the trigger source for a conversion within a burst.

Trigger Source	Meaning
Sample base rate	The A/D conversion start is triggered with the model step size. You can additionally set the sample time. For details, refer to Sample Time (refer to Advanced Page (ADC_CLASS1_BLx) on page 54).
Trigger line	The A/D conversion start is triggered by the line that is specified in **Trigger line (Conversion trigger settings)**.
Timer	The A/D conversion is started by the an ADC timer. You can specify the appropriate conversion trigger period via the **Timer interval** setting.

Note

- If the sample base rate is selected as the burst trigger source and the timer is specified as the A/D conversion trigger source, the block uses the polling functions for reading the conversion results. In all other cases in burst mode, the current read buffer is read immediately without using the polling functions. You can evaluate the read buffer state for reading the conversion results by enabling the status outport of the ADC_CLASS1_BLx block. Alternatively, you can evaluate the data ready interrupt for reading the conversion. For details on the reading methods, refer to ADC Class 1 (MicroLabBox Features 🕮).
- The burst and conversion trigger sources must not be identical on a channel at the same time.

Trigger line (Conversion trigger settings) Lets you select a trigger line number in the range 1 … 16 for the conversion trigger.

This setting is only enabled if the **Trigger source (Conversion trigger settings)** is set to **Trigger line**.

Trigger interval Lets you specify the conversion trigger period in the range 1 µs … 1.34 s.

This setting is only enabled if the **Trigger source (Conversion trigger settings)** is set to **Timer**.

Burst size Lets you specify the number of A/D conversions within a burst in the range 1 … 8192.

Offset Lets you specify a position in the swinging buffer at which to start the reading. The position must be in the range 0 … <Burst size>-1. Variables of your Simulink model can be used. You also have to specify the length.

This setting is only enabled if the **Sample mode** is set to **Triggered** and **Burst size** is greater than one.

Length Lets you specify the number of conversion results which are read from the swinging buffer. The length can be specified in the range 1 … <Burst size>-<Offset>. Variables of your Simulink model can be used. You also have to specify the buffer offset from which the conversion results must be read out.

The offset and length allow you to read a burst of conversion results in fractions of the burst size.

This setting is only enabled if the **Sample mode** is set to **Triggered** and **Burst size** is greater than one.

Related topics

References

Advanced Page (ADC_CLASS1_BLx) 54
Block Description (ADC_CLASS1_BLx) 46
Event Page (ADC_CLASS1_BLx) 53
Single Conv Page (ADC_CLASS1_BLx) 49
Unit Page (ADC_CLASS1_BLx) 48

Event Page (ADC_CLASS1_BLx)

Purpose

To configure the trigger events.

Description

A specified trigger event can be handled by an ADC_CLASS1_HWINT_BLx block.

Dialog settings

Channel Displays the channel number of the A/D converter that you specified on the Unit page.

Enable trigger event Lets you enable the generation of trigger events.

Trigger mode Lets you select under which condition the triggered event should be generated.

Trigger Mode	Meaning
Burst start	The trigger is issued at the start of a burst conversion.
Data ready	The trigger is issued when the swinging buffer is ready to be read.
Failure	The trigger is issued if an error occurs, for example, if the processor board can not read the swinging buffer fast enough.

For details on trigger modes, refer to ADC Class 1 (MicroLabBox Features 🕮).

This setting is only enabled if the Enable trigger event is set.

Trigger line Lets you select a trigger line number that you want to assign to the specified event in the range 1 … 16.

Note

It is not allowed to use the same trigger line number in different event-generating blocks.

This setting is only enabled if the Enable trigger event is set.

Related topics

References

Advanced Page (ADC_CLASS1_BLx)

Purpose

To configure advanced conversion settings, such as the sample time.

Dialog settings

Channel Displays the channel number of the A/D converter that you specified on the Unit page.

Use separate ADC_CL1_START block Lets you specify if A/D conversion is to be started from a separate task.

This setting is only enabled if the Trigger source has been set to Sample base rate.

This setting must be made if the sample base rate is set as one of the three trigger sources and the read-out of the conversion results is performed in a triggered subsystem using the data ready or burst start interrupt. For more information on Simulink models which require an ADC_CL1_START block, refer to A/D Conversion (MicroLabBox Features 🕮).

Sample time Lets you specify the sample time of the block.

Sample Time	Meaning
-1	The sample time is inherited from the blocks the block is connected to. If the block is in a triggered subsystem, you must select this setting.
0	The block uses the discrete sample time of the Simulink model.
> 0	The block uses the sample time you specify.

This setting is enabled only if the sample base rate is set as the trigger source.

You can make the setting for the sample time in the ADC_CLASS1_BLx block independently of the sample time setting in a corresponding ADC_CLASS1_START_BLx block, if used.

Note

If the ADC_CLASS1_BLx block is executed in a timer task, it works with the sample time of the fastest block in the task it is part of. To avoid this behavior, the sample time has to be set to the sample time of the model.

Enable Status outport Lets you enable the Status outport, which indicates whether a read buffer with new conversion results was read, or if the read buffer was already read before. For more information on the reading methods, refer to ADC Class 1 (MicroLabBox Features 🕮).

Related topics

References

ADC_CLASS1_START_BLx

Purpose

To start the A/D conversion by software in a separate task.

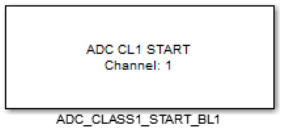
Where to go from here

Information in this section

Block Description (ADC_CLASS1_START_BLx)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To start A/D conversion or A/D burst conversion by software in a separate task.

Description

The ADC_CLASS1_START_BLx block allows you to start A/D conversion or A/D burst conversion by software on the specified A/D conversion channel.

You can use a separate ADC_CLASS1_START_BLx block in your Simulink model if the conversion trigger source or the burst trigger source is set to the sample base rate and the reading of the conversion results is performed in a different task. In all other cases, it is recommended to start A/D conversion in the ADC_CLASS1_BLx.

Note

- The model does not allow more than one ADC_CLASS1_START_BLx block for the same channel.
- The ADC_CLASS1_ADC_START_BLx block is unsuitable for use in an *Enabled Subystem*. The A/D converter is always started independently of the state of the Enable port.
  If you want to start the block only under specific conditions, you must place it in a *Triggered Subsystem* or a *Function-Call Subsystem*.

Other RTI blocks

The ADC_CLASS1_BLx block must also reside in the model – with the same channel number. Additionally, *Use separate ADC_CL1_START block* must be selected in the ADC_CLASS1_BLx settings. For details on the ADC_BLx block, refer to Block Description (ADC_CLASS1_BLx) on page 46.

I/O mapping

For information on the mapping of converter and channel numbers to the related I/O pins of the MicroLabBox I/O connector, refer to Connector Pinouts (MicroLabBox Hardware Installation and Configuration 🕮).

Dialog pages

The dialog settings can be specified on the Unit page (refer to Unit Page (ADC_CLASS1_START_BLx) on page 57).

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- AdcCl1AnalogIn_setConversSwTrigger()
- AdcCl1AnalogIn_setBurstSwTrigger()
- AdcCl1AnalogIn_write()

Unit Page (ADC_CLASS1_START_BLx)

Purpose

To reference the related channel and to start A/D conversion by software.

Dialog settings

Channel number Lets you select a channel number in the range 1 … 24.

Sample time Lets you specify the sample time of the block.

Sample Time	Meaning
-1	The sample time is inherited from the blocks the block is connected to. If the block is in a triggered subsystem, you must select this setting.
0	The block uses the discrete sample time of the Simulink model.
> 0	The block uses the sample time you specify.

Related topics

References

Block Description (ADC_CLASS1_START_BLx) 56

ADC_CLASS1_FAILURE_BLx

Purpose

To indicate the failure state of an A/D conversion.

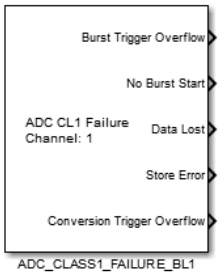
Where to go from here

Information in this section

Block Description (ADC_CLASS1_FAILURE_BLx)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To indicate the failure state of an A/D conversion.

Description

The ADC_CLASS1_FAILURE_BLx returns failure states regarding the current A/D conversion on the specified input channel.

The ADC_CLASS1_BLx block that provides the A/D converter must also reside in the model – with the same channel number. For details, refer to ADC_CLASS1_BLx on page 46.

Each failure state is overwritten with the next reading on the specified channel.

Note

You can use an ADC_CLASS1_FAILURE_BLx block in a *Function Call Subsystem* that is triggered by failure interrupts. When a failure occurs, the outports of the block will indicate the type of failure until the next failure interrupt is received.

I/O characteristics

The following table describes the ports of the block:

Port	Description
Output	
No Burst Start	Indicates that a burst trigger is missing. Data type: Boolean Range: 0 (No error), 1 (Error occurred)
Data Lost	Indicates that a swinging buffer with conversion results, which has not been read by the application, is overwritten with new conversion results. Data type: Boolean Range: 0 (No error), 1 (Error occurred)
Store Error	Indicates that storing the conversion results to the application failed. Data type: Boolean Range: 0 (No error), 1 (Error occurred)
Burst Trigger Overflow	Indicates that a burst trigger is received before the previous burst is completed. Data type: Boolean Range: 0 (No error), 1 (Error occurred)
Conversion Trigger Overflow	Indicates that an A/D conversion trigger is received before the current A/D conversion is completed. Data type: Boolean Range: 0 (No error), 1 (Error occurred)

Dialog pages

The dialog settings can be specified on the Unit page (refer to Unit Page (ADC_CLASS1_FAILURE_BLx) on page 60).

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- AdcCl1AnalogIn_readFailureInfo()
- AdcCl1AnalogIn_getFailureInfo()

Related topics

References

ADC_CLASS1_BLx....46

Unit Page (ADC_CLASS1_FAILURE_BLx)

Purpose

To reference the related channel and return failure information.

Dialog settings

Channel number Lets you select a channel number in the range 1 … 24.

Related topics

References

Block Description (ADC_CLASS1_FAILURE_BLx)....58

ADC_CLASS2_BLx

Purpose

To provide read access to the ADC Class 2 converters of MicroLabBox.

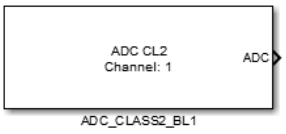
Where to go from here

Information in this section

Block Description (ADC_CLASS2_BLx)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To provide read access to one of the ADC Class 2 converters.

Description

With an ADC_CLASS2_BLx block you can control one of up to 8 differential analog input channels that each have a separate A/D converter.

I/O mapping

For information on the mapping of converter and channel numbers to the related I/O pins of the MicroLabBox I/O connector, refer to Connector Pinouts (MicroLabBox Hardware Installation and Configuration 🕮).

I/O characteristics

The scaling between the analog input voltage and the output of the block is:

Input Voltage Range	Simulink Output
-10.0 V … +10.0 V	-1.0 … +1.0

The following table describes the ports of the block:

Port	Description
Output	
ADC	Outputs the result of the last A/D conversion on the current channel. Data type: Double Range: -1.0 … +1.0

Dialog pages

The dialog settings can be specified on the Unit page (refer to Unit Page (ADC_CLASS2_BLx) on page 62).

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- AdcCl2AnalogIn_create()
- AdcCl2AnalogIn_apply()
- AdcCl2AnalogIn_getInputValue()
- AdcCl2AnalogIn_read()
- AdcCl2AnalogIn_start()

Unit Page (ADC_CLASS2_BLx)

Purpose

To provide access to an A/D converter by specifying the related channel number.

Dialog settings

Channel number Lets you select a channel number in the range 1 … 8.

Note

Concurrent access to the same ADC Class 2 channel by multiple blocks or functions is not allowed.

Signal connector pin Displays the signal connector pin for the selected channel.

Reference connector pin Displays the reference connector pin for the selected channel.

Related topics

References

Block Description (ADC_CLASS2_BLx)..... 61

D/A Conversion

Introduction

The RTI Blockset of MicroLabBox (RTI1202) provides a block that you can use for converting digital signals to analog signals (D/A conversion). For further information, refer to D/A Conversion (MicroLabBox Features 🕮).

DAC_CLASS1_BLx

Purpose

To configure and perform D/A conversion using the MicroLabBox DAC Class 1 converter.

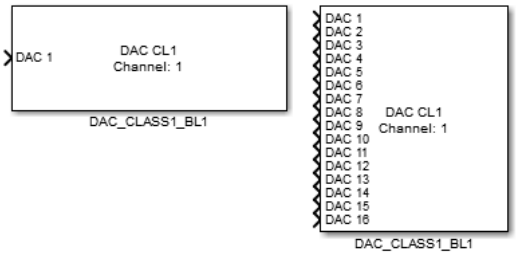
Where to go from here

Information in this section

Block Description (DAC_CLASS1_BLx)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To provide write access to the MicroLabBox DAC Class 1 converters.

Description

One DAC_CLASS1_BLx block controls up to 16 D/A converters. The D/A converters of one block are updated synchronously.

I/O mapping

For information on the mapping of converter and channel numbers to the related I/O pins of the MicroLabBox I/O connector, refer to Connector Pinouts (MicroLabBox Hardware Installation and Configuration 🕮).

I/O characteristics

This table shows the scaling between the input of the block and the analog output voltage:

Simulink Input	Output Voltage Range
-1.0 … +1.0	-10.0 … +10.0 V

The following table describes the ports of the block:

Port	Description
Input	
DAC 1 … DAC 16	Writes the analog value to the D/A channel. Data type: Double

Initialization and termination

- During the model initialization phase, an initial output voltage value is written to each D/A channel. This is especially useful if a channel is used in a triggered or enabled subsystem that is not executed right from the start of the simulation.
- With the initialization value, the D/A channel has a defined output during this simulation phase.
- When the simulation terminates, the D/A channel are set to high impedance (high-Z) by default. Using the parameters **Termination mode** and **Value on termination**, you can specify a user-defined output value on termination and use these settings to drive your external hardware into a safe final condition.

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (DAC_CLASS1_BLx) on page 66)
- Initialization page (refer to Initialization Page (DAC_CLASS1_BLx) on page 66)
- Termination page (refer to Termination Page (DAC_CLASS1_BLx) on page 67)

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- DacCl1AnalogOut_create()
- DacCl1AnalogOut_apply()
- DacCl1AnalogOut_setOutputValue()
- DacCl1AnalogOut_write()
- DacCl1AnalogOut_start()

Unit Page (DAC_CLASS1_BLx)

Purpose

To provide access to one or more D/A converters by referencing the related channel.

Dialog settings

Channel number Lets you select a channel number in the range 1 … 16.

If you want to access more than one D/A channel, this is the number of the first D/A channel.

Note

Concurrent access to the same DAC Class 1 channel by multiple blocks or functions is not allowed.

Number of channels Lets you select the number of channels in the range 1 … 16.

The possible range depends on the first D/A channel spedified via Channel number.

```
NumberOfChannels <= 16 - (ChannelNumber - 1)
```

Related topics

References

Block Description (DAC_CLASS1_BLx)....64
Initialization Page (DAC_CLASS1_BLx)....66
Termination Page (DAC_CLASS1_BLx)....67

Initialization Page (DAC_CLASS1_BLx)

Purpose

To specify the initial values of the enabled D/A channels.

Dialog settings

Channel Displays the range of the enabled D/A channels that are specified on the Unit page.

Initial value Lets you specify the initial output voltage at the start of the simulation for each of the enabled D/A channels. The value must remain in the output voltage range of -10.0 V … +10.0 V.

Related topics

References

Block Description (DAC_CLASS1_BLx)....64
Termination Page (DAC_CLASS1_BLx)....67
Unit Page (ADC_CLASS2_BLx)....62

Termination Page (DAC_CLASS1_BLx)

Purpose

To specify the termination values of the enabled D/A channels.

Dialog settings

Channel Displays the range of the enabled D/A channels that are specified on the Unit page.

Termination mode Lets you set the termination value individually for each of the enabled D/A channels. If the option is cleared, the enabled D/A channels are set to high impedance (high-Z) by default. Previously modified output modes are ignored. If you set the option, the Output mode setting for specifying the termination behavior is enabled.

Output mode Lets you specify whether to use the last output value or a specific output value when the simulation terminates. If you set Output mode to Specific output value, the Value on termination setting is enabled.

Value on termination Lets you select the output voltage at the end of the simulation. The value must remain in the output voltage range of -10.0 V … +10.0 V.

Related topics

References

Block Description (DAC_CLASS1_BLx)....64
Initialization Page (DAC_CLASS1_BLx)....66
Unit Page (ADC_CLASS2_BLx)....62

Bit I/O

Introduction

The RTI Blockset of MicroLabBox (RTI1202) provides blocks that you can use for accessing digital I/O (Bit I/O). For an overview of which RTI block is appropriate for your use case, refer to Bit I/O (MicroLabBox Features 🕮).

Where to go from here

Information in this section

DIO_CLASS1_BIT_IN_BLx

Purpose

To provide read access to the digital input.

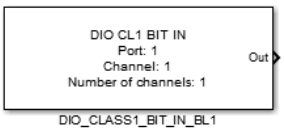
Where to go from here

Information in this section

Block Description (DIO_CLASS1_BIT_IN_BLx)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To provide read access to the digital input.

Description

The block provides read access to a range of channels of a digital I/O port. You can read a single channel by setting the number of channels to one. You can read all channels of the digital I/O port by setting the number of channels to the port width.

I/O mapping

For information on the mapping of channel numbers to the related I/O pins of the MicroLabBox I/O connector, refer to Connector Pinouts (MicroLabBox Hardware Installation and Configuration 🕮).

I/O characteristics

The following table describes the ports of the block:

Port	Description
Output	
Out	Provides the values that are read from the DIO Class 1 input channels. Data type: Array of Booleans. Its length is specified by the number of channels. Range: 0 (= low), 1 (= high)

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (DIO_CLASS1_BIT_IN_BLx) on page 71)
- Event page (refer to Event Page (DIO_CLASS1_BIT_IN_BLx) on page 72)

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- DioCl1DigIn_create()
- DioCl1DigIn_setInputFilter()
- DioCl1DigIn_setInterruptMode()
- DioCl1DigIn_setTrigerMode()
- DioCl1DigIn_setTriggerLineOut()
- DioCl1DigIn_setEventDownsample()
- DioCl1DigIn_setEventDelay()
- DioCl1DigIn_apply()
- DioCl1DigIn_getChMaskInData()
- DioCl1DigIn_read()
- DioCl1DigIn_start()

Unit Page (DIO_CLASS1_BIT_IN_BLx)

Purpose

To specify the digital input channels and noise filtering.

Dialog settings

Port number Lets you select the port number to be used in the range 1 … 3.

Channel number Lets you select the first channel number to be used in the range 1 … 16.

Note

Concurrent access to the same DIO Class 1 channel by other blocks or functions is not allowed.

Number of channels Lets you select the number of channels to be read.

The possible value depends on the width of the digital I/O port and the number of the first channel to be read.

```
NumberOfChannels <= 16 - (ChannelNumber - 1)
```

Note

Concurrent access to the same DIO Class 1 channel by other blocks or functions is not allowed.

Signal connector pin Displays the range of signal connector pins for the selected channels. The range begins with the connector pin of the first selected channel.

Reference connector pin Displays the range of reference connector pins for the selected channels.

Enable noise filter Lets you enable noise filtering for the input signals.

Filter interval Lets you specify the time interval for the noise filter in the range 0 … 10 ms.

For intervals shorter than 650 ns the resolution is 10 ns. For greater intervals, the accuracy is better than 10% of the specified value.

Related topics

References

Block Description (DIO_CLASS1_BIT_IN_BLx) 70
Event Page (DIO_CLASS1_BIT_IN_BLx) 72

Event Page (DIO_CLASS1_BIT_IN_BLx)

Purpose

To enable event generation for the specified channel.

Description

You can configure interrupt event generation, trigger event generation or both at *falling edge* and/or at *rising edge* transitions on a specific channel.

Dialog settings

Port Displays the port number that you selected on the Unit page.

Channel Displays the range of channels that you selected on the Unit page.

Enable event generation Lets you enable the generation of interrupt or trigger events or both.

If you enable event generation, at least one of the transitions *falling edge* or *rising edge* must be activated.

Event mode Lets you select the events to be generated at the activated transition.

Event Mode	Meaning
Interrupt	Generate interrupt events at the activated transitions.
Trigger	Generate trigger events at the activated transitions.
Interrupt and Trigger	Generate both interrupts and trigger events at the activated transitions.

If you select interrupt generation, your model must contain a related DIO_CLASS1_HWINT_BLx block configured with the same port and channel number.

Activate event on falling edge Lets you enable event generation at high to low transition.

Activate event on rising edge Lets you enable event generation at low to high transition.

Event delay Lets you specify the delay time in seconds for trigger and interrupt events in the range 0 … 1.34 s.

Note

For periodic events, the event delay should be less than the period. Otherwise, only each first trigger or interrupt event is delayed and becomes effective. The other trigger or interrupt events generated during the event delay are ignored.

Event rate Lets you specify the event rate in the range 1 … 256. It specifies after how many occurrences of any event cause a trigger or interrupt is generated. For example, if you set the event rate to 3, an event will be generated after each third occurrence of an event cause. The value affects the signal of the trigger line as well as the generation of the interrupt.

Trigger line Lets you select a trigger line number to be used for trigger event generation in the range 1 … 16.

Note

It is not allowed to use the same trigger line number in different event-generating blocks.

This setting is enabled only if an event mode with trigger generation is selected.

Channel number Lets you specify the channel number to which event generation should relate. You can select one channel from the channel range that you selected on the Unit page.

Related topics

References

DIO_CLASS1_BIT_OUT_BLx

Purpose

To provide write access to the digital output.

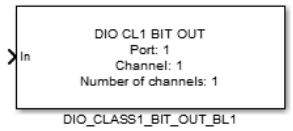
Where to go from here

Information in this section

Block Description (DIO_CLASS1_BIT_OUT_BLx)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To provide write access to the digital output.

Description

The block provides write access to a range of channels of a digital I/O port. You can write to a single channel by setting the number of channels to one. You can write to all channels of the digital I/O port by setting the number of channels to the port width.

I/O mapping

For information on the mapping of channel numbers to the related I/O pins of the MicroLabBox I/O connector, refer to Connector Pinouts (MicroLabBox Hardware Installation and Configuration 🕮).

I/O characteristics

The following table describes the ports of the block:

Port	Description
Input	
In	Specifies the values to be written to the DIO Class 1 output channels. Data type: Array of Booleans. Its length is specified by the number of channels. Range: 0 (= low), 1 (= high)

You can specify the voltage of the *high* level at the **Unit page**.

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (DIO_CLASS1_BIT_IN_BLx) on page 71)
- Initialization page (refer to Initialization Page (DIO_CLASS1_BIT_OUT_BLx) on page 77)
- Termination page (refer to Termination Page (DIO_CLASS1_BIT_OUT_BLx) on page 78)
- Event page (refer to Event Page (DIO_CLASS1_BIT_IN_BLx) on page 72)

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- DioCl1DigOut_create()
- DioCl1DigOut_setSignalVoltage()
- DioCl1DigOut_setInterruptMode()
- DioCl1DigOut_setTrigerMode()
- DioCl1DigOut_setTriggerLineOut()
- DioCl1DigOut_setEventDownsample()
- DioCl1DigOut_setEventDelay()
- DioCl1DigOut_apply()
- DioCl1DigOut_setChMaskOutData()
- DioCl1DigOut_setChMaskOutHighZ()
- DioCl1DigOut_write()
- DioCl1DigOut_start()

Unit Page (DIO_CLASS1_BIT_OUT_BLx)

Purpose

To specify the digital output channels and the voltage for the *High* level.

Dialog settings

Port number Lets you select the port number to be used in the range 1 … 3.

Channel number Lets you select the first channel number to be used in the range 1 … 16.

Note

Concurrent access to the same DIO Class 1 channel by other blocks or functions is not allowed.

Number of channels Lets you select the number of channels to be read. The possible value depends on the width of the digital I/O port and the number of the first channel to be written.

```
NumberOfChannels <= 16 - (ChannelNumber - 1)
```

Note

Concurrent access to the same DIO Class 1 channel by other blocks or functions is not allowed.

Signal connector pin Displays the range of signal connector pins for the selected channels. The range begins with the connector pin of the first selected channel.

Reference connector pin Displays the range of reference connector pins for the selected channels.

High level Lets you set the voltage level of *High* on the output channel to 2.5 V, 3.3 V or 5.0 V.

Related topics

References

Block Description (DIO_CLASS1_BIT_OUT_BLx)..... 75
Event Page (DIO_CLASS1_BIT_OUT_BLx)..... 79
Initialization Page (DIO_CLASS1_BIT_OUT_BLx)..... 77
Termination Page (DIO_CLASS1_BIT_OUT_BLx)..... 78

Initialization Page (DIO_CLASS1_BIT_OUT_BLx)

Purpose

To set the initial output states of the specified channels.

Description

During the model initialization phase, the initial digital output states specified by the **Initial output state** settings are written to all the channels of the specified digital I/O port to ensure a defined output during this simulation phase. This is especially useful if the channels are used in a triggered or enabled subsystem

that is not executed right from the start of the simulation. Before initialization, the digital outputs are set to high impedance.

Dialog settings

Port Displays the port number that you selected on the **Unit** page.

Channel Displays the range of channels that you selected on the **Unit** page.

Initial output state Lets you select the initial output state, *High* or *Low*, for each channel within the range that you specified on the **Unit page**.

Set All Lets you select the initial output state for all channels of the channel range you specified on the **Unit page**.

Related topics

References

Termination Page (DIO_CLASS1_BIT_OUT_BLx)

Purpose

To set the termination output state of the specified channels.

Dialog settings

Port Displays the port number that you selected on the **Unit** page.

Channel Displays the range of channels that you selected on the **Unit** page.

Termination mode To enable or disable the setting of definite output values at the end of the simulation.

Termination Mode Checkbox	Meaning
Disabled	The digital outputs of all the channels of the specified digital I/O port are set to high impedance (high-Z) at the end of the simulation.
Enabled	The output behavior of all the channels of the specified digital I/O port is determined by the **Output** settings (see following table) at the end of the simulation.

The termination mode checkbox is disabled by default.

Output Lets you select the output state of the channels after termination.

Option Button	Meaning
Last output state	Each output channel holds its last digital output value at the end of the simulation.
State on termination	The channels change to the output state that you specified in **Output state on termination**.

Output state on termination Lets you select the definite output state at the end of the simulation. The setting is only enabled if the channel number is within the channel range you specified on the Unit page.

Set All Lets you select the output state on termination for all channels of the channel range you specified on the Unit page.

Related topics

References

Event Page (DIO_CLASS1_BIT_OUT_BLx)

Purpose

To enable event generation for the specified channel.

Description

You can configure interrupt event generation, trigger event generation or both at *falling edge* and/or at *rising edge* transitions on a specific channel.

Dialog settings

Port Displays the port number that you selected on the Unit page.

Channel Displays the range of channels that you selected on the Unit page.

Enable event generation Lets you enable the generation of interrupt or trigger events or both.

If you enable event generation, at least one of the transitions *falling edge* or *rising edge* must be activated.

Event mode Lets you select the events to be generated at the activated transition.

Event Mode	Meaning
Interrupt	Generate interrupt events at the activated transitions.
Trigger	Generate trigger events at the activated transitions.
Interrupt and Trigger	Generate both interrupts and trigger events at the activated transitions.

If you select interrupt generation, your model must contain a related DIO_CLASS1_HWINT_BLx block configured with the same port and channel number.

Activate event on falling edge Lets you enable event generation at high to low transition.

Activate event on rising edge Lets you enable event generation at low to high transition.

Event delay Lets you specify the delay time in seconds for trigger and interrupt events in the range 0 … 1.34 s.

Note

For periodic events, the event delay should be less than the period. Otherwise, only each first trigger or interrupt event is delayed and becomes effective. The other trigger or interrupt events generated during the event delay are ignored.

Event rate Lets you specify the event rate in the range 1 … 256. It specifies after how many occurrences of any event cause a trigger or interrupt is generated. For example, if you set the event rate to 3, an event will be generated after each third occurrence of an event cause. The value affects the signal of the trigger line as well as the generation of the interrupt.

Trigger line Lets you select a trigger line number to be used for trigger event generation in the range 1 … 16.

Note

It is not allowed to use the same trigger line number in different event-generating blocks.

This setting is enabled only if an event mode with trigger generation is selected.

Channel number Lets you specify the channel number to which event generation should relate. You can select one channel from the channel range that you selected on the Unit page.

Related topics

References

Block Description (DIO_CLASS1_BIT_OUT_BLx) 75
DIO_CLASS1_HWINT_BLx 41
Initialization Page (DIO_CLASS1_BIT_OUT_BLx) 77
Termination Page (DIO_CLASS1_BIT_OUT_BLx) 78
Unit Page (DIO_CLASS1_BIT_OUT_BLx) 76

Timing I/O

Introduction

The RTI Blockset of MicroLabBox (RTI1202) provides several blocks that you can use for generating and measuring pulse width modulated square-wave signals (PWM signals) or single pulses. For an overview of which RTI block is appropriate for your use case, refer to Timing I/O (MicroLabBox Features 🕮).

Where to go from here

Information in this section

PWM Signal Generation

Where to go from here

Information in this section

Information in other sections

EMC_BC_PWM_BLx (RTI Electric Motor Control Blockset Reference 🕮)
To generate block-commutated PWM signals to control an electric motor.

EMC_MC_PWM_BLx (RTI Electric Motor Control Blockset Reference 🕮)
To generate a multichannel PWM signal to control an electric motor.

DIO_CLASS1_PWM_BLx

Purpose

To generate a PWM signal on a specified output channel.

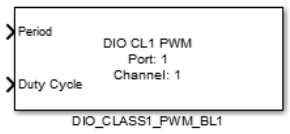
Where to go from here

Information in this section

Block Description (DIO_CLASS1_PWM_BLx)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To generate a PWM signal on the specified output channel.

Description

The period and the duty cycle of the generated PWM signal are adjustable during run time.

I/O mapping

For information on the mapping of channel numbers to the related I/O pins of the MicroLabBox I/O connector, refer to Connector Pinouts (MicroLabBox Hardware Installation and Configuration 🕮).

I/O characteristics

The following table shows the scaling between the duty cycle and the block's Simulink input:

Simulink Input	Duty Cycle
0 … 1	0 … 100%

The following table describes the ports of the block:

Port	Description
Input	
Period	Specifies the period of the PWM signal. Data type: Double Range: 100 ns … 1.34 s Resolution: 10 ns
Duty cycle	Specifies the duty cycle of the PWM signal. Data type: Double Range: 0 … 1 The resolution of the duty cycle depends on the specified period value. The resolution for time intervals within the PWM signal generator unit is 10 ns. If the time interval resulting from the chosen duty cycle can not be represented in multiples of 10 ns the actual value is determined by rounding towards the nearest possible value.

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (DIO_CLASS1_PWM_BLx) on page 84)
- Generation page (refer to Generation Page (DIO_CLASS1_PWM_BLx) on page 85)
- Parameters page (refer to Parameters Page (DIO_CLASS1_PWM_BLx) on page 86)
- Event page (refer to Event Page (DIO_CLASS1_PWM_BLx) on page 87)

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- DioCl1PwmOut_create()
- DioCl1PwmOut_setSignalVoltage()
- DioCl1PwmOut_setUpdateMode()
- DioCl1PwmOut_setInterruptMode()
- DioCl1PwmOut_setTriggerMode()
- DioCl1PwmOut_setTriggerLineOut()
- DioCl1PwmOut_setEventDownsample()
- DioCl1PwmOut_setEventDelay()
- DioCl1PwmOut_apply()
- DioCl1PwmOut_setPeriode()
- DioCl1PwmOut_setDutyCycle()
- DioCl1PwmOut_setOutputHighZ()
- DioCl1PwmOut_write()
- DioCl1PwmOut_start()

Unit Page (DIO_CLASS1_PWM_BLx)

Purpose

To specify the output channel for the PWM signal and the voltage for the *High* level.

Dialog settings

Port number Lets you select the port number to be used in the range 1 … 3.

Channel number Lets you select the channel number in the range 1 … 16.

Note

Concurrent access to the same DIO Class 1 channel by other blocks or functions is not allowed.

Signal connector pin Displays the signal connector pin for the selected channel.

Reference connector pin Displays the reference connector pin for the selected channel.

High level Lets you set the voltage level of *High* on the output channel to 2.5 V, 3.3 V or 5.0 V.

Related topics

References

Generation Page (DIO_CLASS1_PWM_BLx)

Purpose

To specify update and inverting mode of the generated PWM signal.

Dialog settings

Port Displays the port number that you selected on the Unit page.

Channel Displays the channel that you selected on the Unit page.

Update mode Lets you select the PWM update mode for the new values of the period and/or the duty cycle:

Update Mode	Description
Asynchronous	New values for T_{high} and T_{low} are updated immediately. This can result in period and/or duty cycle values that differ from the old *and* the new values for one period.
Synchronous	New values for T_{high} and T_{low} are updated at the next rising edge of the PWM output signal. The output period is constant if $T = T_{high} + T_{low}$ is constant.

For further information on the update mode, refer to PWM Signal Generation (DIO Class 1) (MicroLabBox Features 🕮).

Inverting mode Lets you select the inverting mode for the PWM signal.

Inverting Mode	Description
Not Inverted	The PWM signal is not inverted.
Inverted	The PWM signal is inverted.

Related topics

References

Parameters Page (DIO_CLASS1_PWM_BLx)

Purpose

To specify the initial output behavior and the termination output behavior.

Description

Initialization During the model initialization phase, the PWM output signal is either generated with an initial period or set to constant low or high potential. With **Initial period** and **Initial duty cycle**, the channel has a defined output during this simulation phase. This is especially useful if the channels are used in a triggered or enabled subsystem that is not executed right from the start of the simulation. Before initialization, the digital outputs are set to high impedance.

Termination With the block's **Termination** settings, you can specify an output behavior of the channel on model termination to drive your external hardware into a safe final condition.

The possible output behaviors at the end of the simulation are:

- The output is set to high impedance (high-Z) state.
- The output holds the last duty cycle and period.
- The output is set to a definite duty cycle and period.

Dialog settings

Port Displays the port number that you selected on the **Unit** page.

Channel Displays the channel that you selected on the **Unit** page.

Initial period Lets you enter the period in seconds at the start of the simulation in the range 100 ns … 1.34 s with a resolution of 10 ns.

Initial duty cycle Lets you enter the duty cycle at the start of the simulation in the range 0 … 1. The duty cycle values 0 and 1 yield a constant low and constant high output signal respectively.

Termination mode To enable or disable the setting of a definite output behavior at the end of simulation.

Termination Mode Checkbox	Meaning
Disabled	The output channel is set to high impedance (high-Z) state at the end of simulation.
Enabled	The channel's output behavior is determined by the Output settings (see following table) at the end of the simulation.

The termination mode checkbox is disabled by default.

If the Termination mode is enabled, this setting lets you specify a definite frequency at the end of simulation.

Option Button	Meaning
Last output values	Each output channel holds the last duty cycle and period at the end of simulation.
Specific output values	Lets you set a definite duty cycle and period at the end of simulation.

Period on termination Lets you enter the period of the PWM signal in seconds at the end of the simulation in the range 100 ns … 1.34 s with a resolution of 10 ns.

Duty cycle on termination Lets you enter the duty cycle at the end of the simulation in the range 0 … 1. The duty cycle values 0 and 1 yield a constant low and constant high output signal, respectively.

Related topics

References

Block Description (DIO_CLASS1_PWM_BLx)....83
Event Page (DIO_CLASS1_PWM_BLx)....87
Generation Page (DIO_CLASS1_PWM_BLx)....85
Unit Page (DIO_CLASS1_PWM_BLx)....84

Event Page (DIO_CLASS1_PWM_BLx)

Purpose

To enable event generation for the specified channel.

Description

You can configure interrupt event generation, trigger event generation or both at *falling edge* and/or at *rising edge* transitions on a specific channel.

Dialog settings

Port Displays the port number that you selected on the Unit page.

Channel Displays the channel that you selected on the Unit page.

Enable event generation Lets you enable the generation of interrupt or trigger events or both.

If you enable event generation, at least one of the transitions *falling edge* or *rising edge* must be activated.

Event mode Lets you select the events to be generated at the activated transition.

Event Mode	Meaning
Interrupt	Generate interrupt events at the activated transitions.
Trigger	Generate trigger events at the activated transitions.
Interrupt and Trigger	Generate both interrupts and trigger events at the activated transitions.

If you select interrupt generation, your model must contain a related DIO_CLASS1_HWINT_BLx block configured with the same port and channel number.

Activate event on falling edge Lets you enable event generation at high to low transition.

Activate event on rising edge Lets you enable event generation at low to high transition.

Event delay Lets you specify the delay time in seconds for trigger and interrupt events in the range 0 … 1.34 s.

Note

For periodic events, the event delay should be less than the period. Otherwise, only each first trigger or interrupt event is delayed and becomes effective. The other trigger or interrupt events generated during the event delay are ignored.

Event rate Lets you specify the event rate in the range 1 … 256. It specifies after how many occurrences of any event cause a trigger or interrupt is generated. For example, if you set the event rate to 3, an event will be generated after each third occurrence of an event cause. The value affects the signal of the trigger line as well as the generation of the interrupt.

Trigger line Lets you select a trigger line number to be used for trigger event generation in the range 1 … 16.

Note

It is not allowed to use the same trigger line number in different event-generating blocks.

This setting is enabled only if an event mode with trigger generation is selected.

Related topics

References

PWM Signal Measurement

DIO_CLASS1_PWM2D_BLx

Purpose

To measure a PWM signal on a specified input channel.

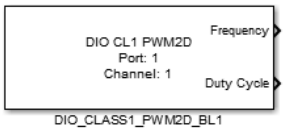
Where to go from here

Information in this section

Block Description (DIO_CLASS1_PWM2D_BLx)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To measure the frequency and the duty cyle of a PWM signal on a specified input channel.

Description

If the maximum input frequency is exceeded or the duty cycle near the maximum frequency is other than 50%, the resulting value of the frequency can not be predicted. If the frequency of the PWM input signal goes below the minimum value the function will output 0.0 Hz as the result.

I/O mapping

For information on the mapping of channel numbers to the related I/O pins of the MicroLabBox I/O connector, refer to Connector Pinouts (MicroLabBox Hardware Installation and Configuration 🕮).

I/O characteristics

The following table shows the scaling between the duty cycle of the measured signal and the block's output in Simulink:

Duty Cycle	Simulink Output
0 … 100%	0 … 1

The following table describes the ports of the block:

Port	Description
Output	
Frequency	Outputs the current frequency. Data type: Double Range: 0.5 Hz … 10 MHz
Duty Cycle	Outputs the current duty cycle. Data type: Double Range: 0 … 1

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (DIO_CLASS1_PWM2D_BLx) on page 92)
- Measurement page (refer to Measurement Page (DIO_CLASS1_PWM2D_BLx) on page 92)
- Event page (refer to Event Page (DIO_CLASS1_PWM2D_BLx) on page 93)

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- DioCl1PwmIn_create()
- DioCl1PwmIn_setUpdateMode()
- DioCl1PwmIn_setTimeout()
- DioCl1PwmIn_setInterruptMode()
- DioCl1PwmIn_setTriggerMode()
- DioCl1PwmIn_setTriggerLineOut()
- DioCl1PwmIn_setEventDownsample()
- DioCl1PwmIn_setEventDelay()
- DioCl1PwmIn_apply()
- DioCl1PwmIn_getFrequency()
- DioCl1PwmIn_getDutyCycle()
- DioCl1PwmIn_read()
- DioCl1PwmIn_start()

Unit Page (DIO_CLASS1_PWM2D_BLx)

Purpose

To specify the input channel for the PWM signal.

Dialog settings

Port number Lets you select the port number to be used in the range 1 … 3.

Channel number Lets you select the channel number in the range 1 … 16.

Note

Concurrent access to the same DIO Class 1 channel by other blocks or functions is not allowed.

Signal connector pin Displays the signal connector pin for the selected channel.

Reference connector pin Displays the reference connector pin for the selected channel.

Related topics

References

Block Description (DIO_CLASS1_PWM2D_BLx) 90
Event Page (DIO_CLASS1_PWM2D_BLx) 93
Measurement Page (DIO_CLASS1_PWM2D_BLx) 92

Measurement Page (DIO_CLASS1_PWM2D_BLx)

Purpose

To specify the updating of the measured values.

Dialog settings

Port Displays the port number that you selected on the Unit page.

Channel Displays the channel that you selected on the Unit page.

PWM Update Lets you select the edge type used to update the measured values.

Edge Type	Description
Both edges	The measured values are updated at each edge of the PWM signal.
Rising edge	The measured values are updated at each rising edge of the PWM signal.
Falling edge	The measured values are updated at each falling edge of the PWM signal.

Timeout limitation Lets you enable or disable a timeout limit for updating the measured values.

Timeout limit Lets you enter a timeout limit in seconds for updating the measured values in the range 50 ms … 1.34 s. Here you can specify the period of time after which the signal is no longer treated as a pulse.

Related topics

References

Block Description (DIO_CLASS1_PWM2D_BLx)....90
Event Page (DIO_CLASS1_PWM2D_BLx)....93
Unit Page (DIO_CLASS1_PWM2D_BLx)....92

Event Page (DIO_CLASS1_PWM2D_BLx)

Purpose

To enable event generation for the specified channel.

Description

You can configure interrupt event generation, trigger event generation or both at *falling edge* and/or at *rising edge* transitions on a specific channel.

Dialog settings

Port Displays the port number that you selected on the Unit page.

Channel Displays the channel that you selected on the Unit page.

Enable event generation Lets you enable the generation of interrupt or trigger events or both.

If you enable event generation, at least one of the transitions *falling edge* or *rising edge* must be activated.

Event mode Lets you select the events to be generated at the activated transition.

Event Mode	Meaning
Interrupt	Generate interrupt events at the activated transitions.
Trigger	Generate trigger events at the activated transitions.
Interrupt and Trigger	Generate both interrupts and trigger events at the activated transitions.

If you select interrupt generation, your model must contain a related DIO_CLASS1_HWINT_BLx block configured with the same port and channel number.

Activate event on falling edge Lets you enable event generation at high to low transition.

Activate event on rising edge Lets you enable event generation at low to high transition.

Event delay Lets you specify the delay time in seconds for trigger and interrupt events in the range 0 … 1.34 s.

Note

For periodic events, the event delay should be less than the period. Otherwise, only each first trigger or interrupt event is delayed and becomes effective. The other trigger or interrupt events generated during the event delay are ignored.

Event rate Lets you specify the event rate in the range 1 … 256. It specifies after how many occurrences of any event cause a trigger or interrupt is generated. For example, if you set the event rate to 3, an event will be generated after each third occurrence of an event cause. The value affects the signal of the trigger line as well as the generation of the interrupt.

Trigger line Lets you select a trigger line number to be used for trigger event generation in the range 1 … 16.

Note

It is not allowed to use the same trigger line number in different event-generating blocks.

This setting is enabled only if an event mode with trigger generation is selected.

Related topics

References

Block Description (DIO_CLASS1_PWM2D_BLx)....90
DIO_CLASS1_HWINT_BLx....41
Measurement Page (DIO_CLASS1_PWM2D_BLx)....92
Unit Page (DIO_CLASS1_PWM2D_BLx)....92

Pulse Generation

DIO_CLASS1_PULSE_BLx

Purpose

To generate a pulse signal of a specified width on the specified output channel.

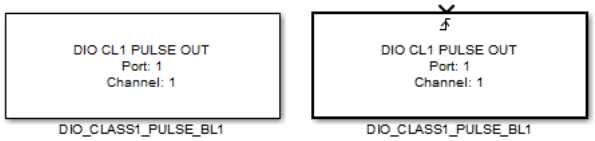
Where to go from here

Information in this section

Block Description (DIO_CLASS1_PULSE_BLx)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To generate a pulse of a specified width on the specified output channel.

Description

The pulse generation can be triggered during run time via a trigger port or via a trigger line.

I/O mapping

For information on the mapping of channel numbers to the related I/O pins of the MicroLabBox I/O connector, refer to Connector Pinouts (MicroLabBox Hardware Installation and Configuration 🕮).

I/O characteristics

The following table describes the ports of the block:

Port	Description
Input	
Trigger input	Controls the pulse generation. Available only if trigger pulse generation by trigger port is selected on the **Trigger** page.

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (DIO_CLASS1_PULSE_BLx) on page 96)
- Generation page (refer to Generation Page (DIO_CLASS1_PULSE_BLx) on page 97)
- Trigger page (refer to Trigger Page (DIO_CLASS1_PULSE_BLx) on page 98)

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- DioCl1PulseOut_create()
- DioCl1PulseOut_setSignalVoltage()
- DioCl1PulseOut_setPulseWidth()
- DioCl1PulseOut_setInvertingMode()
- DioCl1PulseOut_setTriggerLineIn()
- DioCl1PulseOut_apply()
- DioCl1PulseOut_setOutputHighZ()
- DioCl1PulseOut_FirePulse()
- DioCl1PulseOut_write()
- DioCl1PulseOut_start()

Unit Page (DIO_CLASS1_PULSE_BLx)

Purpose

To specify the output channel and the voltage of the *High* level.

Dialog settings

Port number Lets you select the port number to be used in the range 1 … 3.

Channel number Lets you select the channel number in the range 1 … 16.

Note

Concurrent access to the same DIO Class 1 channel by other blocks or functions is not allowed.

Signal connector pin Displays the signal connector pin for the selected channel.

Reference connector pin Displays the reference connector pin for the selected channel.

High level Lets you set the voltage level of *High* on the output channel to 2.5 V, 3.3 V or 5.0 V.

Related topics

References

Generation Page (DIO_CLASS1_PULSE_BLx)

Purpose

To specify pulse width and inverting mode of the generated pulse.

Dialog settings

Port Displays the port number that you selected on the Unit page.

Channel Displays the channel that you selected on the Unit page.

Pulse width Lets you enter the pulse width for the generated pulse signal in seconds in the range 100 ns … 0.5 s with a resolution of 10 ns.

Inverting mode Lets you select the inverting mode for the generated pulse signal.

Inverting Mode	Description
Not Inverted	The voltage changes from *low* to *high* for the duration of the pulse width.
Inverted	During the initialization phase the default voltage level at the specified output channel is switched to *high*. At pulse generation the voltage changes from *high* to *low* for the duration of the pulse width.

Note

After termination of the simualtion, the channel is set to high impedance (high-Z).

Related topics

References

Trigger Page (DIO_CLASS1_PULSE_BLx)

Purpose

To specify the triggering of the pulse generation.

Description

The generation of the pulse signal can be triggered via an optional trigger port of the DIO_CLASS1_PULSE_BLx block or via a trigger line.

Dialog setting

Port Displays the port number that you selected on the Unit page.

Channel Displays the channel that you selected on the Unit page.

Trigger setup Lets you select how the pulse signal generation is triggered:

Option Button	Meaning
Trigger pulse generation by trigger port	The pulse signal generation is triggered via the block's trigger port. You can specify how the triggering is activated in Trigger type.
Trigger pulse generation by trigger line	The pulse signal generation is triggered via the line specified in Trigger line.

Trigger type Lets you specify how triggering via the block's trigger port is activated.

Trigger Type	Meaning
Rising edge	Pulse generation is triggered at each low to high transition.
Falling edge	Pulse generation is triggered at each high to low transition.
Both edges	Pulse generation is triggered at each transition.
Function-call	Pulse generation is triggered by a function call.

This setting is only enabled, if *Trigger pulse generation by trigger port* is selected as Trigger setup.

Trigger line Lets you select a trigger line number in the range 1 … 16.

This setting is only enabled, if *Trigger pulse generation by trigger line* is selected as Trigger setup.

Related topics

References

Pulse Width Measurement

DIO_CLASS1_PW2D_BLx

Purpose

To measure the pulse width of a pulse signal on a specified input channel.

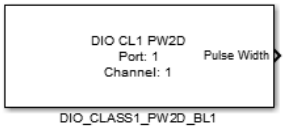
Where to go from here

Information in this section

Block Description (DIO_CLASS1_PW2D_BLx)

Block

Gives you information about the appearance and purpose of the block.



Purpose

Lets you measure the width of a pulse signal on the specified input channel.

I/O mapping

For information on the mapping of channel numbers to the related I/O pins of the MicroLabBox I/O connector, refer to Connector Pinouts (MicroLabBox Hardware Installation and Configuration 🕮).

I/O characteristics

The following table describes the ports of the block:

Port	Description
Output	
Pulse width	Outputs the current pulse width. Data type: Double Unit: Seconds Range: 50 ns … 1.34 s

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (DIO_CLASS1_PW2D_BLx) on page 101)
- Measurement page (refer to Measurement Page (DIO_CLASS1_PW2D_BLx) on page 102)
- Event page (refer to Event Page (DIO_CLASS1_PW2D_BLx) on page 103)

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- DioCl1PulseIn_create()
- DioCl1PulseIn_setEdgePolarity()
- DioCl1PulseIn_setWidthMax()
- DioCl1PulseIn_setInterruptMode()
- DioCl1PulseIn_setTriggerMode()
- DioCl1PulseIn_setTriggerLineOut()
- DioCl1PulseIn_setEventDelay()
- DioCl1PulseIn_apply()
- DioCl1PulseIn_getPulse()
- DioCl1PulseIn_read()
- DioCl1PulseIn_start()

Unit Page (DIO_CLASS1_PW2D_BLx)

Purpose

To specify the input channel for the pulse signal.

Dialog settings

Port number Lets you select the port number to be used in the range 1 … 3.

Channel number Lets you select the channel number in the range 1 … 16.

Note

Concurrent access to the same DIO Class 1 channel by other blocks or functions is not allowed.

Signal connector pin Displays the signal connector pin for the selected channel.

Reference connector pin Displays the reference connector pin for the selected channel.

Related topics

References

Measurement Page (DIO_CLASS1_PW2D_BLx)

Purpose

To specify pulse polarity and pulse width limitations of the measured values.

Dialog settings

Port Displays the port number that you selected on the Unit page.

Channel Displays the channel that you selected on the Unit page.

Polarity mode Lets you select whether to use the high or the low level of the signal for measuring the pulse width.

Mode	Description
High pulse	The duration between a rising edge and the next falling edge (high time) of the square-wave input signal is used as the pulse width.
Low pulse	The duration between a falling edge and the next rising edge (low time) of the square-wave input signal is used as the pulse width.

Pulse width limitation Lets you enable or disable the setting of the Max pulse width.

Max pulse width Lets you enter a maximum pulse width in seconds for updating the measured values in the range 50 ms … 1.34 s. This specifies the amount of time after which the measured value of the pulse width is updated at the latest. If a measured pulse exceeds this time, the maximum pulse width is returned in the Pulse Width outport.

Related topics

References

Event Page (DIO_CLASS1_PW2D_BLx)

Purpose

To enable event generation for the specified channel.

Description

You can configure interrupt event generation, trigger event generation, or both to signal *Data ready* when a new measured value for the pulse width is available.

Dialog settings

Port Displays the port number that you selected on the Unit page.

Channel Displays the channel that you selected on the Unit page.

Enable event generation Lets you enable generation of trigger events or interrupts.

Event mode Lets you select the events to be generated.

Event Mode	Meaning
Interrupt	Generate interrupt events at the activated transitions.
Trigger	Generate trigger events at the activated transitions.
Interrupt and Trigger	Generate both interrupts and trigger events at the activated transitions.

If you select interrupt generation, your model must contain a related DIO_CLASS1_HWINT_BLx block configured with the same port and channel number and *Data ready* as the interrupt type.

Event delay Lets you specify the delay time in seconds for trigger and interrupt events in the range 0 … 1.34 s.

Trigger line Lets you select a trigger line number to be used for trigger event generation in the range 1 … 16.

Note

It is not allowed to use the same trigger line number in different event-generating blocks.

This setting is enabled only if an event mode with trigger generation is selected.

Related topics

References

Nonvolatile Data Handling (NVDATA)

Introduction

Providing information on the RTI blocks used for implementing nonvolatile data handling.

Where to go from here

Information in this section

Information in other sections

Nonvolatile Data Handling (NVDATA) (MicroLabBox Features 🕮)
MicroLabBox provides access to the board's nonvolatile memory by implementing the access in a real-time application or by using the board's web interface.

Using the Web Interface for Nonvolatile Data Handling (MicroLabBox Features 🕮)
The web interface of your MicroLabBox provides a configuration page which lets you manage the data sets stored in the board's nonvolatile memory.

General Information on the NVDATA Blockset

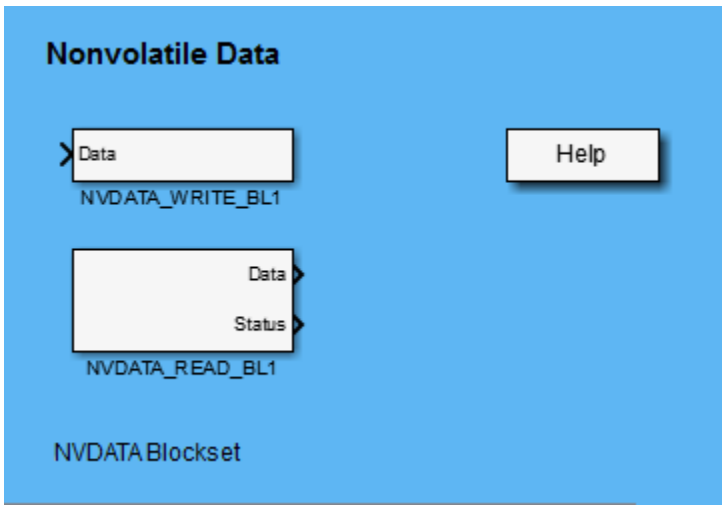
Introduction

To give you basic information on the RTI NVDATA Blockset.

Overview of the NVDATA Blockset

Access

Double-click the Nonvolatile Data button in the Base Blockset to display the NVDATA Blockset.



You can use the NVDATA blocks to implement access to the board's nonvolatile memory.

To access the nonvolatile memory, you always have to use a block pair consisting of an NVDATA_WRITE_BLx block and an NVDATA_READ_BLx block.

NVDATA_READ_BLx

Purpose

To read a data set from the board's nonvolatile memory.

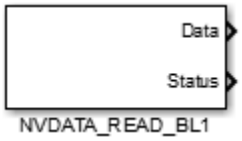
Where to go from here

Information in this section

Information in other sections

Block Description (NVDATA_READ_BLx)

Block



Purpose

To read a data set from the board's nonvolatile memory.

Description

In the block's Data tag setting, specify the name of the data set you want to read. The NVDATA_READ_BLx block must have the same settings as the NVDATA_WRITE_BLx block that wrote the data set.

You can use up to 64 block pairs (NVDATA_READ_BLx and NVDATA_WRITE_BLx) within a model.

I/O characteristics

The following table describes the ports of the block:

Port	Description
Output	
Data	Outputs the specified data set. The name of the port corresponds to the name that you specified in the **Data tag** setting. Data type: Depends on the **Data type** setting Width: Depends on the **Number of elements** setting
Status	Indicates whether the data set that you want to read is valid. There must be one initial write access to the data set before a read access will be valid. Data type: UInt32 Range: 0, 1 ▪ 0: Data set has not been previously written, i.e., it is still uninitialized. ▪ 1: Data set has been written at least once. The **Status** port allows you to specify initial default values of a data set.

Dialog pages

You can specify the dialog settings on the Parameters page (refer to Parameters Page (NVDATA_READ_BLx) on page 108).

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

This RTI block is implemented using the following RTLib functions:

- NvData_read

Parameters Page (NVDATA_READ_BLx)

Purpose

To specify the data set to be read from the board's nonvolatile memory.

Dialog settings

Data tag Lets you specify a unique name for the data set. The name is used for the **Data** port and can consist of up to 63 characters.

Note

A valid data set name is a character string of letters, digits, and underscores. There are the following naming restrictions for the data set name:

- The first character must be a letter.
- The name must not be a keyword, such as `while` or `if`.

Number of elements Lets you specify the number of elements in the data set in the range 1 … 64.

Data type Lets you specify the data type of the elements contained in the data set. You can specify only one data type for the entire data set.

Data Type	Meaning
Int8	8-bit integer values Allocates 1 byte
UInt8	8-bit integer values (unsigned) Allocates 1 byte
Int16	16-bit integer values Allocates 2 bytes
UInt16	16-bit integer values (unsigned) Allocates 2 bytes
Int32	32-bit integer values Allocates 4 bytes
UInt32	32-bit integer values (unsigned) Allocates 4 bytes
Single (Float32)	32-bit float values Allocates 4 bytes
Double (Float64)	64-bit float values Allocates 8 bytes

Related topics

References

Block Description (NVDATA_READ_BLx)..... 107

NVDATA_WRITE_BLx

Purpose

To write a data set to the board's nonvolatile memory.

Where to go from here

Information in this section

Block Description (NVDATA_WRITE_BLx)..... 110
To describe the purpose and function of the block.

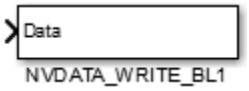
Parameters Page (NVDATA_WRITE_BLx)..... 111
To specify the data set to be written to the board's nonvolatile memory.

Information in other sections

NVDATA_READ_BLx..... 107
To read a data set from the board's nonvolatile memory.

Block Description (NVDATA_WRITE_BLx)

Block



Purpose

To write a data set to the board's nonvolatile memory.

Description

The data set is identified by its name, which is specified by the block's Data tag setting. The data tag must therefore be unique in the model.

For each NVDATA_WRITE_BLx block there must be a NVDATA_READ_BLx block with the same settings.

You can use up to 64 block pairs (NVDATA_WRITE_BLx and NVDATA_READ_BLx) within a model.

I/O characteristics

The following table describes the ports of the block:

Port	Description
Input	
Data	Reads the specified data set. The name of the port corresponds to the name that you specified in the **Data tag** setting. Data type: Depends on the **Data type** setting Width: Depends on the **Number of elements** setting

Dialog pages

The dialog settings can be specified on the Parameters page (refer to Parameters Page (NVDATA_WRITE_BLx) on page 111).

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- NvData_create
- NvData_createDataSet
- NvData_setName
- NvData_setType
- NvData_setDimension
- NvData_apply
- NvData_write

Parameters Page (NVDATA_WRITE_BLx)

Purpose

To specify the data set to be written to the board's nonvolatile memory.

Dialog settings

Data tag Lets you specify a unique name for the data set. The name is used for the **Data** port and can consist of up to 63 characters.

Note

A valid data set name is a character string of letters, digits, and underscores. There are the following naming restrictions for the data set name:

- The first character must be a letter.
- The name must not be a keyword, such as `while` or `if`.

Number of elements Lets you specify the number of elements in the data set in the range 1 … 64.

Data type Lets you specify the data type of the elements contained in the data set. You can specify only one data type for the entire data set.

Data Type	Meaning
Int8	8-bit integer values Allocates 1 byte
UInt8	8-bit integer values (unsigned) Allocates 1 byte
Int16	16-bit integer values Allocates 2 bytes
UInt16	16-bit integer values (unsigned) Allocates 2 bytes
Int32	32-bit integer values Allocates 4 bytes
UInt32	32-bit integer values (unsigned) Allocates 4 bytes
Single (Float32)	32-bit float values Allocates 4 bytes
Double (Float64)	64-bit float values Allocates 8 bytes

Related topics

References

Block Description (NVDATA_WRITE_BLx)..... 110

Serial Interface

Where to go from here

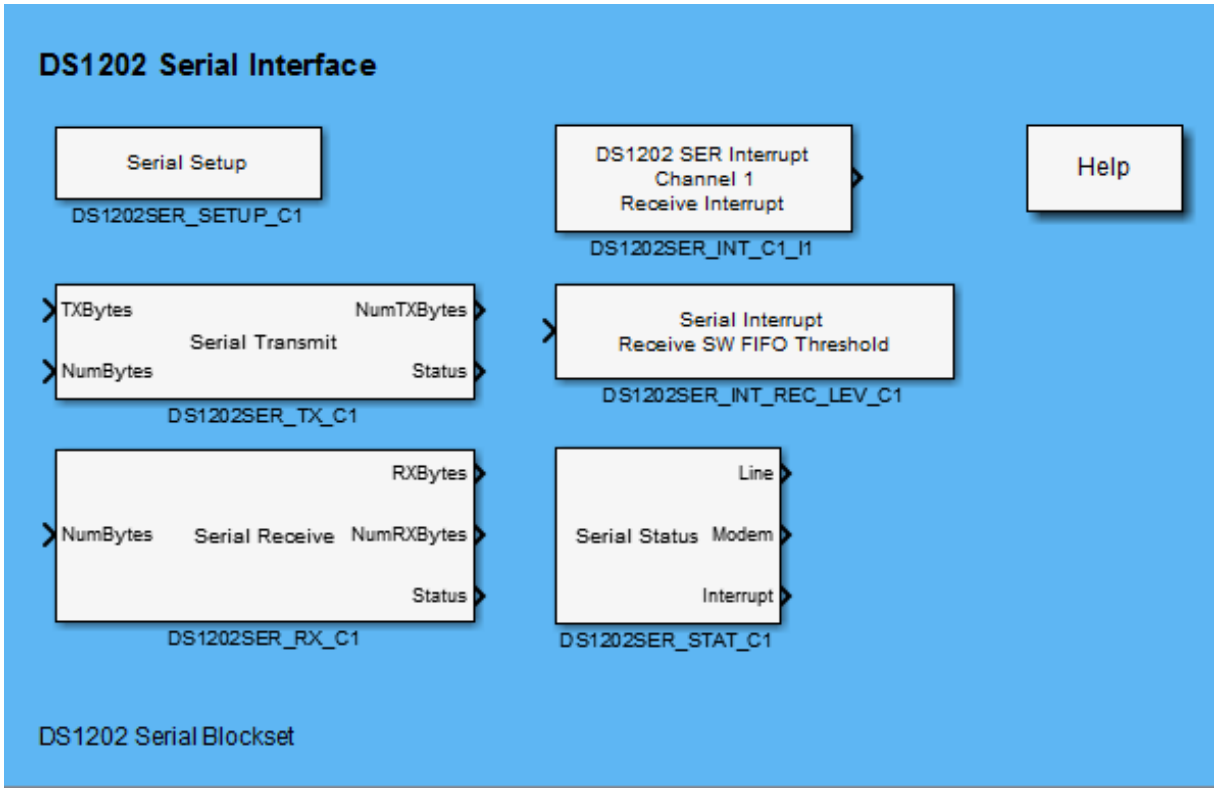
Information in this section

General Information on the Serial Interface

Overview of the Serial Interface

Introduction

After you double-click the Serial Interface button in the DS1202 MicroLabBox FPGA I/O Type 1 sublibrary, the rti1202serlib is displayed.



The Serial Interface blocks can be used to implement serial communication.

Basic principles

Refer to Basic Principles of Serial Communication with RTI Blocks on page 116.

Note

Although the serial blocks of different boards are almost the same, you must always use the board-specific serial blocks.

Library components

The library contains the following RTI blocks:

- DS1202SER_SETUP_Cx on page 118
- DS1202SER_STAT_Cx on page 123
- DS1202SER_TX_Cx on page 127
- DS1202SER_RX_Cx on page 131
- DS1202SER_INT_Cx_Iy on page 135
- DS1202SER_INT_REC_LEV_Cx on page 138

Basic Principles of Serial Communication with RTI Blocks

Purpose

This section explains the basic principles of serial communication with RTI blocks.

Where to go from here

Information in this section

Basics on the Buffer Used for Serial Communication..... 116

Information in other sections

Serial Interface (MicroLabBox Features 🕮)
MicroLabBox provides a serial interface that can be used to implement a serial communication.

Basics on the Buffer Used for Serial Communication

Software FIFO buffer

A software FIFO buffer is installed between your model and the UART. The buffer is a memory that provides the UART with additional space for data storage and ensures that the generic blocks are hardware-independent.

The software FIFO buffer stores data that will be written to the UART (transmit buffer) or that was read by it (receive buffer).

Transmit buffer

To transmit data, you only have to write it to the transmit buffer (TX SW FIFO) with the DS1202SER_TX_Cx block. The data is then transmitted via the UART.

Receive buffer

Data that is received via the serial interface is first copied to the UART buffer. When the number of received bytes exceeds the UART threshold or when the UART timeout is triggered, the bytes are copied to the receive buffer.

UART threshold The UART threshold is defined in the DS1202SER_SETUP_Cx block.

UART timeout The UART timeout is triggered when no signal is received during an interval of 4 signals after the last signal. The time value depends on the number of bits per signal and the baud rate. The worst case is a signal with 12 bits (1 start bit, 8 data bits, 1 parity bit, and 2 stop bits) and a baud rate of 300 baud. In this case the timeout is 160 ms after the last signal is received.

To get the data into your model, use the DS1202SER_RX_Cx block. It reads the data from the receive buffer and copies it to an outport. To get a trigger signal when the receive buffer contains data, use the DS1202SER_INT_Cx_Iy block.

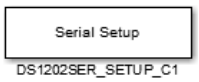
DS1202SER_SETUP_Cx

Where to go from here

Information in this section

Block Description (DS1202SER_SETUP_Cx)

Block



Purpose

To set the global parameters for the serial interface.

Note

- This block has to be placed in the model if any of the other serial blocks is used for the corresponding board.
- This block must not be used more than once per channel.

I/O mapping

For information on the I/O mapping, refer to Serial Interface (MicroLabBox Features 🕮).

Dialog pages

The dialog settings can be specified on the following pages:

- Unit Page (refer to Unit Page (DS1202SER_SETUP_Cx) on page 119)
- UART Page (refer to UART Page (DS1202SER_SETUP_Cx) on page 120)

- **FIFO Page** (refer to FIFO Page (DS1202SER_SETUP_Cx) on page 121)
- **Advanced Page** (refer to Advanced Page (DS1202SER_SETUP_Cx) on page 122)

Related RTLib functions

This RTI block is implemented using the following RTLib functions:

- dsser_init
- dsser_config
- dsser_set

Related topics

References

Unit Page (DS1202SER_SETUP_Cx)

Purpose

To select the channel number.

Dialog settings

Channel number Lets you choose the number of the channel in the range 1 … 2.

Related topics

References

UART Page (DS1202SER_SETUP_Cx)

Purpose

To specify the UART parameters.

Dialog settings

Transceiver Lets you select the transceiver mode:

Transceiver Mode	Meaning
RS232	RS232 mode
RS422	RS422 mode
RS485	RS485 mode

Baud rate Lets you specify the baud rate in bits per second.

Mode	Baud Rate Range
RS232	5 … 230,400 baud
RS422	5 … 10,000,000 baud
RS485	5 … 10,000,000 baud

For further information, refer to Serial Interface (MicroLabBox Features 🕮).

Data bits Lets you choose the number of data bits. The valid values are: 5, 6, 7, 8.

Stop bits Lets you choose the number of stop bits. The valid values are: 1, 1.5 or 2. If you select 1.5 or 2, the number of stop bits depends on the number of specified data bits: For 5 data bits there are 1.5 stop bits; for 6, 7 and 8 data bits there are 2 stop bits.

Parity Lets you choose the parity mode:

Parity Mode	Meaning
No	No parity bits
Odd	Parity bit is set so that there is an odd number of "1" bits in the byte, including the parity bit
Even	Parity bit is set so that there is an even number of "1" bits in the byte, including the parity bit

Parity Mode	Meaning
Forced parity one	Parity bit is forced to a logical 1
Forced parity zero	Parity bit is forced to a logical 0

Copy data to RX SW FIFO after reception of <value> byte(s) at latest Lets you choose the UART threshold at which data is copied from the UART to the receive buffer. Values are: 1, 4, 8, 14.

Note

Use the highest UART threshold possible to generate fewer interrupts, i.e., to decrease the UART's workload.

Enable RTS/CTS mode Lets you enable a hardware handshake (RTS/CTS).

Related topics

References

Advanced Page (DS1202SER_SETUP_Cx) 122
Block Description (DS1202SER_SETUP_Cx) 118
FIFO Page (DS1202SER_SETUP_Cx) 121
Unit Page (DS1202SER_SETUP_Cx) 119

FIFO Page (DS1202SER_SETUP_Cx)

Purpose

To specify the software FIFO buffer.

Dialog settings

SW FIFO size Lets you specify the size of the software buffer. The size must be a power of two (2^n) and at least 64 bytes great. The maximum size depends on the available memory.

Overwrite mode Lets you choose the behavior of the receive buffer when an overrun occurs:

Overwrite Mode	Meaning
Discard new data	If the receive buffer is full, the new data is discarded.
Replace old data with FIFO method	If the receive buffer is full, the new data replaces the oldest data in the buffer. The number of bytes that are replaced is defined by Block size.

Block size Lets you specify the number of bytes that are deleted in RX SW FIFO overrun (see table above). Use this parameter to set up the appropriate data consistency for your model. Value range: 1 … (`SW FIFO size`-1)

Related topics

References

Advanced Page (DS1202SER_SETUP_Cx) 122
Block Description (DS1202SER_SETUP_Cx) 118
UART Page (DS1202SER_SETUP_Cx) 120
Unit Page (DS1202SER_SETUP_Cx) 119

Advanced Page (DS1202SER_SETUP_Cx)

Purpose

To specify the behavior on model termination.

Dialog settings

Disable UART on termination Lets you choose the UART behavior on model termination. If the UART is disabled, data is neither transmitted nor received. No interrupts are generated in this case.

Related topics

References

Block Description (DS1202SER_SETUP_Cx) 118
FIFO Page (DS1202SER_SETUP_Cx) 121
UART Page (DS1202SER_SETUP_Cx) 120
Unit Page (DS1202SER_SETUP_Cx) 119

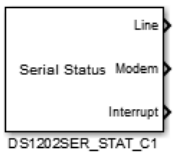
DS1202SER_STAT_Cx

Where to go from here

Information in this section

Block Description (DS1202SER_STAT_Cx)

Block



Purpose

To read the contents of the UART status register.

Note

This block can only be used in interrupt-driven subsystems (see DS1202SER_INT_Cx_Iy on page 135).

- The Line status delivers correct results only if the block resides in a subsystem driven by the Line status interrupt.
- The Modem status delivers correct results only if the block resides in a subsystem driven by the Modem status interrupt.
- The Interrupt status is non-functional at the moment.

Description

The block reads the line, modem and interrupt statuses and writes the values to the outports. If you do not want to evaluate a status register, you can disable its outport with the block dialog.

I/O mapping

For information on the I/O mapping, refer to Serial Interface (MicroLabBox Features 🕮).

I/O characteristics

The outports show the values of the UART's register.

- The **Line** port outputs the 8 bits of the line status register. The following table shows the meanings of the individual bits:

Index	Meaning
1	Data ready (DR) indicator
2	Overrun error (OE) indicator
3	Parity error (PE) indicator
4	Framing error (FE) indicator
5	Break interrupt (BI) indicator
6	Transmitter holding register empty (THRE) indicator
7	Transmitter empty (TEMT) indicator
8	Error in receiver FIFO

- The **Modem** port outputs the 8 bits of the modem status register. The following table shows the meanings of the individual bits:

Index	Meaning
1	Clear-to-send (CTS) changed state
2	Data-set-ready (DSR) changed state
3	Ring-indicator (RI) changed state
4	Data-carrier-detect (DCD) changed state
5	Complement of CTS
6	Complement of DSR
7	Complement of RI
8	Complement of DCD

- The **Interrupt** port outputs the 8 bits of the interrupt status register. The following table shows the meanings of the individual bits:

Index	Meaning
1	Interrupt status: 0 if interrupt pending
2	Interrupt ID bit 1
3	Interrupt ID bit 2
4	Interrupt ID bit 3
5	Not relevant
6	Not relevant
7	FIFOs enabled (bit 0)
8	FIFOs enabled (bit 1)

- The following table shows the characteristics of the block outputs:

Port	Characteristics	Value
Line	Datatype	Boolean
	Range	0, 1
	Size	8
Modem	Datatype	Boolean
	Range	0, 1
	Size	8
Interrupt	Datatype	Boolean
	Range	0, 1
	Size	8

Dialog pages

The dialog settings can be specified on the following pages:

- **Unit Page** (refer to Unit Page (DS1202SER_STAT_Cx) on page 125)
- **Status Page** (refer to Status Page (DS1202SER_STAT_Cx) on page 126)

Related RTLib functions

This RTI block is implemented using the following RTLib function:

- dsser_status_read

Related topics

References

dsser_status_read (MicroLabBox RTLib Reference 🕮)
Status Page (DS1202SER_STAT_Cx)..... 126
Unit Page (DS1202SER_STAT_Cx)..... 125

Unit Page (DS1202SER_STAT_Cx)

Purpose

To specify the channel number used for reading the status.

Dialog settings

Channel number Lets you choose the number of the channel within the range 1 … 2.

Related topics

References

Block Description (DS1202SER_STAT_Cx)..... 123

Status Page (DS1202SER_STAT_Cx)

Purpose

To enable the status registers to be read.

Dialog settings

Enable Line status port Lets you enable the line status output of the UART.

Enable Modem status port Lets you enable the modem status output of the UART.

Enable Interrupt status port Lets you enable the interrupt status output of the UART.

Related topics

References

Block Description (DS1202SER_STAT_Cx)..... 123
Unit Page (DS1202SER_STAT_Cx)..... 125

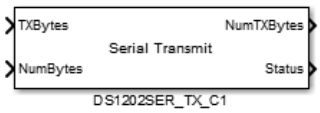
DS1202SER_TX_Cx

Where to go from here

Information in this section

Block Description (DS1202SER_TX_Cx)

Block



Purpose

To send data via the serial interface.

Description

The block sends the bytes of the TXBytes input via the serial interface during one sample step. The number of bytes to be sent can be either fixed or variable. If the number of bytes to be sent is fixed, you have to specify it with a block parameter. If the number of bytes to be sent is variable, you can specify it with either a block parameter or an inport. The status and the number of bytes that were sent are returned via outports.

You can disable the NumBytes input, NumTXBytes output and Status output with the block dialog.

I/O mapping

For information on the I/O mapping, refer to Serial Interface (MicroLabBox Features 🕮).

I/O characteristics

- The TXBytes input must be the stream of bytes to be written to the software buffer within one sample step.

- The **NumBytes** input must be the number of bytes to be sent within one sample step. The value must be less than or equal to the **Maximum number of bytes** block parameter. If it is less, only the specified number of bytes is sent.
- The **NumTXBytes** port outputs the number of bytes that could be written to the software buffer within the current sample step. You can use this output value and the **NumTXBytes** input to verify whether all the data could be sent.
- The **Status** port outputs the status of writing data to the software buffer within the current sample step. One of the following values is returned:

Return Value	Meaning
0	No error
202	The FIFO is filled or not all data could be copied to the FIFO

- The following table shows the characteristics of the block inputs and outputs:

Port	Characteristics	Value
TXBytes	Datatype	UInt8
	Range	0 … 255
	Size	1 … (SW FIFO size - 1)
NumBytes	Datatype	UInt32
	Range	1 … (SW FIFO size - 1)
NumTXBytes	Datatype	UInt32
	Range	1 … (SW FIFO size - 1)
Status	Datatype	Int32
	Range	int32

SW FIFO size is a block parameter. For further information, refer to DS1202SER_SETUP_Cx on page 118.

Dialog pages

The dialog settings can be specified on the following pages:

- **Unit Page** (refer to Unit Page (DS1202SER_TX_Cx) on page 129)
- **Tx Parameters Page** (refer to TX Parameters Page (DS1202SER_TX_Cx) on page 129)
- **Advanced Page** (refer to Advanced Page (DS1202SER_TX_Cx) on page 130)

Related RTLib functions

This RTI block is implemented using the following RTLib function:

- dsser_transmit

Related topics

References

Unit Page (DS1202SER_TX_Cx)

Purpose

To specify the channel number used for sending data.

Dialog settings

Channel number Lets you choose the number of the channel within the range 1 … 2.

Related topics

References

TX Parameters Page (DS1202SER_TX_Cx)

Purpose

To specify the transmitting parameters.

Dialog settings

Transmission SW FIFO mode Lets you specify how to react if there is not enough free space in the transmit buffer:

Data Handling	Meaning
Discard all new data	All data in the sample step is discarded. Data consistency is ensured but you have to repeat the complete data from this sample step.
Write as much data as possible	The transmit buffer is filled until it is full. You only have to repeat bytes which did not fit into the transmit buffer.

Parameter flexibility Lets you specify whether the number of bytes to be sent is fixed (non-tunable) or variable (tunable).

Number of bytes Lets you specify the number of bytes to be sent within one sample step.

Maximum number of bytes Lets you specify the maximum number of bytes that can be sent within one sample step. The valid value range is:
1 … (SW FIFO size-1) (SW FIFO size is a block parameter, see DS1202SER_SETUP_Cx on page 118).

Specify the number of bytes Lets you specify whether to set the number of bytes to be sent within one sample step via the **NumBytes** inport or the block parameter.

Related topics

References

Advanced Page (DS1202SER_TX_Cx)

Purpose

To specify the output.

Dialog settings

Enable NumTXBytes port Lets you specify whether to output the number of bytes that could be sent or not.

Enable Status port Lets you specify whether to output the transmission status or not.

Related topics

References

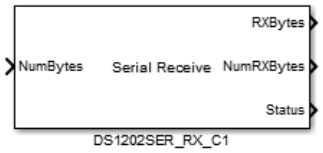
DS1202SER_RX_Cx

Where to go from here

Information in this section

Block Description (DS1202SER_RX_Cx)

Block



Purpose

To read bytes from the serial interface.

Description

The block receives bytes via a serial interface and writes them to the RXBytes output. The number of bytes to be received can be either fixed or variable. If the number of bytes to be received is fixed, you have to specify it with a block parameter. If the number of bytes to be received is variable, you can specify it with either a block parameter or an inport. The status and the number of received bytes are returned via outports.

You can disable the NumBytes input, NumRXBytes output and Status output with the block dialog.

Note

The run-time code of the block is not generated in `MdlOutputs()` but in `rti_mdl_sample_input()`. If this block is placed in an enabled subsystem, received data is therefore read from the RX SW FIFO even if the Enable input signal of the subsystem is 0.

I/O mapping

For information on the I/O mapping, refer to Serial Interface (MicroLabBox Features 🕮).

I/O characteristics

- The **NumBytes** input must be the number of bytes to be read from the software buffer within one sample step.
- The **RXBytes** port outputs the stream of data that could be read from the software buffer within one sample step. If fewer than the expected number of bytes could be received, the last bytes of the output still contain the data from the previous sample step.
- The **NumRXBytes** port outputs the number of bytes that could be read from the software buffer within one sample step.
- The **Status** port outputs the reception status. One of the following values is returned:

Return Value	Meaning
0	No error
4	The operation failed with no effect on the input or output data. No data is written to or read from the FIFO.
5	No new data is read from the FIFO.
202	The FIFO is filled or not all data could be copied to the FIFO.

- The following table shows the characteristics of the block input and outputs:

Port	Characteristics	Value
NumBytes	Datatype	UInt32
	Range	1 … (SW FIFO size - 1)
RXBytes	Datatype	UInt8
	Range	0 … 255
	Size	1 … (SW FIFO size - 1)
NumRXBytes	Datatype	UInt32
	Range	1 … (SW FIFO size - 1)
Status	Datatype	Int32
	Range	Int32

SW FIFO size is a block parameter. For further information, refer to DS1202SER_SETUP_Cx on page 118.

Dialog pages

The dialog settings can be specified on the following pages:

- **Unit Page** (refer to Unit Page (DS1202SER_RX_Cx) on page 133)
- **RX Parameters Page** (refer to RX Parameters Page (DS1202SER_RX_Cx) on page 133)
- **Advanced Page** (refer to Advanced Page (DS1202SER_RX_Cx) on page 134)

Related RTLib functions

This RTI block is implemented using the following RTLib functions:

- dsser_receive
- dsser_receive_term

Related topics

References

Advanced Page (DS1202SER_RX_Cx) 134
DS1202SER_SETUP_Cx 118
dsser_receive (MicroLabBox RTLib Reference 🕮)
dsser_receive_term (MicroLabBox RTLib Reference 🕮)
RX Parameters Page (DS1202SER_RX_Cx) 133
Unit Page (DS1202SER_RX_Cx) 133

Unit Page (DS1202SER_RX_Cx)

Purpose

To specify the channel number used for reading data.

Dialog settings

Channel number Lets you choose the number of the channel in the range 1 … 2.

Related topics

References

Advanced Page (DS1202SER_RX_Cx) 134
Block Description (DS1202SER_RX_Cx) 131
RX Parameters Page (DS1202SER_RX_Cx) 133

RX Parameters Page (DS1202SER_RX_Cx)

Purpose

To specify the receiving parameters.

Dialog settings

Reception mode Lets you specify how to react if there are fewer than the expected number of bytes in the receive buffer:

Data Handling	Meaning
Skip read operation	The new data is left in the receive buffer. The received data is collected in the receive buffer until the specified number

Data Handling	Meaning
	of bytes is reached. Then it is copied to the RXBytes output.
Read available data anyway	All the available data is copied from the receive buffer to the RXBytes output.

Parameter flexibility Lets you specify whether the number of bytes to be received is fixed (non-tunable) or variable (tunable).

Number of bytes Lets you specify the number of bytes to be received within one sample step.

Maximum number of bytes Lets you specify the maximum number of bytes that can be received within one sample step. Value range: 1 … (SW FIFO size-1) (SW FIFO size is a block parameter, see DS1202SER_SETUP_Cx on page 118).

Specify the number of bytes Lets you specify whether to set the number of bytes to be received within one sample step via the NumBytes input or the block parameter.

Related topics

References

Advanced Page (DS1202SER_RX_Cx)

Purpose

To specify the output.

Dialog settings

Enable NumRXBytes port Lets you specify whether to output the number of bytes that could be received or not.

Enable Status port Lets you specify whether to output the transmission status or not.

Related topics

References

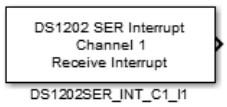
DS1202SER_INT_Cx_Iy

Where to go from here

Information in this section

Block Description (DS1202SER_INT_Cx_Iy)

Block



Purpose

To make the interrupts of the serial interface available as trigger sources in the model.

I/O mapping

For information on the I/O mapping, refer to Serial Interface (MicroLabBox Features 🕮).

I/O characteristics

The output triggers a function call to a subsystem if it is connected.

Dialog pages

The dialog settings can be specified on the following pages:

- **Unit Page** (refer to Unit Page (DS1202SER_INT_Cx_Iy) on page 136)
- **Interrupt Page** (refer to Interrupt Page (DS1202SER_INT_Cx_Iy) on page 136)

Related RTLib functions

This RTI block is implemented using the following RTLib functions:

- dsser_subint_handler_inst
- dsser_subint_enable
- dsser_subint_disable

Related topics

References

dsser_subint_disable (MicroLabBox RTLib Reference 🕮)
dsser_subint_enable (MicroLabBox RTLib Reference 🕮)
dsser_subint_handler_inst (MicroLabBox RTLib Reference 🕮)
Interrupt Page (DS1202SER_INT_Cx_Iy) 136
Unit Page (DS1202SER_INT_Cx_Iy) 136

Unit Page (DS1202SER_INT_Cx_Iy)

Purpose

To specify the channel on which an interrupt will be made available.

Dialog settings

Channel number Lets you choose the number of the channel within the range 1 … 2.

Related topics

References

Block Description (DS1202SER_INT_Cx_Iy) 135
Interrupt Page (DS1202SER_INT_Cx_Iy) 136

Interrupt Page (DS1202SER_INT_Cx_Iy)

Purpose

To specify the interrupt source.

Dialog settings

Interrupt source Lets you choose the interrupt type. The following table shows the available interrupt types:

Interrupt Type	Meaning
RX SW FIFO	Interrupt triggered when the number of bytes in the receive buffer reaches the specified threshold (see Initial RX SW FIFO threshold)
TX SW FIFO	Interrupt triggered when the transmit buffer is empty
Line status	Line status interrupt of the UART
Modem status	Modem status interrupt of the UART

Initial RX SW FIFO threshold Lets you specify the RX SW FIFO threshold for the receive interrupt in the range 1 … (SW FIFO size -1) . The value should be a multiple of the UART threshold (see DS1202SER_SETUP_Cx on page 118).

The RX SW FIFO threshold can be changed during run time by using the block DS1202SER_INT_REC_LEV_Cx on page 138.

Related topics

References

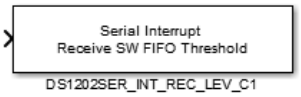
DS1202SER_INT_REC_LEV_Cx

Where to go from here

Information in this section

Block Description (DS1202SER_INT_REC_LEV_Cx)

Block



Purpose

To change the RX SW FIFO threshold during run time.

Description

The block changes the RX SW FIFO threshold that is initially specified by the **DS1202SER_INT_Cx_Iy** block (see DS1202SER_INT_Cx_Iy on page 135).

I/O mapping

For information on the I/O mapping, refer to Serial Interface (MicroLabBox Features 🕮).

I/O characteristics

- The **Receive SW FIFO Threshold** input sets a new RX SW FIFO threshold.
- The following table shows the characteristics of the block input:

Port	Characteristics	Value
Receive SW FIFO Threshold	Datatype	UInt32
	Range	1 … (SW FIFO size - 1)

SW FIFO size is a block parameter. For further information, refer to DS1202SER_SETUP_Cx on page 118.

Dialog pages

The dialog settings can be specified on the following page:

- Unit page (refer to Unit Page (DS1202SER_INT_REC_LEV_Cx) on page 139)

Related RTLib functions

This RTI block is implemented using the following RTLib functions:

- dsser_config
- dsser_fifo_reset
- dsser_transmit_fifo_level
- dsser_receive_fifo_level

Related topics

References

DS1202SER_INT_Cx_ly 135
DS1202SER_SETUP_Cx 118
dsser_config (MicroLabBox RTLib Reference 🕮)
dsser_fifo_reset (MicroLabBox RTLib Reference 🕮)
dsser_receive_fifo_level (MicroLabBox RTLib Reference 🕮)
dsser_transmit_fifo_level (MicroLabBox RTLib Reference 🕮)
Unit Page (DS1202SER_INT_REC_LEV_Cx) 139

Unit Page (DS1202SER_INT_REC_LEV_Cx)

Purpose

To specify the channel on which the RX SW FIFO threshold will be changed.

Dialog settings

Channel number Lets you choose the number of the channel within the range 1 … 2.

Related topics

References

Block Description (DS1202SER_INT_REC_LEV_Cx) 138

Serial Peripheral Interface

Introduction

The DIO CLASS 1 Blockset of MicroLabBox provides RTI blocks for implementing synchronous communication via a serial peripheral interface (SPI).

Where to go from here

Information in this section

DIO_CLASS1_SPI_SETUP_BLx

Purpose

To set up a serial peripheral interface on DIO Class 1.

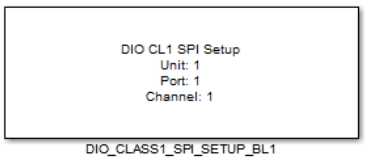
Where to go from here

Information in this section

Block Description (DIO_CLASS1_SPI_SETUP_BLx)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To set up a serial peripheral interface on DIO Class 1.

Description

You can use this block to configure an SPI unit. The unit number, output port number, output channel number and the number of chip select channels are used to identify an interface. Within an SPI communication network, the DIO_CLASS1_SPI blocks support only the SPI master mode.

The input channel can be configured as source for the generation of an interrupt or a trigger event.

Signal	Connector Pin	Description
MISO	(PortNo[1)] - 1) * 16 + ChannelNo[2)]	Master In, Slave Out (also known as Data Out)
SCLK	(PortNo[1)] - 1) * 16 + ChannelNo[2)] + 1	Serial clock (also known as SPI clock)
MOSI	(PortNo[1)] - 1) * 16 + ChannelNo[2)] + 2	Master Out, Slave In (also known as Data In)

Signal	Connector Pin	Description
CS1	(PortNo[1] - 1) * 16 + ChannelNo[2] + 3	Chip Select 1 (also known as Slave Select) A chip select channel is used to address a certain SPI slave.
CS2	(PortNo[1] - 1) * 16 + ChannelNo[2] + 4	Chip Select 2 (optional) Using a multiplexer, you can address up to 15 SPI slaves. Refer to Parameters Page (DIO_CLASS1_SPI_CYCLE_SETUP_BLx) on page 148.
…	…	…
CS4	(PortNo[1] - 1) * 16 + ChannelNo[2] + 6	Chip Select 4 (optional)

[1] Specified by the Port number on the Unit page.
[2] Specified by the Channel number on the Unit page.

Note

DIO Class 1 supports up to 4 SPI units.

I/O mapping

For information on the mapping of channel numbers to the related I/O pins of the MicroLabBox I/O connector, refer to Connector Pinouts (MicroLabBox Hardware Installation and Configuration 🕮).

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (DIO_CLASS1_SPI_SETUP_BLx) on page 144)
- Event page (refer to Event Page (DIO_CLASS1_SPI_SETUP_BLx) on page 145)

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- DioCl1Spi_create()
- DioCl1Spi_setSignalVoltage()
- DioCl1Spi_setChipSelectPolarity()
- DioCl1Spi_setInterruptMode()
- DioCl1Spi_setTriggerMode()
- DioCl1Spi_setTriggerLineOut()
- DioCl1Spi_apply()
- DioCl1Spi_write()
- DioCl1Spi_start()

Unit Page (DIO_CLASS1_SPI_SETUP_BLx)

Purpose

To specify the channels of the SPI unit and the voltage for the *High* level.

Dialog settings

Unit number Lets you select a unique unit number in the range 1 … 4.

Port number Lets you select the port number to be used in the range 1 … 3.

Channel number Lets you select the channel number for the input of the MISO signal. The consecutive channel numbers are reserved for the output of the SCLK, MOSI and CS1 signals

Note

Concurrent access to the same DIO Class 1 channel by other blocks or functions is not allowed.

Number of chip select channels Lets you select the number of channels to be used for chip select signals. The range depends on the remaining number of channels according to the selected Channel number.

Channel Number	Possible Number of Chip Select Channels
1 … 10	1 … 4
11	1 … 3
12	1 … 2
13	1

MISO connector pin Displays the connector pin for the Master-In-Slave-Out signal, also known as *Data Out* signal.

Clock connector pin Displays the connector pin for the serial clock signal, also known as *SPI clock* signal.

MOSI connector pin Displays the connector pin for the Master-Out-Slave-In signal, also known as *Data In* signal.

Chip select connector pin Displays the range of connector pins that are used for chip select signals, also known as *Slave Select* signals.

Reference connector pin Displays the range of reference connector pins for the selected channels.

High level Lets you set the voltage level of *High* on the output channel to 2.5 V, 3.3 V or 5.0 V.

Event Page (DIO_CLASS1_SPI_SETUP_BLx)

Purpose

To enable event generation for the specified channel.

Description

The event channel is assigned to the specified SPI input channel, but the state of the input channel has no effect on event generation. You can configure interrupt event generation, trigger event generation, or both to signal that an SPI cycle is finished (chip select signal is reset to inactive).

Dialog settings

Port Displays the port number that you selected on the **Unit** page.

Channel Displays the range of channels that you selected on the **Unit** page.

Enable event generation Lets you enable the generation of trigger events or interrupts or both.

Event mode Lets you select the events to be generated at the activated transition.

Event Mode	Meaning
Interrupt	Generate interrupt events at the activated transitions.
Trigger	Generate trigger events at the activated transitions.
Interrupt and Trigger	Generate both interrupts and trigger events at the activated transitions.

If you select interrupt generation, your model must contain a related DIO_CLASS1_HWINT_BLx block configured with the same port and channel number and *End of SPI cycle* as the interrupt type.

Trigger line Lets you select a trigger line number to be used for trigger event generation in the range 1 … 16.

Note

It is not allowed to use the same trigger line number in different event-generating blocks.

This setting is enabled only if an event mode with trigger generation is selected.

DIO_CLASS1_SPI_CYCLE_SETUP_BLx

Purpose

To specify an SPI cycle configuration.

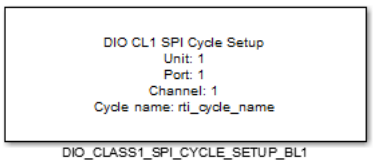
Where to go from here

Information in this section

Block Description (DIO_CLASS1_SPI_CYCLE_SETUP_BLx)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To specify an SPI chip select cycle.

Description

A chip select cycle or an SPI cycle configuration is used to specify the communication details. For example, the configuration includes the parameters for the timing behavior, the baud rate, the signal polarity, and the number of words to be transmitted or received. To execute the RTI blocks that transmit or receive messages, you only have to specify the identifier of an existing SPI cycle configuration.

You can define up to 64 SPI cycle configurations per SPI unit. An SPI unit must always be represented by a DIO_CLASS1_SPI_SETUP_BLx block. If the related DIO_CLASS1_SPI_SETUP_BLx block is removed from the model, the SPI unit is unknown and the cycle configuration is invalid.

I/O mapping

For information on the mapping of channel numbers to the related I/O pins of the MicroLabBox I/O connector, refer to Connector Pinouts (MicroLabBox Hardware Installation and Configuration 🕮).

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (DIO_CLASS1_SPI_CYCLE_SETUP_BLx) on page 147)
- Parameters page (refer to Parameters Page (DIO_CLASS1_SPI_CYCLE_SETUP_BLx) on page 148)

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- DioCl1SpiCycle_create()
- DioCl1SpiCycle_setBitDirection()
- DioCl1SpiCycle_setChipSelectConf()
- DioCl1SpiCycle_setBaudRate()
- DioCl1SpiCycle_setClockPolarity()
- DioCl1SpiCycle_setClockPhase()
- DioCl1SpiCycle_setTimeBeforeTran()
- DioCl1SpiCycle_setTimeAfterTran()
- DioCl1SpiCycle_setTimeCSInactive()
- DioCl1SpiCycle_setTimeBetwWords()
- DioCl1SpiCycle_apply()
- DioCl1SpiCycle_start()

Unit Page (DIO_CLASS1_SPI_CYCLE_SETUP_BLx)

Purpose

To specify a name for the cycle configuration.

Dialog settings

SPI unit Lets you select a serial peripheral interface specified by a DIO_CLASS1_SPI_SETUP_BLx block in the model. The identifier of the SPI consists of the unit number, the port number, and the channel number specified for the input channel, for example, **[Unit 1]: Port: 1, Channel: 1**.

The value of this setting is set to NOT SELECTED when this block is initially added to the model or if the specified SPI does not match the parameters of the DIO_CLASS1_SPI_SETUP_BLx block in the model.

Cycle name Lets you enter a name for the SPI cycle configuration. It must be unique for the specified SPI unit.

The string must follow the MATLAB naming conventions.

Related topics

References

Parameters Page (DIO_CLASS1_SPI_CYCLE_SETUP_BLx)

Purpose

To specify the parameters of the cycle configuration.

Description

For communicating via SPI bus, many settings must be specified. A chip select cycle configuration collects all these information.

A chip select cycle configuration consists of:

- Data settings

  To specify the data to be transmitted, for example, the number of words and the bits per word.
- Clock settings

  To specify the transfer protocol, for example, the baud rate, the polarity and the phase of the clock signal.
- Chip select settings

  To specify the timing behavior of the transmission and the chip select channels to be used.

For further information, refer to Serial Peripheral Interface (MicroLabBox Features 🕮).

Dialog settings

Cycle name Displays the cycle name specified on the Unit page.

Number of words Lets you specify the number of words to be transmitted within an SPI cycle in the range 1 … 64.

Bits per word Lets you specify the number of bits per word in the range 1 … 128.

Note

- The buffer size specified by the Number of words parameter and Bits per word parameter must not exceed 4096 bits. For example, if you specify 128 bits per word, the maximum number of words is 32.
- The data type used for transmitting is `UInt32`. If you specify a value greater than 32 bits, the Data inport of the DIO_CLASS1_SPI_TX_BLx block expects a vector. The width of the vector can be calculated as follows.

  ```
  Width = RoundUp(BitsPerWord / 32) · NumOfWords
  ```

  For example, if you have specified 2 words with 60 bits each, the resulting vector width is 4.

Bit direction Lets you specify the bit direction.

Setting	Description
LSB first	The sent data starts with the least significant bit. This is bit 0.
MSB first	The sent data starts with the most significant bit. This is bit <(BitsPerWord-1)>.

Baudrate Lets you specify the desired baud rate of the SPI clock output in the range 5 kHz … 2.5 MHz.

The specified baud rate is adjusted to the SPI clock precision. The resulting baud rate is calculated by the following formula:

$$Baudrate = 25 \cdot 10^6 / RoundDown\left(25 \cdot 10^6 / DesiredBaudRate + 0.5\right) Hz$$

If the desired baud rate contains an undefined variable or the value is out of range, the resulting baud rate is set to "XXXX".

Polarity Lets you specify the polarity of the SPI clock signal (SCLK) before the first edge.

Setting	Description
Low	The idle clock signal is low.
High	The idle clock signal is high.

Phase Lets you specify the phase of the SPI clock signal (also known as CPHA) to start the data capture.

Setting	Description
Leading	Data is captured at each odd-numbered edge of the clock signal (CPHA = 0).
Trailing	Data is captured at each even-numbered edge of the clock signal (CPHA = 1).

The Polarity parameter specifies whether the edge is a rising or falling. The combination of polarity and phase is also known as *mode*.

For more information on the timing behavior, refer to Serial Peripheral Interface (DIO Class 1) (MicroLabBox Features 🕮).

Time before transfer Lets you specify the desired time before starting a data transfer in seconds in the range 0 … 800 µs.

For more information, on how the resolution of timing parameters depend on the values, refer to Serial Peripheral Interface (DIO Class 1) (MicroLabBox Features 🕮).

This is the time during which the chip select signal is activated before the start of each word (Time between data words = 0) or the entire cycle (Time between data words > 0).

If the specified value contains an undefined variable or is out of range, the resulting value is set to "XXXX".

Time after transfer Lets you specify the desired time in seconds after stopping a data transfer in the range 0 … 800 µs.

For more information, on how the resolution of timing parameters depend on the values, refer to Serial Peripheral Interface (DIO Class 1) (MicroLabBox Features 🕮).

This is the time during which the chip select signal remains active after the clock signal ends, either after the last word (Time between data words = 0) or the entire cycle (Time between data words > 0).

If the specified value contains an undefined variable or is out of range, the resulting value is set to "XXXX".

Time between chip select cycles Lets you specify the desired time between two chip select cycles in seconds in the range 20 ns … 800 µs.

For more information, on how the resolution of timing parameters depend on the values, refer to Serial Peripheral Interface (DIO Class 1) (MicroLabBox Features 🕮).

This is the minimum time during which the chip select signal is driven to inactive state.

If the specified value contains an undefined variable or is out of range, the resulting value is set to "XXXX".

Note

This parameter is also used to specify the time during which the chip select signal is to be driven to inactive state between two *words* of a chip select cycle. Refer to the Time between data words parameter.

Time between data words Lets you specify the desired time between two data words in seconds in the range 0 µs … 800 µs.

For more information, on how the resolution of timing parameters depend on the values, refer to Serial Peripheral Interface (DIO Class 1) (MicroLabBox Features 🕮).

This value specifies the time between the last bit of the current word of the MOSI signal and the first bit of the next word of the MOSI signal. The chip select signal remains active for the specified time instead of being negated between two consecutive words of an SPI cycle.

If you specify 0 for the time between data words, the chip select signal is negated for the time specified by the **Time between chip select cycles** parameter between two consecutive words of an SPI cycle. The other timing parameters are also adapted to the timing behavior.

If the specified value contains an undefined variable or is out of range, the resulting value is set to "XXXX".

Activate Lets you specify the chip select output(s) to be activated when this cycle is transferred.

Your selection is handled as a bit mask, so that you can use it for a multiplexer. For example, if you have specified four chip select channels and you activate all of them, you can address up to 15 channels using a multiplexer.

You can only activate the chip select channels configured by the **DIO_CLASS1_SPI_SETUP_BLx** block.

At least one chip select output channel must be selected.

Related topics

References

Block Description (DIO_CLASS1_SPI_SETUP_BLx)..... 142
Unit Page (DIO_CLASS1_SPI_CYCLE_SETUP_BLx)..... 147

DIO_CLASS1_SPI_RX_BLx

Purpose

To receive the data of a specific chip select cycle.

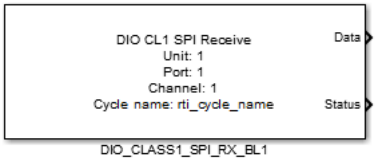
Where to go from here

Information in this section

Block Description (DIO_CLASS1_SPI_RX_BLx)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To receive the data of a specific chip select cycle.

Description

This block receives data from the SPI bus. The entire configuration of the communication is specified by the given SPI unit and cycle name. With the Data outport, you can read the received data. The result of the data receipt is returned by the Status outport.

Note

- There must be a related DIO_CLASS1_SPI_CYCLE_SETUP_BLx block for each DIO_CLASS1_SPI_RX_BLx block in the model to provide a chip select cycle configuration that can be referenced.
- There must be a related DIO_CLASS1_SPI_TX_BLx block for each DIO_CLASS1_SPI_RX_BLx block in the model to provide a sender of the data.

I/O mapping

For information on the mapping of channel numbers to the related I/O pins of the MicroLabBox I/O connector, refer to Connector Pinouts (MicroLabBox Hardware Installation and Configuration 🕮).

I/O characteristics

The following table describes the ports of the block:

Port	Description
Output	
Data	Outputs the received data of a specific chip select cycle. Data type: UInt32 Range: Depends on the width. Width: Depends on the specified data settings in the DIO_CLASS1_SPI_CYCLE_SETUP_BLx block. The width can be calculated by the following formula: `Width = RoundUp(BitsPerWord / 32) · NumOfWords` The width is set to 1 if it cannot be calculated because of an undefined variable, because the value is out of range or the reference to the SPI unit is missing.
Status	Outputs the status of the data receipt. Data type: UInt16 Range: 0 … 2 ▪ 0: No error, new data is available ▪ 1: No new data available ▪ 2: Data lost Width: 1

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (DIO_CLASS1_SPI_RX_BLx) on page 153)

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- DioCl1SpiCycle_read()
- DioCl1SpiCycle_getCycleData()

Unit Page (DIO_CLASS1_SPI_RX_BLx)

Purpose

To specify the chip select cycle to receive data from.

Dialog settings

SPI unit Lets you select a serial peripheral interface specified by a DIO_CLASS1_SPI_SETUP_BLx block in the model. The identifier of the SPI

consists of the unit number, the port number, and the channel number specified for the input channel, for example, [Unit 1]: Port: 1, Channel: 1.

The value of this setting is set to NOT SELECTED when this block is initially added to the model or if the specified SPI does not match the parameters of the DIO_CLASS1_SPI_SETUP_BLx block in the model.

Cycle name Lets you select a chip select cycle specified by a DIO_CLASS1_SPI_CYCLE_SETUP_BLx block in the model.

The value of this setting is set to NOT SELECTED when this block is initially added to the model, and if the specified cycle name has been modified in the related DIO_CLASS1_SPI_CYCLE_SETUP_BLx block.

Related topics

References

Block Description (DIO_CLASS1_SPI_RX_BLx)..... 152

DIO_CLASS1_SPI_TX_BLx

Purpose

To transmit data of a specific chip select cycle.

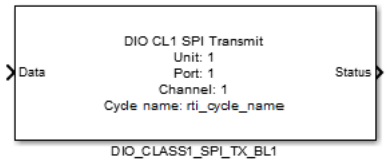
Where to go from here

Information in this section

Block Description (DIO_CLASS1_SPI_TX_BLx)

Block

Gives you information about the appearance and purpose of the block.



Purpose

To transmit data of a specific chip select cycle.

Description

This block is used to send data to the SPI bus. The entire configuration of the communication is specified by the given SPI unit and cycle name. With the Data inport, you can specify the data to be sent. The result of the transmission is returned by the Status outport.

Note

There must be a related DIO_CLASS1_SPI_CYCLE_SETUP_BLx block for each DIO_CLASS1_SPI_TX_BLx block in the model to provide a chip select cycle configuration.

I/O mapping

For information on the mapping of channel numbers to the related I/O pins of the MicroLabBox I/O connector, refer to Connector Pinouts (MicroLabBox Hardware Installation and Configuration 🕮).

I/O characteristics

The following table describes the ports of the block:

Port	Description
Input	
Data	Specifies the data of a specific chip select cycle that is to be sent. Data type: UInt32 Range: Depends on the width. Width: Depends on the specified data settings in the DIO_CLASS1_SPI_CYCLE_SETUP_BLx block. The width can be calculated by the following formula: `Width = RoundUp(BitsPerWord / 32) · NumOfWords` The width is set to 1 if it cannot be calculated because of an undefined variable, because a value is out of range or the reference to the SPI interface is missing.
Output	
Status	Outputs the status of the data transmission. Data type: UInt16 Range: 0 … 2 ▪ 0: No error Transmission was successful. ▪ 1: FIFO overflow The current message length was greater than the available FIFO buffer size. The data was therefore not transmitted. ▪ 2: Data lost An attempt was made to transfer new data for this SPI cycle before the data of the previous transfer was completely sent. The data of the previous transfer is (partially) lost. Width: 1

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (DIO_CLASS1_SPI_TX_BLx) on page 157)

Related RTLib functions

This RTI block is implemented using the following RTLib functions. The *MicroLabBox RTLib Reference* contains descriptions of these functions.

- DioCl1SpiCycle_setCycleData()
- DioCl1SpiCycle_write()

Unit Page (DIO_CLASS1_SPI_TX_BLx)

Purpose

To specify the chip select cycle to send data to.

Dialog settings

SPI unit Lets you select a serial peripheral interface specified by a DIO_CLASS1_SPI_SETUP_BLx block in the model. The identifier of the SPI consists of the unit number, the port number, and the channel number specified for the input channel, for example, [Unit 1]: Port: 1, Channel: 1.

The value of this setting is set to NOT SELECTED when this block is initially added to the model or if the specified SPI does not match the parameters of the DIO_CLASS1_SPI_SETUP_BLx block in the model.

Cycle name Lets you select a chip select cycle specified by a DIO_CLASS1_SPI_CYCLE_SETUP_BLx block in the model.

The value of this setting is set to NOT SELECTED when this block is initially added to the model, and if the specified cycle name has been modified in the related DIO_CLASS1_SPI_CYCLE_SETUP_BLx block.

Related topics

References

Block Description (DIO_CLASS1_SPI_TX_BLx)..... 155