RTI Ethernet Blockset

Reference

For RTI Ethernet Blockset 1.2.5

Release 2021-A – May 2021

dSPACE

Contents

About This Reference

Content

This RTI Reference is a complete description of the Real-Time Interface (RTI) blocks and their settings provided by the RTI Ethernet blockset. You can use this blockset to model communication in a Simulink® model via the Ethernet interface.

The RTI Ethernet blockset is supported by

- DS1007 PPC Processor Board
- MicroLabBox

Required knowledge

If you want to implement a network-based communication with this blockset, basic knowledge in IP-based networks and handling Simulink models is assumed.

This reference is primarily for engineers who implement real-time applications by using MATLAB®/Simulink®.

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.

Symbol	Description
⁇	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.
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 Ethernet Blockset

Objective

Provides basic information on the RTI Ethernet Blockset.

Overview of the RTI Ethernet Blockset

Objective

To provide a short description of the blockset's main features, its components and how to access them.

Main features

The RTI Ethernet Blockset is a Simulink® blockset for modeling communication via an Ethernet interface using the TCP/IP and UDP/IP protocols. With this, you have access to all the external devices that also provide an Ethernet interface, such as another dSPACE board or a calibration device.

The blockset provides RTI blocks for configuring the Ethernet interface by specifying its IP address, and for sending and receiving data via the TCP/IP or UDP/IP protocols.

Hardware support   The blockset supports the DS1007 PPC Processor Board and MicroLabBox. Both platforms provide Gigabit Ethernet switches with three RJ45 connectors. While one connector is always to be used for the host PC communication, the other two connectors can be individually configured for I/O communication or host PC communication.

Supported network features and limitations   For an overview, here is a list of basic features and blockset-specific limitations.

- The blockset supports the TCP/IP and UDP/IP protocols.
- You can define up to 256 sockets for bidirectional communication.
- Each socket can be configured with a maximum frame size of 65535 bytes for TCP/IP and 65507 bytes for UDP/IP.
- Interrupt-driven data receipt for both protocols (one interrupt per socket).
- Auto negotiation is supported.

- Data rates of 10/100/1000 MBit/s are supported.
- DHCP is supported.
- Static gateway configuration is supported.
- Broadcasts are supported.
- Listening to any IP address and port is supported.
- IP fragmentation is supported.
- Multicast is not supported.
- Ethernet flow control is not supported.
- IPv6 is not supported, only the IPv4 protocol can be used.

Connecting the board

For information on the Ethernet connectors, refer to Board Overview (PHS Bus System Hardware Reference 🕮).

Library access

The library can be opened with the following methods:

> **Note**
>
> Choose the platform support before you open the RTI Ethernet Blockset to get the predefined board-specific settings.

- Enter `rtiethernet` in the MATLAB Command Window.
- Click Blocksets - RTI ETHERNET Blockset in the board's blockset.
- Navigate to the RTI Ethernet Blockset folder in the Simulink Library Browser to access the RTI blocks of the library separately.

When you open the block library, the blockset is displayed.

Library components

The library provides the following RTI blocks:

- Setup
    - ETHERNET_SETUP_BLx on page 14
    - ETHERNET_TCP_SETUP_BLx on page 19
    - ETHERNET_UDP_SETUP_BLx on page 33
- Send and receive
    - ETHERNET_TCP_TX_BLx on page 29 to send messages to the specified TCP destination.
    - ETHERNET_TCP_RX_BLx on page 24 to receive messages from the specified TCP source.
    - ETHERNET_UDP_TX_BLx on page 41 to send messages to the specified UDP destination.
    - ETHERNET_UDP_RX_BLx on page 37 to receive messages from the specified UDP source.
- Interrupt generation
    - ETHERNET_HWINT_BLx on page 45 to make interrupts that are generated on data receipt available as trigger sources.

Demo model

For a Simulink model that shows how to use the RTI Ethernet Blockset, refer to the blockset's Demo library. This model also contains preconfigured blocks for encoding and decoding the signals.

Components of the RTI Ethernet Blockset

Objective

The RTI Ethernet library provides RTI blocks that you use in the Simulink model to implement access to external devices via the Ethernet interface using TCP/IP and UDP/IP protocols.

Where to go from here

Information in this section

ETHERNET_SETUP_BLx

Purpose   To configure the Ethernet interface.

Where to go from here Information in this section

Information in other sections

Block Description (ETHERNET_SETUP_BLx)

Block   To give information about the appearance and purpose of the block.

Purpose   To initialize the Ethernet interface.

Description   In a single-processor or single-core system, the model can contain one
ETHERNET_SETUP_BLx block only.

In a multiprocessor or multicore system, each submodel, which contains Ethernet
blocks, must have its own **ETHERNET_SETUP_BLx** block. Each submodel can be
specified with an individual IP address.

> **Note**
>
> - Both submodels should use the same DHCP setting, see Obtain an IP address automatically on the block's Unit Page (ETHERNET_SETUP_BLx).
>   Mixed mode is currently not supported.
> - The local loopback address 127.0.0.1 can only be used if the IP addresses are automatically obtained.

I/O characteristics

The following table describes the ports of the block:

Port	Description
Output	
Link Status	Outputs the link status of the Ethernet interface.
	Range: 0 … 7, 255
	Data type: UInt32
	Width: Specified number of Ethernet connectors
	- 0: Not connected.
	- 1: Connected - 10BASE-T[1] half-duplex
	- 2: Connected - 10BASE-T[1] full-duplex
	- 3: Connected - 100BASE-TX[1] half-duplex
	- 4: Connected - 100BASE-T4[1]
	- 5: Connected - 100BASE-TX[1] full-duplex
	- 6: Connected - 1000BASE-T[1] half-duplex
	- 7: Connected - 1000BASE-T[1] full-duplex
	- 255: Unknown link status
	If the specified number of Ethernet connectors exceeds the number of available Ethernet connectors on your hardware platform, the link status is set to 255 for the exceeding connectors. The Link Status outport is only available if the Enable Link Status port parameter is set. DS1007:

Port	Description
	MicroLabBox:
Watchdog Status	Outputs the status of the operating IP stack. Range: 0 … 1 Data type: UInt32 ▪ 0: No response from the operating IP stack ▪ 1: Operating IP stack is alive The **Watchdog Status** outport is only available if the **Enable Watchdog Status port** parameter is set.
DHCP Status	Outputs the status of the DHCP configuration process. If you have set the **Obtain an IP address automatically** option in the **Unit** page, the related network has to contain a DHCP server. The Ethernet interface operates in this case as a DHCP client. It sends a request to the DHCP server during the initialization to acquire its own IP setting. The state is set to 1. Once the DHCP configuration process is successfully completed, the DHCP Status is set to 2 and the real-time application is able to communicate within its network. If no DHCP server can be reached, the DHCP status changes to 255. If the **Obtain an IP address automatically** option is cleared, the Ethernet interface operates without a DHCP server. The state is then set to 0. Range: 0 … 2, 255 Data type: UInt32 ▪ 0: DHCP client not started (initial state) ▪ 1: Obtaining IP in progress (DHCP client is started - IP request is sent to DHCP server) ▪ 2: Obtaining IP done successfully (ready state) ▪ 255: No DHCP server available (no response) The **DHCP Status** outport is only available if the **Enable DHCP Status port** parameter is set.
Local IP	Outputs the IP address of the Ethernet interface. This can be the specified IP address on the **Unit** page or the IP address obtained by the DHCP server. The **Local IP** outport is only available if the **Enable Local IP port** parameter is set.

[1] Adapted and reprinted with permission from IEEE. Copyright IEEE 2018. All rights reserved.

Dialog pages

The dialog settings can be specified on the following pages:

Unit Page (ETHERNET_SETUP_BLx)

Purpose

To specify the properties of the Ethernet interface.

Dialog settings

Obtain an IP address automatically   Lets you specify whether the Ethernet interface operates as a DHCP client.

The Dynamic Host Configuration Protocol (DHCP) is a network protocol that is used to configure network devices so that they can communicate in an IP network. A DHCP server assigns IP addresses, network masks and gateway addresses to network clients.

If the Obtain an IP address automatically option is selected the Local IP address, Subnet mask and Gateway address options are disabled.

Local IP address [0 … 255]   Lets you specify the IP address of the board providing the Ethernet interface. Each number of the IP address must be in the range 0 … 255, for example, 192.168.140.6.

> **Note**
>
> Do not use the same IP address for the Local IP address setting and the board's host interface.
> The board's host interface can be switched with the Ethernet interface. The same IP address is then used twice on your network. This might result in unpredictable communication results.

Subnet mask [0 … 255]   Lets you specify the subnet mask according to the IP address. Each number of the Subnet mask must be in the range 0 … 255.

Gateway address [0 … 255]   Lets you specify the gateway address of the local network. Each number of the Gateway address must be in the range 0 … 255.

Related topics

References

Advanced Page (ETHERNET_SETUP_BLx)

Purpose   To specify whether to show or hide Ethernet/IP information.

Dialog settings   **Enable Watchdog Status port**   Lets you enable the watchdog status outport.

Enable Local IP port   Lets you enable the local IP outport.

Enable Link Status port   Lets you enable the link status outport.

Number of Ethernet connectors   Lets you select the number of Ethernet connectors you want to use in the range 1 … 8.

If the specified number of Ethernet connectors exceeds the number of available Ethernet connectors on your hardware platform, the link status is set to 255 for the exceeding connectors.

This option is enabled if the Enable Link Status port option is selected.

Related topics   References

ETHERNET_TCP_SETUP_BLx

Purpose

To configure the TCP/IP communication.

Where to go from here

Information in this section

Information in other sections

Block Description (ETHERNET_TCP_SETUP_BLx)

Block

To give information on the appearance and purpose of the block.

Purpose

To configure the TCP/IP communication.

Description

The **ETHERNET_TCP_SETUP_BLx** block creates a TCP socket, that represents the TCP communication. The block is used to manage one **ETHERNET_TCP_RX_BLx** receive block and one **ETHERNET_TCP_TX_BLx** transmit block.

I/O characteristics The following table describes the ports of the block:

Port	Description
Input	
Reconnect	Controls the TCP connection to the remote device. A rising edge signal closes the related socket. Dependencies on the **Operation mode**: ▪ Server: The connection must be re-activated by the client. ▪ Client: The connection is automatically re-activated. Closing the socket takes effect on the **ETHERNET_TCP_RX_BLx** and **ETHERNET_TCP_TX_BLx** blocks. The **Status** outport of both blocks is switched to 3. Data type: Boolean
Output	
Connection State	Outputs the state of the TCP connection. Data type: UInt32 ▪ 0: Not connected ▪ 1: Connected
Connection Event	Outputs the most recent event of the connection. Data type: UInt32 ▪ 0: No event ▪ 1: Reserved internal event ▪ 2: Open connection request has been acknowledged internally ▪ 3: Connection has been closed ▪ 4: Close connection request has been acknowledged internally ▪ 5: Connection has been established ▪ 6: Connection error
RX Dropped Frames	Outputs the number of received TCP data frames that were dropped due to an overflow of the socket receive queue. The counter is reset when the real-time application starts or restarts. Data type: Double The **RX Dropped Frames** outport is only available if the **Enable RX Dropped Frames port** parameter is set.

Dialog pages The dialog settings can be specified on the following pages:

▪ Unit page (refer to Unit Page (ETHERNET_TCP_SETUP_BLx) on page 21)

▪ Advanced page (refer to Advanced Page (ETHERNET_TCP_SETUP_BLx) on page 23)

Unit Page (ETHERNET_TCP_SETUP_BLx)

Purpose   To configure the TCP/IP communication.

Dialog settings   **Socket ID**   Displays the socket ID.

> **Note**
>
> There must be an **ETHERNET_SETUP_BLx** block in the model to get a socket ID. The TCP setup block is represented uniquely by this socket ID within the whole Simulink model.

Bind socket to local IP address   Lets you bind the specified socket to the local IP address. Only the bound local IP address will be used for this socket for sending and receiving data. If you want to use a loopback address within your model, such as 127.0.0.1, this option must be disabled.

Operation mode   Lets you specify whether the operation mode of the selected socket is Server or Client.

There are two different operation modes:

- Server

  The server socket provides a service to a single client. A server process is always alert and ready to serve incoming connection requests. The setup block creates a server socket and listens to the specified local port.

- Client

  The client socket initiates a connection request for the service provided by the server. The setup block creates a client socket and builds up a connection to the server. The server is specified by the remote IP address and remote port.

Dependencies to the operation modes are shown in the table below.

Operation Mode	Dependencies
Server	- The Accept any remote IP address option is available. - The Accept any remote port option is available.
Client	- The Specify local port option is available.

Accept any remote IP address   Lets you specify the remote IP used for the TCP connection.

- If this option is selected, the server socket will accept any client to connect to.
- If this option is not selected, a specific client can be specified via the Remote IP address setting.

Remote IP address [0 … 255]   Lets you specify the IP address of the remote system.

The behavior of the specified remote IP address depends on the specified operation mode.

Operation Mode	Description
Server	The remote IP address is used to await connection requests from.
Client	The remote IP address is used to send connection requests to. An ARP (Address Resolution Protocol) request will be sent out of the client during initialization. This reduces the execution time to find the MAC address that belongs to the IP address of the server.

Accept any remote port   Lets you specify the remote port used for the TCP connection.

- If this option is selected, the server socket will accept clients to connect to any remote port.
- If this option is not selected, a specific remote port can be specified via the Remote port setting.

Remote port [1 … 65535]   Lets you specify the remote port number for the TCP connection.

Specify local port   Lets you specify whether the local TCP port is set by the user or is automatically selected.

Local port [1 … 65535]   Lets you specify the local TCP port for receiving and sending data streams.

Send mode (Nagle's algorithm)   Lets you specify the way how TCP packets have to be sent. The offered send modes have the task to improve the efficiency of the TCP/IP network communication.

There are two different send modes:

- No delay
  The Nagle-Algorithm is disabled, which leads to an immediate transmission.
- Delay
  The Nagle-Algorithm is enabled, then small packets will be delayed and combined to increase data throughput.

Related topics

References

Advanced Page (ETHERNET_TCP_SETUP_BLx)

Purpose	To specify whether to show or hide TCP/IP connection information.

Dialog settings

Socket ID　Displays the socket ID.

Enable interrupt on data receipt　Lets you activate the interrupt generation.

Enable Connection Event port　Lets you enable the Connection Event outport.

Enable RX Dropped Frames port　Lets you enable the RX Dropped Frames outport.

Related topics

References

ETHERNET_TCP_RX_BLx

Purpose

To receive data via Ethernet using the TCP/IP protocol.

Where to go from here

Information in this section

Information in other sections

Block Description (ETHERNET_TCP_RX_BLx)

Block

To give information on the appearance and purpose of the block.

Purpose

To receive data via Ethernet using the TCP/IP protocol.

Description

With this block, you can receive data via the TCP/IP protocol from a remote Ethernet network device. The data output is a vector of UInt8 values.

I/O characteristics   The following table describes the ports of the block.

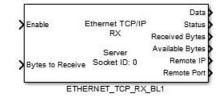| Port	Description
Input	
Enable	Controls the block's functionality. Data type: Boolean - 0: Disabled (see the description below) - 1: Enabled Note the following behavior if the block is disabled: - The socket remains open. - The specified Ethernet interface continues receiving the relevant TCP socket data as usual. The received TCP data will be internally stored in a related socket receive queue. - The interrupt generation is not affected, if it is activated via the according ETHERNET_TCP_SETUP_BLx block. An interrupt is provoked when a valid TCP datagram (according to the specified frame size setting) is received. - No TCP data will be delivered to the Simulink model. Internally stored data frames are not read out of the socket receive queue. This can lead to data loss caused by a buffer overflow.
Bytes to Receive	Provides the possibility to dynamically specify the number of bytes to be received if the Enable variable frame size option is set. This inport is available if the Enable variable frame size is selected and the Variable frame size mode is set to By input port. In this case, the maximum number of bytes to be read is limited by the Maximum frame size setting.
Output	
Data	Outputs the data received via Ethernet by using the TCP/IP protocol. The width of the port depends on the specified value of the Maximum frame size setting. Data type: UInt8

Status	Outputs the status of the last data read from the software socket receive queue.
	Range: 0 … 4
	Data type: UInt32
	
	- 0: Successful data read
	  The relevant number of bytes is successfully read out of the socket receive queue.
	- 1: No data read
	  The Enable inport is set to Disabled.
	- 2: No data read
	  The socket receive queue is empty or it does not contain the required number of bytes.
	- 3: No data read
	  The connection is closed or not yet established after reconnecting.
	- 4: No data read
	  The model is in its initialization phase and no data has been received yet.

Received Frames	Outputs the number of received TCP data frames which are successfully read out of the socket receive queue. Data type: Double You can display either the Received Bytes port or the Received Frames port. This outport is therefore not available, if the Enable variable frame size option is set.

Port	Description
Received Bytes	Outputs the number of received data bytes which are successfully read out of the socket receive queue. Range: 0 … 65535 Data type: UInt32 This outport is available if the **Enable variable frame size** option is set.
Available Bytes	Outputs the number of bytes that are available in the socket receive queue. Data type: UInt32 This outport is available if the **Enable Available Bytes port** option is set.
Remote IP	Outputs the IP address of the connected remote device. This outport is useful if the **Accept any remote IP address** option is set, because it allows you to get information on the current sender. Range: 0 … 255 Data type: UInt8 This outport is available if the **Enable Remote IP port** option is set.
Remote Port	Outputs the port number of the connected remote device. Range: 0 … 65535 Data type: UInt16 This outport is available if the **Enable Remote Port port** option is set.

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (ETHERNET_TCP_RX_BLx) on page 27)
- Advanced page (refer to Advanced Page (ETHERNET_TCP_RX_BLx) on page 28)

Related topics

References

Unit Page (ETHERNET_TCP_RX_BLx)

Purpose   To receive data via Ethernet using the TCP/IP protocol.

Dialog settings   **Socket ID**   Lets you specify the socket ID.

> **Note**
>
> There must be an ETHERNET_TCP_SETUP_BLx block in the model to specify the socket ID.

Operation mode   Displays the operation mode specified for the selected socket.

Remote IP address   Displays the specified remote IP address of the selected socket.

Remote port   Displays the specified remote port of the selected socket.

Local port   Displays the specified local port number of the selected socket.

Frame size [1 … 65535] / Maximum frame size [1 … 65535]   Lets you specify the length of the data frame in bytes to be read from the socket receive queue.

Enable variable frame size   Lets you specify whether the amount of data to be read is always fix or the amount of data is variable.

Dependencies to the block's ports and settings are shown in the table below.

Enable Variable Frame Size	Dependencies
Off	Received Frames outport is available.
On	▪ The dialog text of the Frame size [1 … 65535] setting is changed to Maximum frame size [1 … 65535]. ▪ The Variable frame size mode option is available. ▪ The Bytes to Receive inport is available, if the Variable frame size mode option is set to By block input port. ▪ The Received Frames outport is not available. ▪ The Received Bytes outport is available.

Variable frame size mode   Lets you specify how the amount of bytes is dynamically read from the socket receive queue.

Mode	Description
By input port	The Bytes to Receive inport is used to control reading from the socket receive queue. You can specify at most the specified maximum frame size. ▪ If less data is available in the socket receive queue than specified by the Bytes to Receive inport, no data is read. The Status outport is set to 2 and no interrupt will be generated.

Mode	Description
	▪ If at least the number of bytes to receive is available in the socket receive queue, the specified number of bytes will be read and output in the Data outport. If the Enable interrupt on data receipt option is set, an interrupt is generated. The Status outport is set to 0. Note, that for initialization purposes, the first interrupt in the run-time phase is generated independently of the number of received bytes.
All available	All the bytes up to the specified maximum frame size that are available in the socket receive queue will be read and output in the Data outport. The Received Bytes outport is set to the number of received bytes. The Status outport is set to 0. If the Enable interrupt on data receipt option is set, an interrupt is generated on every data receipt.

Related topics

References

Advanced Page (ETHERNET_TCP_RX_BLx)

Purpose

To specify whether to show or hide receive information.

Dialog settings

Socket ID   Displays the socket ID that you can select on the Unit page.

Enable Available Bytes port   Lets you enable the Available Bytes outport.

Enable Remote IP port   Lets you enable the Remote IP outport.

Enable Remote Port port   Lets you enable the Remote Port outport.

Related topics

References

ETHERNET_TCP_TX_BLx

Purpose   To transmit data via Ethernet using the TCP/IP protocol.

Where to go from here

Information in this section

Information in other sections

Block Description (ETHERNET_TCP_TX_BLx)

Block   To give information on the appearance and purpose of the block.

Purpose   To send data via Ethernet using the TCP/IP protocol.

Description   With this block, you can send data via the TCP/IP protocol to a remote Ethernet
network device. The required data input must be a vector of UInt8 values.

I/O characteristics   The following table describes the ports of the block.

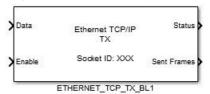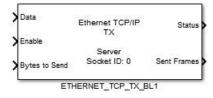| Port	Description
Input	
Data	Provides the data to be sent. The width of the port depends on the specified value of the **Frame size** option. Data type: UInt8
Enable	Controls the block's functionality. Data type: Boolean ▪ 0: Disabled (see the description below) ▪ 1: Enabled Note the following behavior if the block is disabled: ▪ The socket remains open. ▪ The specified Ethernet interface continues transmitting the relevant TCP socket data as usual. ▪ New data is not stored in the socket transmit queue.
Bytes to Send	Provides the possibility to determine the number of bytes that are sent from the socket transmit queue. The maximum number of bytes to be sent is limited by the **Frame size** option. Range: 0 … 65535 Data type: UInt32 This inport is available if the **Enable variable frame size** option is set.
Output	

Status	Outputs the status of the last data send to the socket transmit queue.
	Range: 0 … 4
	Data type: UInt32
	▪ 0: Successful data transfer
	  The relevant number of bytes is successfully written to the socket transmit queue.
	▪ 1: No data transfer
	  The **Enable** inport is set to **Disabled**.
	▪ 2: No data transfer
	  The socket transmit queue is full.
	▪ 3: No data transfer
	  The connection is not established yet or the connection is closed.
	▪ 4: No data transfer
	  The model is in its initialization phase and no data has been sent yet.

Sent Frames	Outputs the number of data frames copied successfully to the socket transmit queue. However, the outport does not show the number of frames that were really sent by the Ethernet interface. This outport is available if the **Enable Sent Frames port** option is set.

Dialog pages   The dialog settings can be specified on the following pages:

▪

▪

Unit Page (ETHERNET_TCP_TX_BLx)

Purpose   To send data via Ethernet by using the TCP/IP protocol.

Dialog settings   **Socket ID**   Lets you specify the socket ID.

> **Note**
>
> There must be an ETHERNET_TCP_SETUP_BLx block in the model to specify the socket ID.

Operation mode   Displays the operation mode specified for the selected socket.

Remote IP address   Displays the specified remote IP address of the selected socket.

Remote port   Displays the specified remote port of the selected socket.

Local port   Displays the specified local port number of the selected socket.

Frame size [1 … 65535]   Lets you specify the length of the data frame in bytes.

Enable variable frame size   Lets you specify whether the number of bytes to be sent will be determined using the value of the Bytes to Send inport. If the Enable variable frame size option is not set, the payload length of the sent frame is always fixed according to the Frame size setting.

Related topics   References

Advanced Page (ETHERNET_TCP_TX_BLx)

Purpose   To specify whether to show or hide send information.

Dialog settings   **Socket ID**   Displays the socket ID that you can select on the Unit page.

Enable Sent Frames port   Lets you enable the Sent Frames outport.
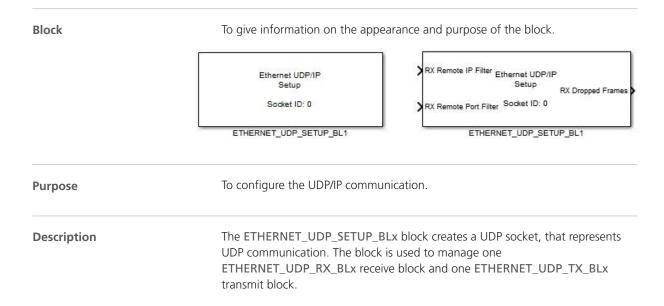
Related topics

References

ETHERNET_UDP_SETUP_BLx

Purpose	To configure the UDP/IP communication.

Where to go from here

Information in this section

Information in other sections

Block Description (ETHERNET_UDP_SETUP_BLx)

Block

To give information on the appearance and purpose of the block.

Purpose

To configure the UDP/IP communication.

Description

The **ETHERNET_UDP_SETUP_BLx** block creates a UDP socket, that represents UDP communication. The block is used to manage one **ETHERNET_UDP_RX_BLx** receive block and one **ETHERNET_UDP_TX_BLx** transmit block.

I/O characteristics The following table describes the ports of the block:

Port	Description
Input	
RX Remote IP Filter	Provides an input to dynamically specify the IP address of the remote device to accept UDP data from. The modification of the IP address by the filter takes affect at the next time step. If you set the filter to IP 0.0.0.0, you enable the port to accept data from any remote device. Range: 0 … 255 Data type: UInt8 This inport is available if the **Enable dynamic remote IP address filter** option is set. Note, that the **Accept any remote IP address** option must be disabled.
RX Remote Port Filter	Provides an input to dynamically specify the port of the remote device to accept UDP data from. The modification of the port number by the filter takes affect at the next time step. If you set the filter to port 0, you enable the port to accept data from any remote port. Range: 0 … 65535 Data type: UInt16 This inport is available if the **Enable dynamic remote port filter** option is set. Note, that the **Accept any remote port** option must be disabled.
Output	
RX Dropped Frames	Outputs the number of received UDP data frames dropped. UDP frames are dropped if: ▪ Invalid frame size (output signal 1) 　The received packet length does not comply with the data frame setting of the related **ETHERNET_UDP_RX_BLx**. ▪ Socket receive queue overflow (output signal 2) 　The software socket receive queue is full. Data type: Double This outport is available if the **Enable RX Dropped Frames port** option is set.

Dialog pages The dialog settings can be specified on the following pages:

▪ Unit page (refer to Unit Page (ETHERNET_UDP_SETUP_BLx) on page 35)

▪ Advanced page (refer to Advanced Page (ETHERNET_UDP_SETUP_BLx) on page 36)

Unit Page (ETHERNET_UDP_SETUP_BLx)

Purpose   To configure the UDP/IP communication.

Dialog settings   **Socket ID**   Displays the socket ID.

> **Note**
>
> There must be an **ETHERNET_UDP_SETUP_BLx** block in the model to get a socket ID. The UDP setup block is represented uniquely by this socket ID within the whole Simulink model.

Bind socket to local IP address   Lets you bind the specified socket to the local IP address. Only the bound local IP address will be used for this socket for sending and receiving data. If you want to use a loopback address within your model, such as 127.0.0.1, this option must be disabled.

Accept any remote IP address   Lets you specify whether you want to accept UDP datagrams from any remote IP address or only from the one specified in the Remote IP address setting.

Remote IP address [0 ... 255]   Lets you specify the IP address of the remote device.
An ARP request will be sent out of the related ETHERNET_UDP_TX_BLx block during initialization. This reduces the execution time required to find the MAC address that belongs to the Remote IP address.

Enable dynamic remote IP address filter   Enables the RX Remote IP Filter inport.
This setting is enabled, if the Accept any remote IP Address option is disabled..

Accept any remote port   Lets you specify whether you want to accept UDP datagrams from any remote port or only from the one specified in the Remote port setting.

Remote port [1 … 65535]   Lets you specify the remote port number for the UDP connection.

Enable dynamic remote port filter   Enables the RX Remote Port Filter inport.
This setting is enabled, if the Accept any remote port option is disabled.

Specify local port   Lets you specify whether the local UDP port is set by the user or is automatically set.

Local port [1 … 65535]   Lets you specify the local UDP port for receiving and sending data packets.
This setting is enabled only if you have set the Specify local port option.

Related topics

Advanced Page (ETHERNET_UDP_SETUP_BLx)

Purpose

To specify whether to show or hide UDP/IP communication information.

Dialog settings

Socket ID   Displays the socket ID that you can select on the Unit page.

Enable interrupt on data receipt   Lets you activate the interrupt generation.
If the ETHERNET_UDP_RX_BLx block listening on the specified socket has successfully received a UDP data frame, an interrupt is generated that can be handled by the related ETHERNET_HWINT_BLx block.

Enable RX Dropped Frames port   Lets you enable the RX Dropped Frames port outport.

Related topics

ETHERNET_UDP_RX_BLx

Purpose

To receive data via Ethernet using the UDP/IP protocol.

Where to go from here
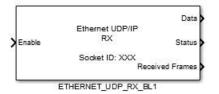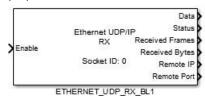
Information in this section

Information in other sections

Block Description (ETHERNET_UDP_RX_BLx)

Block

To give information on the appearance and purpose of the block.

Purpose

To receive data via Ethernet using the UDP/IP protocol.

Description

With this block, you can receive data via the UDP/IP protocol from a remote Ethernet network device. The data output is a vector of UInt8 values.

I/O characteristics   The following table describes the ports of the block:

Port	Description
Input	
Enable	Controls the block's functionality. Data type: Boolean ■ 0: Disabled (see the description below) ■ 1: Enabled Behavior if the block is disabled: ■ The socket remains open. ■ The Ethernet interface (via the IP stack) continues receiving the relevant UDP socket data as usual according to the specified frame size setting. The accepted UDP data will be internally stored in a related socket receive queue with limited size. ■ The interrupt generation is not affected, if it is activated via the according ETHERNET_UDP_SETUP_BLx block. An interrupt is provoked when a valid UDP datagram according to the specified frame size setting is received. ■ No UDP data will be delivered to the Simulink model. Internally stored data frames are not read out of the socket receive queue. This can lead to data loss caused by a buffer overflow.
Output	
Data	Outputs the data received via Ethernet by using the UDP/IP protocol. The width of the port depends on the specified value of the **Frame size** setting. Data type: UInt8
Status	Outputs the status of the last data read from the socket receive queue. Range: 0 … 4 Data type: Uint32 ■ 0: Successful data read The relevant number of bytes is successfully read out of the socket receive queue. ■ 1: No data read The **Enable** inport is set to **Disabled**. ■ 2: No data read The socket receive queue is empty. ■ 3: No data read The socket is closed. ■ 4: No data read The model is in its initialization phase and no data has been received yet.
Received Bytes	Outputs the number of received UDP data bytes which are successfully read out of the socket receive queue. Range: 0 … 65507 Data type: UInt32 This outport is available if the **Enable variable frame size** option is set.
Received Frames	Outputs the number of received UDP data frames which are successfully read out of the socket receive queue. Data type: Double
Remote IP	Outputs the IP address of the remote device that is responsible for sending the last read UDP data frame.

Port	Description
Remote Port	This outport is useful if the **Accept any remote IP address** option is set, because it allows you to get information on the current sender. Range: 0 … 255 Data type: UInt8 This outport is available if the **Enable Remote IP port** option is set.
	Outputs the port number of the remote device that is responsible for sending the last read UDP data frame. Range: 0 … 65535 Data type: UInt16 This outport is available if the **Enable Remote Port port** is selected.

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to)
- Advanced page (refer to)

Unit Page (ETHERNET_UDP_RX_BLx)

Purpose

To receive data via Ethernet by using the UDP/IP protocol.

Dialog settings

Socket ID   Lets you specify the socket ID.

> **Note**
>
> There must be an **ETHERNET_UDP_SETUP_BLx** block in the model to get a socket ID.

Remote IP address   Displays the specified remote IP address of the selected socket.

Remote port   Displays the specified remote port of the selected socket.

Local port   Displays the specified local port number of the selected socket.

Frame size [1 … 65507] / Maximum frame size [1 … 65507]   Lets you specify the length of the UDP data frame in bytes to be read out of the socket receive queue.

Enable variable frame size   Lets you specify whether the amount of data to be read is always fix or the amount of data is variable.

Dependencies to the block's ports and settings are shown in the table below.

Enable Variable Frame Size	Dependencies
Off	Received Frames outport is available. Only those UDP packets are received that are exactly of the specified frame size. All other packets are dropped. The number of dropped packets is available at the RX Dropped Frames outport.
On	▪ The dialog text of the Frame size [1 … 65507] setting is changed to Maximum frame size [1 … 65507]. ▪ UDP packets are received if they are smaller than or equal to the specified maximum frame size. All other packets are dropped. The number of dropped packets is available at the RX Dropped Frames outport.

If the Enable interrupt on data receipt option is set, an interrupt is generated for a received UDP packet.

Related topics

References

Advanced Page (ETHERNET_UDP_RX_BLx)

Purpose

To specify whether to show or hide receive information.

Dialog settings

Socket ID   Displays the socket ID that you can select on the Unit page.

Enable Remote IP port   Lets you enable the Remote IP outport.

Enable Remote Port port   Lets you enable the Remote Port outport.

Related topics

References

ETHERNET_UDP_TX_BLx

Purpose   To transmit data via Ethernet using the UDP/IP protocol.

Where to go from here   Information in this section
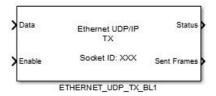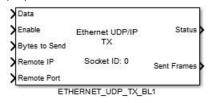
Information in other sections

Block Description (ETHERNET_UDP_TX_BLx)

Block   To give information on the appearance and purpose of the block.

Purpose   To send data via Ethernet using the UDP/IP protocol.

Description   With this block, you can send data via the UDP/IP protocol to a remote Ethernet
network device. The required data input must be a vector of UInt8 values.

I/O characteristics   The following table describes the ports of the block:

Port	Description
Input	
Data	Provides the data to be sent. The width of the port depends on the specified value of the Frame size parameter. Data type: UInt8
Enable	Controls the block's functionality. Data type: Boolean ▪ 0: Disabled (see the description below) ▪ 1: Enabled Note the following behavior if the block is disabled: ▪ The socket remains open. ▪ New data is not stored in the socket transmit queue.
Bytes to Send	Provides the possibility to determine the number of bytes that are written to the socket transmit queue. The maximum number of bytes to be sent is limited by the Frame size setting. Range: 0 … 65507 Data type: UInt32 This inport is available if the Enable variable frame size option is set.
Remote IP	Provides the IP address of the remote device the UDP data frame is to be sent to. Range: 0 … 255 Data type: UInt8 This inport is available if the Enable dynamic remote IP address option is set.
Remote Port	Provides the port number of the remote device the UDP data frame is to be sent to. Range: 1 … 65535 Data type: UInt16 This inport is available if the Enable dynamic remote port option is set.
Output	

Status	Outputs the status of the last data written to the socket transmit queue.
	Range: 0 … 4
	Data type: UInt32
	▪ 0: Successful data transfer
	  The relevant number of bytes is successfully written to the socket transmit queue.
	▪ 1: No data transfer
	  The Enable inport is set to Disabled or the Bytes to send inport is set to 0.
	▪ 2: No data transfer
	  The socket transmit queue is full.
	▪ 3: No data transfer
	  The socket is closed.
	▪ 4: No data transfer
	  The model is in its initialization phase and no data has been transmitted yet.

Port	Description
Sent Frames	Outputs the number of UDP data frames copied successfully to the socket transmit queue. However, the output does not show the number of frames that were really sent by the Ethernet interface. This outport is available if the **Enable Sent Frames port** option is set.

Dialog pages

The dialog settings can be specified on the following pages:

- Unit page (refer to Unit Page (ETHERNET_UDP_TX_BLx) on page 43)
- Advanced page (refer to Advanced Page (ETHERNET_UDP_TX_BLx) on page 44)

Unit Page (ETHERNET_UDP_TX_BLx)

Purpose

To send data via Ethernet by using the UDP/IP protocol.

Dialog settings

Socket ID   Lets you specify the socket ID.

> **Note**
>
> There must be an **ETHERNET_UDP_SETUP_BLx** block in the model to get a socket ID.

Enable dynamic remote IP address   Lets you specify whether to specify the IP address of the remote device via the **Remote IP** inport.

Enable dynamic remote port   Lets you specify whether to specify the port number of the remote device via the **Remote Port** inport.

Remote IP address   Displays the specified IP address of the selected socket.

Remote port   Displays the remote port specified for the selected socket.

Local port   Displays the local port number of the selected socket.

Frame size [1 … 65507]   Lets you specify the length of the UDP data frame in bytes.

Enable variable frame size   Lets you specify whether the number of bytes to be sent will be determined using the value of the **Bytes to Send** inport. If the **Enable variable frame size** option is not set, the payload length of the sent frame is always fixed according to the **Frame size** setting.

Related topics

Advanced Page (ETHERNET_UDP_TX_BLx)

Purpose

To specify whether to show or hide send information.

Dialog settings

Socket ID   Displays the socket ID that you can select on the Unit page.

Enable Sent Frames port   Lets you enable the Sent Frames outport.

Related topics

ETHERNET_HWINT_BLx

Purpose   To make the interrupts on data receipt available as trigger sources.

Where to go from here   Information in this section

Information in other sections

Block Description (ETHERNET_HWINT_BLx)

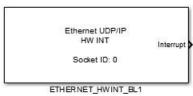**Block**   The block's appearance depends on the selected communication protocol.

TCP/IP

UDP/IP

Purpose   To make the interrupts on data receipt available as trigger sources.

Description

With this block, you can start the execution of a triggered subsystem, when data has been received on the specified socket.

You have to enable the interrupt generation for the specified socket in its related setup block. For the receipt of data, the model has to contain an RX block listening on the specified socket.

Multiple use of the same interrupt source is not allowed. If there are ETHERNET_HWINT_BLx blocks in your model with identical settings, the build process terminates with an error message.

I/O characteristics

The following table describes the ports of the block:

Port	Description
Output	
Interrupt	Trigger output. Data type: Boolean

Dialog pages

The dialog settings can be specified on the following pages:
- Unit page (refer to Unit Page (ETHERNET_HWINT_BLx) on page 46)

Related topics

References

Unit Page (ETHERNET_HWINT_BLx)..46

Unit Page (ETHERNET_HWINT_BLx)

Purpose

To configure the interface to be detected for providing interrupts.

Dialog settings

Communication protocol   Lets you specify either TCP/IP or UDP/IP as communication protocol.

Socket ID   Lets you select the socket of the specified protocol (TCP or UDP) from a list of sockets, which you have enabled for generating interrupts on data receipt.

Operation mode   Displays the operation mode specified for the selected socket. This is only valid for TCP/IP protocol, see ETHERNET_TCP_SETUP_BLx on page 19.

Remote IP address   Displays the specified remote IP address of the selected socket.

Remote port   Displays the specified remote port of the selected socket.

Local port   Displays the specified local port number of the selected socket.

Related topics

References