RTI CAN MultiMessage Blockset

Reference

For RTI CAN MultiMessage Blockset 5.6

Release 2021-A – May 2021

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

Contents

This reference provides a full description of the RTI CAN MultiMessage Blockset, which is supported by:

- DS1006-/DS1007-based systems with a DS2202, DS2210, DS2211, DS4302, and/or DS4505
- MicroAutoBox II
- MicroLabBox
- SCALEXIO systems with a DS2671 Bus Board, DS2672 Bus Module, DS6301 CAN/LIN Board, DS6341 CAN Board, and/or DS6342 CAN Board

Note

The RTI CAN MultiMessage Blockset is not supported by the MicroAutoBox III. If you work with the MicroAutoBox III, you must use the Bus Manager to implement CAN and CAN FD bus simulation.

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/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 CAN MultiMessage Blockset

Where to go from here

Information in this section

Overview of the RTI CAN MultiMessage Blockset

Introduction

The RTI CAN MultiMessage Blockset is a Simulink blockset for efficient and dynamic handling of complex CAN setups in application scenarios for rapid control prototyping (RCP) and hardware-in-the-loop (HIL) simulation. All the incoming RX messages and outgoing TX messages of an entire CAN controller can be controlled by a single Simulink block. The CAN communication can be configured via database files of different file types, for example, database container (DBC) files or AUTOSAR system description files.

The library contains the dSPACE RTI blocks to initialize a CAN controller as well as blocks to set up the entire CAN communication via the controller.

Supported dSPACE platforms

The RTI CAN MultiMessage Blockset is supported by the following dSPACE platforms:

- SCALEXIO systems with a DS2671 Bus Board, DS2672 Bus Module, DS6301 CAN/LIN Board, DS6341 CAN Board, and/or DS6342 CAN Board
- MicroAutoBox II
- MicroLabBox
- PHS-bus-based systems (DS1006 or DS1007 modular systems) containing one of the following I/O boards:
  - DS2202 HIL I/O Board
  - DS2210 HIL I/O Board
  - DS2211 HIL I/O Board
  - DS4302 CAN Interface Board
  - DS4505 Interface Board, if the DS4505 is equipped with DS4342 CAN FD Interface Modules

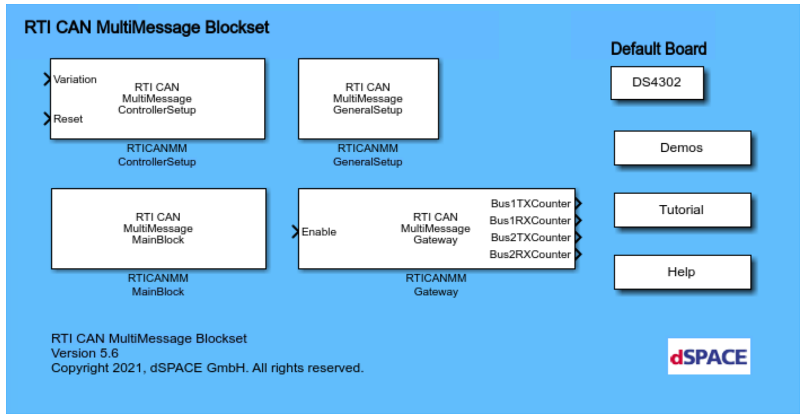
The dSPACE platforms provide 1 ... 4 CAN controllers (exception: DS6342 provides 1 ... 8 CAN controllers). A CAN controller performs serial communication according to the CAN protocol. To use a dSPACE board with CAN bus interface, you must configure the CAN controller – called the CAN channel – according to the application.

Note

The RTI CAN MultiMessage Blockset is not supported by the MicroAutoBox III. If you work with the MicroAutoBox III, you must use the Bus Manager to implement CAN and CAN FD bus simulation.

Opening the library

Double-click the RTI CANMM Blockset icon in the dSPACE_Blocksets library to open the rticanmmlib library:



Tip

You can open the RTI CAN MultiMessage Blockset directly by entering `rticanmm` in the MATLAB Command Window.

Available blocks

The following blocks belong to the RTI CAN MultiMessage Blockset:

- RTICANMM ControllerSetup on page 43
- RTICANMM GeneralSetup on page 38
- RTICANMM MainBlock on page 68
- RTICANMM Gateway on page 280

System target

Most of the basic settings in your modeling environment depend on the specified system target. For this reason, the system target file match fit the real-time hardware for which you want to generate code. The system target file is specified on the **Code Generation** page of the **Configuration Parameters** dialog. You open the **Configuration Parameters** dialog via the **Simulation** ribbon. On the ribbon, click **Prepare – Model settings**.

When selecting a dSPACE RTI platform support in MATLAB, the corresponding board-specific system target file (`rti<xxxx>.tlc`) is automatically selected in the Simulink® Coder™. If you work with a SCALEXIO system, you must select `dsrt.tlc` as the system target file.

Demo models

For Simulink models that show how to use the RTICANMM blocks, refer to the RTI demo library of the RTI CAN MultiMessage Blockset.

To use one of the demo models that come with the RTI CAN MultiMessage Blockset, you must first update the S-functions for all its RTICANMM blocks. You can use the **Create S-Function for all CAN Blocks** command from the **Options** menu of the RTICANMM GeneralSetup block for this. Refer to Options Menu (RTICANMM GeneralSetup) on page 39.

Tutorial

Opens the rticanmm_tutorial_lib sublibrary, which guides you through some typical use scenarios step by step and provides access to the database files used in the RTI CAN MultiMessage Blockset Tutorial 📖.

Tip

The tutorial is modular, i.e., you can choose the use scenarios you want to work through.

MEX compiler

To work with the RTI CAN MultiMessage Blockset, you need a MEX compiler.

Only the MinGW and Microsoft Visual Studio 2017 Professional compilers are supported as MEX compilers. For details and limitations, refer to Required C and C++ Compilers (Installing dSPACE Software 📖).

In MATLAB, the compiler has to be initialized via the `mex -setup` command.

Note

Whenever you find terms like <model> in this document, replace them with the actual name of your model. For example, if your Simulink model has the name `smd_simulation.slx` and you are asked to edit the file <model>_usr.c you actually have to edit the file `smd_simulation_usr.c`.

Related topics

Basics

Basics on Working with the RTI CAN MultiMessage Blockset

Creating a model

A Simulink model containing blocks from the RTI CAN MultiMessage Blockset is created in the same way as Simulink models without these blocks. However, there are some limitations to note when working with the RTI CAN MultiMessage Blockset. Refer to Limitations on page 309.

Before you can build a real-time application for a model with blocks from the RTI CAN MultiMessage Blockset, you have to let S-functions be created for the blocks of the RTI CAN MultiMessage Blockset. Creating these S-functions requires no handcoding.

Specifics for SCALEXIO systems If you model the CAN communication for a SCALEXIO system and want to perform model separation later on in ConfigurationDesk to execute the separated models on single cores of the SCALEXIO system, you need to know some specifics. Refer to Separating Models That Contain CAN or LIN Bus Communication (Model Interface Package for Simulink - Modeling Guide 📖).

Building the application

After you built the model and created the S-functions for the blocks of the RTI CAN MultiMessage Blockset, you must build the real-time application for the model and download it to the dSPACE real-time hardware.

- If you work with a MicroAutoBox II, MicroLabBox, or dSPACE modular system based on the DS1006, or DS1007, refer to How to Start the Build and Download (RTI and RTI-MP Implementation Guide 📖).
- The build process for SCALEXIO systems with a DS2671 Bus Board, DS2672 Bus Module, DS6301 CAN/LIN Board, DS6341 CAN Board, and/or DS6342 CAN Board is started in ConfigurationDesk. Refer to Building Real-Time Applications (ConfigurationDesk Real-Time Implementation Guide 📖).

A bus configuration file (CANCFG file) is created together with the real-time application. ControlDesk uses the CANCFG file to generate the bus configuration. You can use the bus configuration to control the dSPACE hardware's CAN communication from within ControlDesk while the real-time application is running.

For more information on working with the Bus Navigator in ControlDesk, refer to ControlDesk Bus Navigator 📖.

Further information

For more information on how to work with the RTI CAN MultiMessage Blockset, it is recommended to read the following documents:

- RTI CAN MultiMessage Blockset Tutorial 📖

  The tutorial guides you through your first steps with the RTI CAN MultiMessage Blockset. It is divided into lessons in which you learn typical procedures step-by-step. The tutorial comes with several models including the lesson results.
- If you work with a SCALEXIO system, refer to CAN Bus Connection (SCALEXIO – Hardware and Software Overview 📖). You get an overview of the workflow steps that are necessary to connect an ECU to a CAN bus simulated by the SCALEXIO system.

Related topics

Basics

Overview of the RTI CAN MultiMessage Blockset.. 13

Aspects of Miscellaneous Supported AUTOSAR Features

Container IPDUs and contained IPDUs

The RTI CAN MultiMessage Blockset supports *container IPDUs*. A container IPDU is an IPDU that contains one or several smaller *contained IPDUs*. A container IPDU can be used to transmit multiple contained IPDUs in one container IPDU in a single CAN FD message on the CAN bus.

Contained IPDUs are not sent directly on the bus. When a trigger condition for a contained IPDU is fulfilled (the RTICANMM standard trigger conditions, such as CycleTime and Kickout, are used here), the contained IPDU is written to its container IPDU. The container IPDU itself is sent on the bus when one of its trigger conditions is met. Refer to Triggering of container IPDUs on page 18. When a container IPDU is mapped to a frame, all its contained IPDUs are included in that frame as well. According to AUTOSAR, a contained IPDU must be contained in exactly one container IPDU.

During database import, the RTI CAN MultiMessage Blockset automatically names the container IPDUs `Cntr_<IPDU name>`. Contained IPDUs are named `Cntd_<IPDU name>`. The name extension makes it easier for you to manage the PDUs on the RTI CAN MultiMessage Blockset dialog pages. Adding the prefixes automatically groups the PDUs by type because the messages are sorted in alphabetical order.

Dynamic and static container layout The container layout of container IPDUs can be organized dynamically or statically:

Container Layout	Description
Dynamic container layout	The contained IPDUs do not have a predefined position in the container IPDU. The receiving ECU identifies the contained IPDUs via their header IDs. For this purpose, two header types are available (`SHORT_HEADER`, `LONG_HEADER`). A contained IPDU can provide a separate header ID for each of these header types. The container IPDU defines the required header type: The ID of the required header type must be specified for each contained IPDU, and the receiving ECU evaluates only these IDs. The RTI CAN MultiMessage Blockset supports only the short header type.
Static container layout	The contained IPDUs have a fixed position within the container IPDU. The receiving ECU identifies the contained IPDUs unambiguously by their position within the container IPDU. For this purpose, the header type of the container IPDU must be set to `NO_HEADER` and an offset value must be specified for each contained IPDU. The offset value determines the position of the contained IPDU within the container IPDU.

Triggering of container IPDUs The triggering of container IPDUs depends on various timing and triggering conditions that can be specified for container and/or contained IPDUs in the communication matrix. For example, a container IPDU can be transmitted after the following events:

- A specified timeout value elapses.
- Immediately after the first contained IPDU is added.
- A specific contained IPDU triggers its transmission.

Queuing of contained IPDUs For each contained IPDU that is transmitted via the bus, the communication matrix must specify the collection semantics:

- With a queued semantics, multiple instances of the contained IPDU can be added to one container IPDU.

  A queued semantics is supported only for contained IPDUs that are included in container IPDUs with a dynamic container layout.
- With a last-is-best semantics, only one instance of the contained IPDU can be added to one container IPDU. In this case, the data of the contained IPDU is buffered and only the latest data is added to the container IPDU just before it is transmitted.

  For contained IPDUs that are included in container IPDUs with a static container layout, only the last-is-best semantics is supported.

The RTI CAN MultiMessage Blockset supports both semantics. However, when a container IPDU is received that contains several instances of a contained IPDU, only the last received instance can be displayed for the contained IPDU. However, the container IPDU layout displays all received contained IPDU instances as raw data.

Identifying updated contained IPDUs in container IPDUs with static container layout When a container IPDU with a static container layout is transmitted, some of its contained IPDUs might not have been updated since the previous transmission and therefore provide old data. There are two ways to identify the update state:

Setting	Description
Update bit of contained IPDU	Contained IPDUs can provide an update bit. The bit is set if a contained IPDU has been updated between two transmissions of the related container IPDU.
Unused bit pattern of container IPDU	Container IPDUs can provide an unused bit pattern. If a contained IPDU of the container IPDU is not updated, the content of the contained IPDU is cleared and replaced with the unused bit pattern.

The terms 'container IPDU' and 'contained IPDU' are terms according to AUTOSAR.

When you work with container IPDUs and contained IPDUs, some limitations apply. Refer to Limitations with RTICANMM on page 309.

Secure onboard communication for PDUs

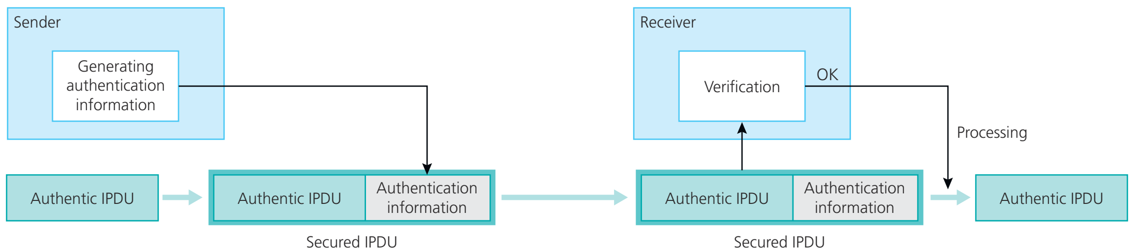
The RTI CAN MultiMessage Blockset supports secure onboard communication (SecOC) according to AUTOSAR 4.3.0. Secure onboard communication provides authentication mechanisms to identify PDU data that was modified or transmitted without authorization, e.g., due to a replay attack.

PDUs for SecOC In secure onboard communication scenarios, the following PDUs are used:

- *Authentic IPDUs*: Authentic IPDUs contain the payload, which will be secured by the authentication mechanisms of SecOC. The required authentication information itself is not included in authentic IPDUs but in secured IPDUs.

According to AUTOSAR, authentic IPDUs can be ISignal IPDUs, container IPDUs, or multiplexed IPDUs, for example.

- *Secured IPDUs* and *cryptographic IPDUs*: Secured IPDUs are used to secure the payload of authentic IPDUs, i.e., they contain the required authentication information. The sender includes the authentication information in the secured IPDU and the receiver verifies the received authentication information, as shown in the following example.



The handling of the related authentic IPDUs depends on whether secured IPDUs are configured as cryptographic IPDUs:

  - If a secured IPDU is configured as a cryptographic IPDU, the related authentic IPDU is not included in the secured IPDU. In this case, the authentic IPDU and the cryptographic IPDU are transmitted separately via the bus. The receiver verifies the authentication information each time the authentic IPDU or cryptographic IPDU is received.
  - If a secured IPDU is not configured as a cryptographic IPDU, the related authentic IPDU is directly included in the secured IPDU. In this case, only one PDU is exchanged via the bus.

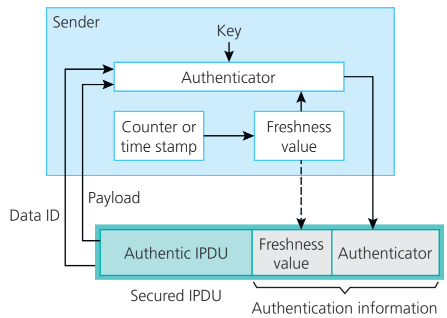
Authentication information According to AUTOSAR, there are various ways to generate and verify authentication information. To generate and verify specific authentication information, OEM-specific implementations are required. In general, authentication information that is included in a secured IPDU consists of a freshness value and an authenticator:

- *Freshness value*: The freshness value is a monotonously increasing value. Depending on the OEM-specific implementation, the freshness value can be a counter value or a time stamp value. The freshness value is required for calculating the authenticator. Additionally, the freshness value can directly be included in the secured IPDU. If it is, it can be included completely or in part, i.e., as a truncated freshness value. The truncated freshness value contains the lower bits of the complete freshness value (the more significant bits are discarded). The size of the truncated value is specified in the database.

  The RTI CAN MultiMessage Blockset supports only time stamp values as freshness values.
- *Authenticator*: The authenticator is calculated according to OEM-specific algorithms and keys. For calculating the authenticator, the following data is required:
  - Data identifier of the secured IPDU
  - Payload of the authentic IPDU
  - Freshness value

The authenticator is included in the secured IPDU either completely or in part, i.e., as a truncated authenticator. The truncated authenticator value contains the higher bits of the complete authenticator value (the less significant bits are discarded). The size of the truncated authenticator is specified in the database.

The following illustration is an example of authentication information in a secured IPDU.



Secured PDU header A secured IPDU can contain the optional *secured PDU header* part. The secured PDU header indicates the length of the authentic IPDU. Different header types are possible: `noHeader`, `securedPduHeader08Bit`, `securedPduHeader16Bit`, and `securedPduHeader32Bit`.

The following illustration shows the structure of a secured IPDU with a secured PDU header:

'Secured IPDU' is a term according to AUTOSAR.

Working with secured IPDUs in the RTI CAN MultiMessage Blockset After you specify a database for an RTICANMM MainBlock, the secured IPDUs are displayed and can be used on the dialog pages. For secured IPDUs, the (truncated) freshness and authenticator signals are automatically appended to the authentic IPDUs during database import. The signals are named `Signal_<Authentic IPDU short name>_Freshness` and `Signal_<Authentic IPDU short name>_Auth`. For secured IPDUs with a secured PDU header whose header type is set to any value but `noHeader`, secured PDU header signals named `Signal_<Authentic IPDU short name>_Header` are automatically generated and added at the beginning of the secured IPDUs during database import.

To implement secure onboard communication, you must enable SecOC support and provide the OEM-specific implementation for generating authentication information via user code. Refer to Secure Onboard Communication Page (RTICANMM MainBlock) on page 207.

You get access to authentication data and status information on the verification of secured IPDUs at run time in ControlDesk. Suitable TRC file entries for reading and manipulating the values are automatically generated and included in the TRC file if secure onboard communication is enabled in the RTICANMM MainBlock and if a secured IPDU is selected on the TX Messages Page (RTICANMM MainBlock) or RX Messages Page (RTICANMM MainBlock).

When you work with secure onboard communication, some limitations apply. Refer to Limitations with RTICANMM on page 309.

Global time synchronization

The RTI CAN MultiMessage Blockset supports *global time synchronization (GTS)* according to AUTOSAR 4.3.0. Global time synchronization means providing and distributing synchronized times in a bus network across all ECUs.

Time bases A synchronized time is called a *time base*. A time base is a unique time entity characterized by the progression of time, the ownership, and the reference to the physical world. There are two types of time bases:

- *Synchronized time base*: A synchronized time base is a time base that is synchronized with other time bases at different processing entities. These synchronized time bases constitute a global time. A synchronized time base does not depend on the existence of another time base.
- *Offset time base*: An offset time base is a time base that depends on a particular synchronized time base and holds an offset value with respect to that time base.

Time bases can provide relative times (e.g., time after power-up of an ECU or an operating hours counter) or absolute times (e.g., a UTC time). Because there can be more than just one time inside a vehicle, multiple time bases can exist in parallel. Time bases can be based on other time bases. Per ECU, there can be up to 16 synchronized time bases and offset time bases each.

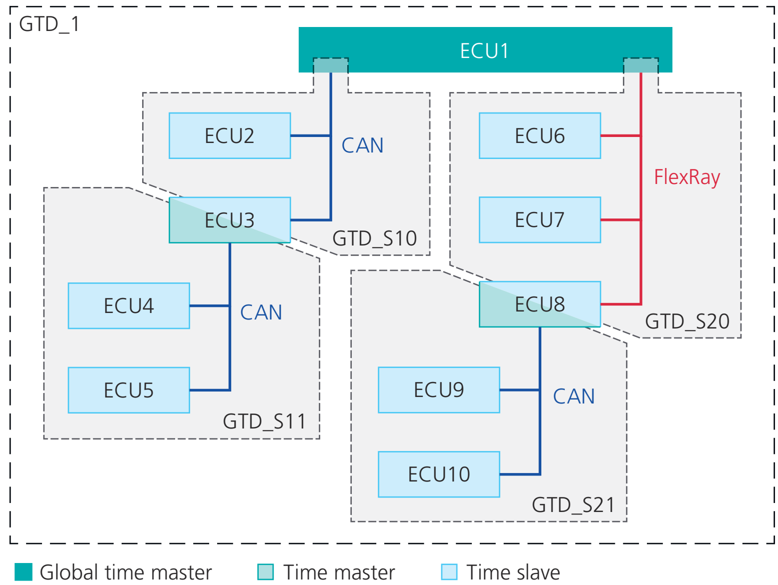
The RTI CAN MultiMessage Blockset does not support offset time bases.

Global time domains All the components (e.g., nodes and communication systems) that are linked to the same synchronized time base constitute a *global time domain*. There can be several global time domains in a bus network. Global time domains are distinguished by *domain identifiers* as follows:

- Global time domains that use different synchronized time bases have different domain identifiers.
- Global time domains that use the same synchronized time base have the same domain identifier.

Global time domains that use the same synchronized time base are connected by the means of *subdomains*. A subdomain denotes which components (e.g., nodes) are linked to the related time base where the scope is limited to exactly one communication cluster. A global time domain can declare several other global time domains as its subdomains.

The following illustration shows an example for the global time distribution in a network topology with four communication clusters and five global time domains. *GTD_S10* and *GTD_S20* are subdomains of *GTD_1*. *GTD_S11* and *GTD_S21* are subdomains of *GTD_S10* and *GTD_S20*, respectively. All global time domains use the same domain identifier.



Depending on its role in the network, an ECU can be a time master and/or a time slave.

Time master, time slave, and time gateway Global time synchronization is based on a master-slave system. An ECU can be both time master and time slave at the same time. Additionally, if synchronized times are processed via multiple communication clusters, an ECU can act as a time gateway.

- A *time master* is an entity that is the master for a certain time base and distributes this time base to a set of time slaves within a certain cluster of a bus network. If a time master is also the global owner and origin of the time base, it is called the *global time master*.
- A *time slave* is an entity that is the recipient of a certain time base within a certain cluster of a bus network. Each time slave has its own time base instance, i.e., a local instance of the time. Whenever a time slave receives a new time from the time master, the time slave interpolates the time until the next synchronization with the time master.
- A *time gateway* is a time base instance that receives the synchronized time as a time slave from a time master on one cluster and then forwards it to another cluster as a time master. A time gateway typically consists of one time slave and one or more time masters.

Time synchronization messages The messages that distribute the time information are called *time synchronization messages*.

With time synchronization over CAN, each time synchronization message consists of two messages that are transmitted separately: a time synchronization (SYNC) message and a follow-up (FUP) message. The latter completes the synchronization process. SYNC and FUP messages are transmitted using the same CAN identifier. SYNC and FUP messages are identified by their message type, and their message layout depends on whether they are CRC-secured.

The following illustration shows the structures of a CRC-secured time synchronization message (message type: 0x20) and follow-up message (message type: 0x28):

SYNC:

Message type	CRC	Time domain	E2E sequence counter	User byte 0	Seconds
8 bits	*8 bits*	*4 bits*	*4 bits*	*8 bits*	*32 bits*
Byte 0	Byte 1	Byte 2		Byte 3	Byte 4 ... Byte 7

MSB LSB

FUP:

Message type	CRC	Time domain	E2E sequence counter	Reserved	SGW (1 bit)	Overflow of seconds (2 bits)	Nanoseconds
8 bits	*8 bits*	*4 bits*	*4 bits*	*5 bits*			*32 bits*
Byte 0	Byte 1	Byte 2		Byte 3			Byte 4 ... Byte 7

MSB LSB

The following illustration shows the structures of an unsecured time synchronization message (message type: 0x10) and follow-up message (message type: 0x18):

SYNC:

Message type	User byte 1	Time domain	E2E sequence counter	User byte 0	Seconds
8 bits	*8 bits*	*4 bits*	*4 bits*	*8 bits*	*32 bits*
Byte 0	Byte 1	Byte 2		Byte 3	Byte 4 ... Byte 7

MSB LSB

FUP:

Message type	User byte 2	Time domain	E2E sequence counter	Reserved	SGW (1 bit)	Overflow of seconds (2 bits)	Nanoseconds
8 bits	*8 bits*	*4 bits*	*4 bits*	*5 bits*			*32 bits*
Byte 0	Byte 1	Byte 2		Byte 3			Byte 4 ... Byte 7

MSB LSB

Working with global time synchronization in the RTI CAN MultiMessage Blockset To configure time synchronization via CAN, you must select the corresponding time synchronization messages on the TX Messages Page (RTICANMM MainBlock) or RX Messages Page (RTICANMM MainBlock). A time master for a time domain is simulated by selecting a corresponding SYNC time synchronization message on the TX Messages page. A time slave for a time domain is simulated by selecting the corresponding SYNC time synchronization messages on the RX Messages page. You can identify the signals of TX GTS messages on the Signal Default Manipulation Page (RTICANMM MainBlock). When the **User-defined defaults** checkbox is selected, the signals' default manipulation option is **GTS**.

The cycle time variable can be used with GTS messages as with other message types. The cycle time of a GTS message determines the time difference between two sendings of the SYNC message. After a SYNC message is sent, the corresponding FUP message is sent after the FUP offset. For database versions that do not provide the `followUpOffset` attribute (e.g., AUTOSAR R20-11), the `debounceTime` value is used instead. However, if the cycle time is shorter than the FUP offset, only the SYNC messages are transmitted.

You get access to GTS data and status information at run time in the Simulink model and via ControlDesk. Suitable MainBlock outports and TRC file entries for reading and manipulating the values are automatically generated if GTS messages are selected in the RTICANMM MainBlock on the TX Messages page or RX Messages page.

For time slaves, the TRC variable `<slave ECU name>_TD<time domain ID>_ValidationEnable` is generated, which lets you override the necessary condition for updating the time in the time base manager instance. For more information, refer to Block Description (RTICANMM MainBlock) on page 69.

When you work with global time synchronization, some limitations apply. Refer to Limitations with RTICANMM on page 309.

End-to-end protection for ISignal groups

The RTI CAN MultiMessage Blockset supports end-to-end (E2E) communication protection according to the following AUTOSAR end-to-end protection profiles (E2E profiles):

- Profile 01
- Profile 02
- Profile 05
- Profile 06
- Profile 11
- Profile 22

The RTI CAN MultiMessage Blockset supports E2E transformers for profile 02, profile 05, and profile 06 for calling the end-to-end communication protection library (E2E library).

The end-to-end protection must be specified in the communication matrix.

For more information, refer to Checksum Definition Page (RTICANMM MainBlock) on page 200.

Related topics

References

Basics on Working with CAN FD

Introduction

Using the CAN FD protocol allows data rates higher than 1 Mbit/s and payloads longer than 8 bytes per message.

Basics on CAN FD

CAN FD stands for *CAN with Flexible Data Rate*. The CAN FD protocol is based on the CAN protocol as specified in ISO 11898-1. Compared with the classic CAN protocol, CAN FD comes with an increased bandwidth for the serial communication. The improvement is based on two factors:

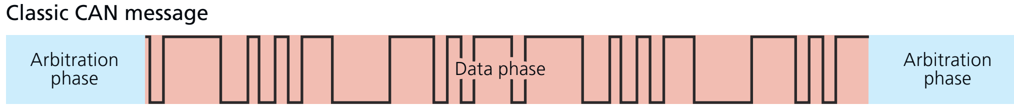
- The CAN FD protocol allows you to use CAN messages with longer data fields (up to 64 bytes).
- The CAN FD protocol allows you to use a higher bit rate (typically higher by a factor of 8). It is possible to switch inside the message to the faster bit rate.

Arbitration phase and data phase CAN FD messages consist of two phases: a data phase and an arbitration phase. The data phase spans the phase where the data bits, CRC and length information are transferred. The rest of the frame, outside the data phase, is the arbitration phase. The data phase can be configured to have a higher bit rate than the arbitration phase.

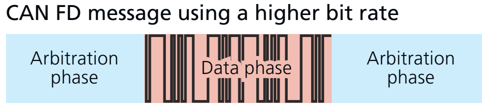
CAN FD still uses the CAN bus arbitration method. During the arbitration process, the standard data rate is used. After CAN bus arbitration is decided, the data rate can be increased. The data bits are transferred with the preconfigured higher bit rate. At the end of the data phase, CAN FD returns to the standard data rate.

The following illustration shows a classic CAN message, a CAN FD message using a higher bit rate during the data phase, and a CAN FD message with longer payload using a higher bit rate. You can see the implications of the CAN FD features: The arbitration phases are identical in all cases, because the standard bit rate is always used. The lengths of the data phases differ depending on the payload length and bit rate used.

Classic CAN message



CAN FD message using a higher bit rate



CAN FD message with longer payload using a higher bit rate

CAN FD protocols

Currently, there are two CAN FD protocols on the market, which are not compatible with each other.

- The *non-ISO CAN FD protocol* represents the original CAN FD protocol from Bosch.
- The *ISO CAN FD protocol* represents the CAN FD protocol according to the ISO 11898-1:2015 standard.

Compared to the non-ISO CAN FD protocol, the ISO CAN FD protocol comes with an improved failure detection capability.

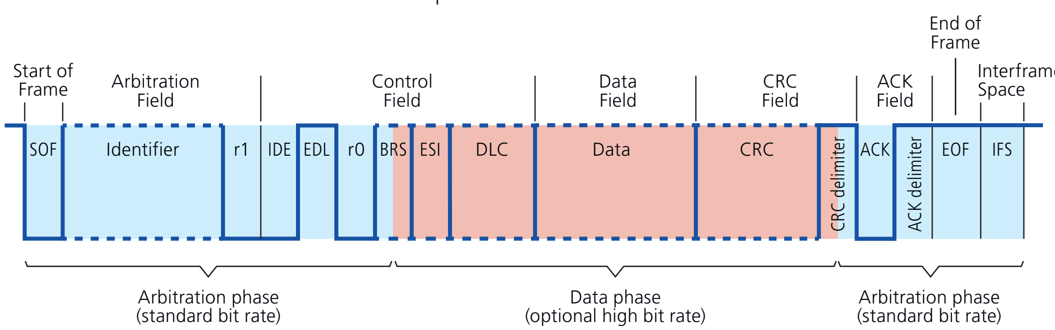
The RTI CAN MultiMessage Blockset supports both CAN FD protocols.

CAN FD message characteristics

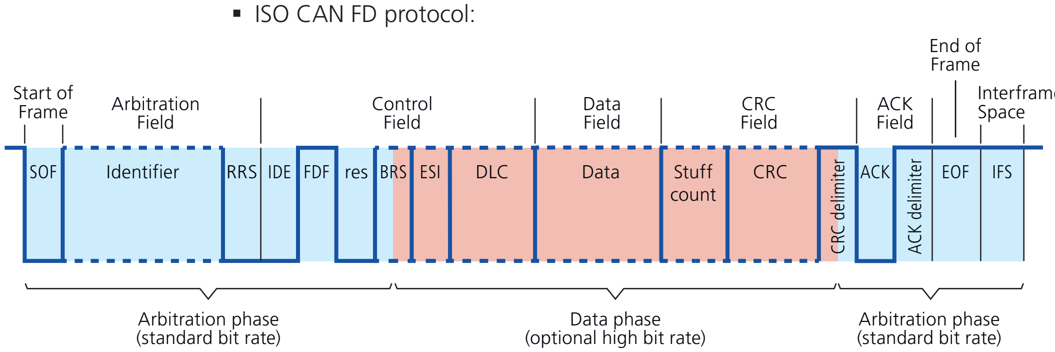
In principle, CAN FD messages and CAN messages consist of the same elements and have the same message structure.

For an overview of the fields of a CAN FD message and the message components for each of the two CAN FD protocols, refer to the following illustration:

- Non-ISO CAN FD protocol:



- ISO CAN FD protocol:



There are some differences between CAN FD messages and CAN messages:

- The Data field and the CRC field of CAN FD messages can be longer. The maximum payload length of a CAN FD message is 64 bytes.
- The Control fields are different:
  - A classic CAN message always has a dominant (= 0) reserved bit immediately before the data length code.

    In a CAN FD message, this bit is always transmitted with a recessive level (= 1) and is called *EDL* (Extended Data Length) or *FDF* (FD Format), depending on the CAN FD protocol you are working with. So CAN messages and CAN FD messages are always distinguishable by looking at the EDL/FDF bit. A recessive EDL/FDF bit indicates a CAN FD message, a dominant EDL/FDF bit indicates a CAN message.

In CAN FD messages, the EDL/FDF bit is always followed by a dominant reserved bit (*r0/res*), which is reserved for future use.

- A CAN FD message has the additional *BRS* (Bit Rate Switching) bit, which allows you to switch the bit rate for the data phase. A recessive BRS bit switches from the standard bit rate to the preconfigured alternate bit rate. A dominant BRS bit means that the bit rate is not switched and the standard bit rate is used.
- A CAN FD message has the additional *ESI* (Error State Indicator) bit. The ESI bit is used to identify the error status of a CAN FD node. A recessive ESI bit indicates a transmitting node in the 'error active' state. A dominant ESI bit indicates a transmitting node in the 'error passive' state.

- Since CAN FD messages can contain up to 64 data bytes, the coding of the DLC (data length code) has been expanded. The following table shows the possible data field lengths of CAN FD messages and the corresponding DLC values.

DLC	**Number of Data Bytes**
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	12
1010	16
1011	20
1100	24
1101	32
1110	48
1111	64

If necessary, padding bytes are used to fill the data field of a CAN FD message to the next greater possible DLC value.

For classic CAN messages, the DLC values 1000 ... 1111 are interpreted as 8 data bytes.

- (Valid for the ISO CAN FD protocol only) The CRC fields are different:
  - The CRC field of a CAN FD message was extended by a stuff count before the actual CRC sequence. The stuff count consists of a 3-bit stuff bit counter (reflects the number of the data-dependent dynamic stuff bits (modulo 8)), and an additional parity bit to secure the counter.
  - The start value for the CRC calculation was changed from '0...0' to '10...0'.

Activating CAN FD mode in the RTI CAN MultiMessage Blockset

To ensure that CAN FD messages are properly transmitted and received during run time, the CAN FD mode must be enabled at two places in the RTI CAN MultiMessage Blockset: in the MainBlock and at the CAN controller selected in the MainBlock. To do so, perform the following steps:

- In the ControllerSetup block, select the CAN FD protocol to be used. Refer to Setup Page (RTICANMM ControllerSetup) on page 48.
- In the MainBlock, select the **CAN FD support** checkbox. Refer to General Settings Page (RTICANMM MainBlock) on page 100.

To monitor CAN FD messages, it is sufficient to enable CAN FD support in the ControllerSetup block.

Supported database file types

To work with CAN FD messages, you need a suitable database file containing descriptions in the CAN FD format. The RTI CAN MultiMessage Blockset supports CAN FD for the following database file types:

- DBC file
- AUTOSAR system description file
- FIBEX file (FIBEX 4.1.2 only)

CanFrameTxBehavior and CanFrameRxBehavior attributes In AUTOSAR and FIBEX files, the `CanFrameTxBehavior` and/or `CanFrameRxBehavior` attributes of a message can be defined to specify whether the message is to be treated as a CAN FD message or classic CAN 2.0 message. The RTI CAN MultiMessage Blockset evaluates the attribute as follows:

- If the `CanFrameTxBehavior` attribute is defined for a message in the database file, RTICANMM uses this setting for the message on the CAN bus for both directions, i.e., for sending and receiving the message.
- If the `CanFrameTxBehavior` attribute is not defined in the database for a message, RTICANMM uses the `CanFrameRxBehavior` setting of the message for sending and receiving the message.

Supported dSPACE platforms

The RTI CAN MultiMessage Blockset supports working with CAN FD messages for the following dSPACE hardware:

- SCALEXIO systems with a DS2671 Bus Board, DS2672 Bus Module, DS6301 CAN/LIN Board, DS6341 CAN Board, or DS6342 CAN Board
- The following dSPACE platforms if they are equipped with DS4342 CAN FD Interface Modules:
  - DS1006 modular system with DS4505 Interface Board
  - DS1007 modular system with DS4505 Interface Board
  - MicroAutoBox II in the following variants:
    - 1401/1507
    - 1401/1511/1514
    - 1401/1513/1514

When you connect a DS4342 CAN FD Module to a CAN bus, you must note some specific aspects (such as bus termination and using feed-through bus lines). For more information, refer to:

- PHS-bus-based system with DS4505: DS4342 Connections in Different Topologies (PHS Bus System Hardware Reference 📖)
- MicroAutoBox II: DS4342 Connections in Different Topologies (MicroAutoBox II Hardware Installation and Configuration Guide 📖)

Working with CAN messages and CAN FD messages

Both messages in CAN format and in CAN FD format can be transmitted and received via the same network.

Related topics

References

Setup Page (RTICANMM ControllerSetup).. 48

Basics on Working with a J1939-Compliant DBC File

Introduction

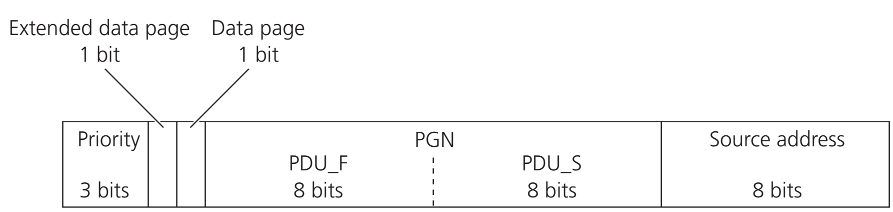
J1939 is a vehicle CAN bus standard defined by the Society of Automotive Engineers (SAE). It is used for communication in heavy-duty vehicles, for example, for communication between a tractor and its trailer.

The RTI CAN MultiMessage Blockset supports you in working with J1939-compliant DBC files.

Broadcast and peer-to-peer communication

CAN message identifier for J1939 Standard CAN messages use an 11-bit message identifier (CAN 2.0 A). J1939 messages use an extended 29-bit message identifier (CAN 2.0 B).

The 29-bit message identifier is split into different parts (according to SAE J1939/21 Data Link Layer):



29-bit CAN message ID

- The 3-bit *priority* is used during the arbitration process. A value of 0 represents the highest priority, a value of 7 represents the lowest priority.

- The 1-bit *extended data page* can be used as an extension of the PGN. The RTI CAN MultiMessage Blockset lets you specify whether to use the extended data page bit this way. Refer to Code Options Page (RTICANMM MainBlock) on page 257.
- The 1-bit *data page* is a selector for the PDU_F in the PGN. It can be taken as an extension of the PGN. The RTI CAN MultiMessage Blockset uses the data page bit in this way.
- The 16-bit *PGN* is the parameter group number. It is described in this section.
- The 8-bit *source address* is the address of the transmitting network node.

Parameter group number (PGN) A 16-bit number in the 29-bit message identifier of a CAN message defined in a J1939-compliant DBC file. Each PGN references a *parameter group* that groups parameters and assigns them to the 8-byte data field of the message. A parameter group can be the engine temperature including the engine coolant temperature, the fuel temperature, etc. PGNs and parameter groups are defined by the SAE (see SAE J1939/71 Vehicle Application Layer).

The first 8 bits of the PGN represent the *PDU_F* (Protocol Data Unit format). The PDU_F value specifies the communication mode of the message (peer-to-peer or broadcast). The interpretation of the *PDU_S* value (PDU-specific) depends on the PDU_F value. For messages with a PDU_F < 240 (peer-to-peer communication, also called PDU1 messages), PDU_S is not relevant for the PGN, but contains the destination address of the network node that receives the message. For messages with a PDU_F ≥ 240 (broadcast messages, also called PDU2 messages), PDU_S specifies the second 8 bits of the PGN and represents the group extension. A group extension is used to increase the number of messages that can be broadcast in the network.

PDU_F (first 8 bits)	PDU_S (second 8 bits)	Communication Mode
< 240	Destination address	Peer-to-peer (message is transmitted to one destination network node)
≥ 240	Group extension	Broadcast (message is transmitted to any network node connected to the network)

Message attributes in J1939-compliant DBC files

A message described in a J1939-compliant DBC file is described by the ID attribute and others.

DBC files created with CANalyzer 5.1 or earlier In a DBC file created with CANalyzer 5.1 or earlier, the ID attribute describing a message actually is the *message PGN*. Thus, the ID provides no information on the source and destination of the message. The source and destination can be specified by the *J1939PGSrc* and *J1939PGDest* attributes.

DBC files created with CANalyzer 5.2 or later In a DBC file created with CANalyzer 5.2 or later, the ID attribute describing a message actually is the *CAN message ID, which consists of the priority, PGN, and source address*. Thus, the ID provides information on the source and destination of the message. Further senders can be specified for a message in Vector Informatik's CANdb++ Editor

(*_BO_TX_BU* attribute). When a DBC file is read in, RTICANMM automatically creates new instances of the message for its other senders.

Source/destination mapping

Messages in a J1939-compliant DBC file that are described only by the PGN have no *source/destination mapping*. Messages that are described by the CAN message ID (consisting of the priority, PGN, and source address) have source/destination mapping.

Tip

The RTI CAN MultiMessage Blockset lets you specify source/destination mapping for messages that are described only by the PGN. The mapping can be specified in the RTICANMM MainBlock or in the DBC file using the *J1939Mapping* attribute.

Container and instance messages

The RTI CAN MultiMessage Blockset distinguishes between *container* and *instance messages* when you work with a J1939-compliant DBC file:

Container message A J1939 message defined by its PGN (and Data Page bit). A container can receive all the messages with its PGN in general. If several messages are received in one sampling step, only the last received message is held in the container. Container messages can be useful, for example, when you configure a gateway with message manipulation.

Instance message A J1939 message defined by its PGN (and Data Page bit), for which the source (transmitting) network node and the destination (receiving) network node (only for peer-to-peer communication) are defined.

Note

The RTI CAN MultiMessage Blockset only imports instances for which valid source node and destination node specifications are defined in the DBC file. In contrast to instances, containers are imported regardless of whether or not valid source node and destination node specifications are known for them during the import. This lets you configure instances in the RTI CAN MultiMessage Blockset.

There is one container for each PGN (except for proprietary PGNs). If you work with DBC files created with CANalyzer 5.1 or earlier, the container can be clearly derived from the DBC file according to its name. With DBC files created with CANalyzer 5.2 or later, several messages with the same PGN might be defined. In this case, either the message with the shortest name or an additionally created message (named `CONT_<shortest_message_name>`) is used as the container for the PGN. The RTI CAN MultiMessage Blockset lets you specify the container type in the RTICANMM MainBlock. If several messages fulfill the condition of the shortest name, the one that is listed first in the DBC file is used. For messages with proprietary PGNs, each message is its own container (because proprietary

PGNs can have different contents according to their source and destination nodes).

Network management

The J1939 CAN standard defines a multimaster communication system with decentralized network management. J1939 network management defines the automatic assignment of network node addresses via address claiming, and provides identification for each network node and its primary function.

Each network node must hold exactly one 64-bit name and one associated address for identification.

Address The 8-bit network node *address* defines the source or destination for J1939 messages in the network. The address of a network node must be unique. If there is an address conflict, the network nodes try to perform dynamic network node addressing (address claiming) to ensure unique addresses, if this is enabled for the network nodes.

The J1939 standard reserves the following addresses:

- Address 0xFE (254) is reserved as the 'null address' that is used as the source address by network nodes that have not yet claimed an address or have failed to claim an address.
- Address 0xFF (255) is reserved as the 'global address' and is exclusively used as a destination address in order to support message broadcasting (for example, for address claims).

The RTI CAN MultiMessage Blockset does not allow J1939 messages to be sent from the null or global addresses.

Note

The RTI CAN MultiMessage Blockset interprets attributes in the DBC file like this:

- In a DBC file created with CANalyzer 5.1 or earlier, the *name* network node attributes and the *J1939PGSrc* and *J1939PGDest* message attributes are read in. The J1939PGSrc attribute is interpreted as the address of the node that sends the message, the J1939PGDest attribute is interpreted as the address of the node that receives the message.
- In a DBC file created with CANalyzer 5.2 or later, the *name* and *NMStationAddress* network node attributes are read in. The NMStationAddress attribute is interpreted as the network node address.

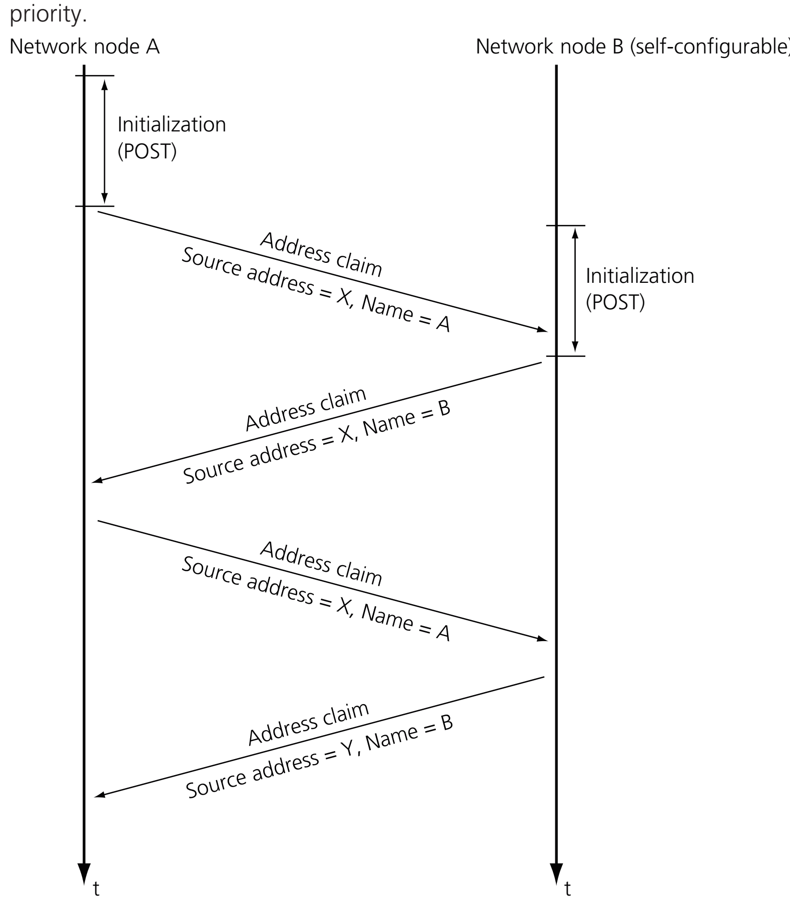
Name The J1939 standard defines a 64-bit *name* to identify each network node. The name indicates the main function of the network node with the associated address and provides information on the manufacturer.

Arbitrary Address Capable	Industry Group	Vehicle System Instance	Vehicle System	Reserved	Function	Function Instance	ECU Instance	Manufacturer Code	Identity Number
1 bit	3 bit	4 bit	7 bit	1 bit	8 bit	5 bit	3 bit	11 bit	21 bit

Address claiming The J1939 standard defines an address claiming process in which addresses are autonomously assigned to network nodes during network initialization. The process ensures that each address is unique.

Each network node sends an address claim to the CAN bus. The nodes receiving an address claim verify the claimed address. If there is an address conflict, the network node with the lowest 64-bit name value (highest priority) gets the claimed address. The other network nodes must claim different addresses.

The following illustration shows the address claiming process with two network nodes claiming the same address. Network node A has a 64-bit name of higher priority.



The following steps are performed in the address claiming procedure:

- Node A starts initialization and the power-on self-test (POST).
- While node B performs initialization and POST, node A sends its address claim message.
- After performing initialization and POST, node B sends its address claim message, trying to claim the same source address as node A.

- In response to the address claim message of node B, the 64-bit names of the network nodes are compared. Because the name of network node A has a higher priority, network node A succeeds and can use the claimed address. Node A sends its address claim message again.
- Network node B receives the address claim message and determines that node A's name has higher priority. Node B claims a different address by sending another address claim message.

The RTI CAN MultiMessage Blockset supports J1939 network management including address claiming for self-configurable address network nodes. This allows network nodes simulated by the RTI CAN MultiMessage Blockset to change their addresses, if necessary, and to update their internal address assignments if addresses of external network nodes are changed.

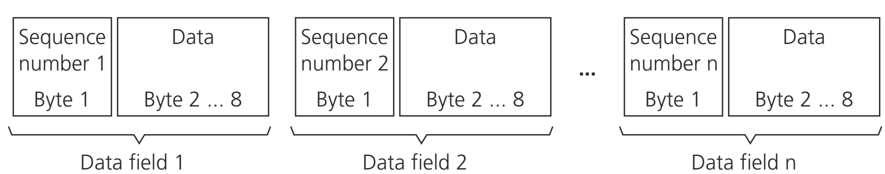
Note

The RTI CAN MultiMessage Blockset supports network management only for network nodes for which network addresses are contained in the DBC file and that have unique 64-bit name identifiers. The node configuration is taken directly from the DBC file and can be adapted on the RTI CAN MultiMessage Blockset dialog pages.

Messages > 8 bytes (message packaging)

Standard CAN messages have a data field of variable length (0 ... 8 bytes). The J1939 protocol defines transport protocol functions which allow the transfer of up to 1785 bytes in one message.

A multipacket message contains up to 255 data fields, each of which has a length of 8 bytes. Each data field is addressed by the 1-byte sequence number, and contains 7 bytes of data. This yields a maximum of 1785 (= 255 · 7) bytes in one message.



The RTI CAN MultiMessage Blockset supports J1939 message packaging via BAM and RTS/CTS:

Broadcasting multipacket messages via BAM To broadcast a multipacket message, the sending network node first sends a *Broadcast Announce Message* (BAM). It contains the PGN and size of the multipacket message, and the number of packages. The BAM allows all receiving network nodes to prepare for the reception. The sender then starts the actual data transfer.

Peer-to-peer communication of multipacket messages via RTS/CTS To transfer a multipacket message to a specific destination in the network, the sending network node sends a *request to send* (RTS). The receiving network node responds with either a *clear to send* (CTS) message or a connection abort message if the connection cannot be established. When the sending network node receives the CTS message, it starts the actual data transfer.

By default, the number of CTS packets is set to 1. To allow peer-to-peer communication for multipacket J1939 messages longer than 8 bytes via RTS/CTS, you can change the default number of CTS packets. The RTI CAN MultiMessage Blockset provides the special MATLAB preference `set_j1939_cts_packet_number`. To change the default number of CTS packets, you must type the following command in the MATLAB Command Window:

```
rtimmsu_private('fcnlib', 'set_j1939_cts_packet_number', 'can',
<n>);
```

The argument `<n>` describes the number of CTS packets. The value must be in the range [1, 255].

Related topics

Basics

Lesson 13 (Advanced): Working with a J1939-Compliant DBC File (RTI CAN MultiMessage Blockset Tutorial 📖)

CAN Setup Blocks

Introduction

The **dSPACE_Blocksets/rticanmmlib** library provides CAN Setup blocks that let you specify folders for working with RTICANMM and set up the CAN bus.

Where to go from here

Information in this section

RTICANMM GeneralSetup

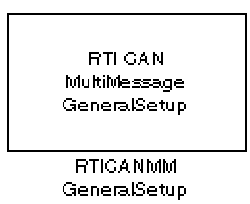
Where to go from here

Information in this section

Information in other sections

Block Description (RTICANMM GeneralSetup)

Block



Purpose

To specify general settings for using the RTI CAN MultiMessage Blockset.

Description

You have to specify the folder to which the RTI CAN MultiMessage Blockset will save all generated files (destination folder).

The RTI CAN MultiMessage Blockset supports CAN bus monitoring with the **Bus Navigator**.

- CAN bus monitoring always uses host service 31 for the following platforms:
  - DS1006-/DS1007-based systems with a DS2202, DS2210, DS2211, DS4302, and/or DS4505

  - MicroAutoBox II
  - MicroLabBox
- CAN bus monitoring always uses host service 63 for SCALEXIO systems.

> **Note**
>
> The settings you specify in the RTICANMM GeneralSetup block apply to all the RTICANMM blocks in a model. For this reason, there must always be one and only one RTICANMM GeneralSetup block in your model if you want to use any of the other RTICANMM blocks.
> Before you add the RTICANMM GeneralSetup block to the model, you must save the model.

Dialog menu

Settings global to the RTICANMM GeneralSetup block can be specified on the dialog menu. Refer to Options Menu (RTICANMM GeneralSetup) on page 39.

Dialog pages

The dialog settings can be specified on the dialog page. Refer to Main Page (RTICANMM GeneralSetup) on page 40.

Related topics

Basics

Lesson 1: Building Your First Model Using the RTI CAN MultiMessage Blockset (RTI CAN MultiMessage Blockset Tutorial 🕮)

References

Options Menu (RTICANMM GeneralSetup)

Purpose

To specify the folders on your file system required for working with the RTI CAN MultiMessage Blockset.

Menu commands

Create S-Function for All CAN Blocks Lets you create S-functions for all the RTICANMM blocks in the current model.

Create S-Function for Some CAN Blocks Lets you create S-functions for one or more RTICANMM blocks in the current model. You can select one or more blocks from the list containing all the blocks in the model. You can use the

Mapping block option to overwrite the options for a Mapping to RTICANMM Block. You can select one of the following options:

- **BlockSettings** Lets you use the options for a Mapping to RTICANMM Block specified on the Peripheral Options Page (RTICANMM MainBlock) on page 262.
- **Add** Lets you add a new Mapping to RTICANMM Block without overwriting existing Mapping to RTICANMM Blocks.
- **Overwrite** Lets you overwrite an existing Mapping to RTICANMM Block by a new one.

You can create S-functions for the selected blocks by clicking the **Create** button.

Check CAN Lets you check the correct use of the RTICANMM MainBlocks in your model without having to start a build process.

Set Compiler Options Lets you set the compiler options to the compiler default optimization **-O** (minimum compiler optimization). You should use these options to avoid long compilation times when working with extensive CAN communication.

> **Tip**
>
> If the compilation times are too long even though you selected the minimum compiler optimization, you can deactivate the optimization on the **RTI general build options page** in the Code Generation Dialog (Model Configuration Parameters Dialogs) (RTI and RTI-MP Implementation Reference 📖).

Clean up Destination Folder Lets you delete all generated files located in the destination folder the RTI CAN MultiMessage Blockset does not need any longer from your file system. MAT files containing the RTICANMM MainBlock configuration and custom files are not deleted.

Related topics

Basics

References

Main Page (RTICANMM GeneralSetup)

Purpose

To specify the folders on your file system required for working with the RTI CAN MultiMessage Blockset.

Dialog settings

Model root Displays the root folder of the current model. You cannot change the setting.

> **Note**
>
> All the paths you specify in the RTI CAN MultiMessage Blockset are relative to the model root folder. This also applies to the paths of the files generated by the RTI CAN MultiMessage Blockset, such as TRC files. For this reason, you should change the location of the model only if you also change the location of all other files generated by the RTI CAN MultiMessage Blockset.

Destination folder for generated files Lets you specify the folder to which the RTI CAN MultiMessage Blockset will save all the files it generates.

Use relative path Lets you set the folder relative to the model root folder or as an absolute path.

> **Tip**
>
> Using a relative path allows you to easily move the model and all the files belonging to it to another location on your file system without having to change any path settings.

Add paths to MATLAB search path Lets you add the destination folder to the MATLAB search path. You do not have to select this option if you add the destination folder to the MATLAB search yourself. To use Simulink® Coder™, the destination folder must be in the MATLAB search path.

Remove paths when model is closed Lets you remove the destination folder from the MATLAB search path whenever you close the current model.

> **Note**
>
> If you do not remove the destination folder, an error may occur when you open another model.

Add TRC exclusions to RTICANMM blocks Lets you exclude the RTICANMM blocks from the TRC file, because they are not important to it.

> **Note**
>
> You must not select this checkbox if you want to use mask and workspace parameters for parameter tuning.

Bus Navigator Monitoring Service Displays the host service used for CAN bus monitoring with the Bus Navigator. CAN bus monitoring with the Bus Navigator is always activated and always uses the displayed host service.

For more information on CAN bus monitoring with the **Bus Navigator** in ControlDesk, refer to ControlDesk Bus Navigator 📖.

Related topics

Basics

References

RTICANMM ControllerSetup

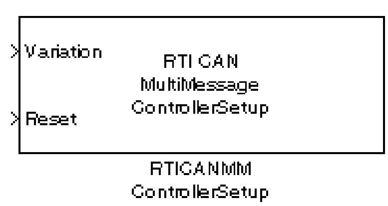
Where to go from here

Information in this section

Information in other sections

Block Description (RTICANMM ControllerSetup)

Block



Purpose

To specify basic settings of a CAN controller.

Description

To use the RTI CAN MultiMessage Blockset, a CAN controller of a dSPACE board must be initialized via the RTICANMM ControllerSetup block. To initialize the CAN controller, you have to provide information such as the baud rate, transceiver type, termination, and bit timing parameters.

Note

- Before you can use any of the other RTICANMM blocks, you have to:
  - Add an RTICANMM ControllerSetup block to your model
  - Open the block and specify its dialog settings
  - Confirm your settings by clicking **OK** in the block dialog

  As a result, an S-function is created for the RTICANMM ControllerSetup block. You can now specify CAN communication in the RTICANMM MainBlock.
- Each CAN controller to be used in the model must be specified by a separate RTICANMM ControllerSetup block.
- (Not relevant for SCALEXIO systems) If several CAN boards are connected to different PHS buses of a multiprocessor system, identical board numbers are assigned to the boards. Each CAN board requires its own RTICANMM ControllerSetup block. The controller names are used to distinguish between RTICANMM ControllerSetup blocks.
- The settings you specify in the RTICANMM GeneralSetup block apply to all the RTICANMM blocks in a model. For this reason, there must always be one RTICANMM GeneralSetup block in your model if you want to use any of the other RTICANMM blocks.

I/O characteristics

The table below describes the available block inports:

Simulink Inport	Range	Simulink Data Type	Description
Variation	0 … 20	Uint8	Specifies the variation for the CAN controller. You can use this inport to activate the RTICANMM MainBlock that corresponds to the specified CAN controller variation. The variation of an RTICANMM MainBlock can be specified on the General Settings Page (RTICANMM MainBlock) on page 100. If the input is `0`, the CAN controller is disabled. If the variation has been switched by the **Bus Navigator**, it is not possible to directly switch the variation from within the model. First, you must set the **Variation_Switch** variable back to `0` manually. The variable is in the **Variables** controlbar of ControlDesk below BusSystems/CAN/ControllerSetup_Blocks/<controller name>. For detailed information on variation switching by the **Bus Navigator**, refer to Lesson 10: Working with Variants of a CAN Controller (RTI CAN MultiMessage Blockset Tutorial) and Overview of the Bus Navigator (ControlDesk Bus Navigator).
Reset	0 … 1	Boolean	Resets the CAN controller on each occurrence of a signal edge. Do not leave the **Reset** input unconnected. You can connect it to a Simulink Constant block, for example.
Status	0 … 3	Uint8	(Available only if you select ISO11898-6 or ISO11898-5 as the transceiver type. Low power mode is available only with these transceiver types.)

Simulink Inport	Range	Simulink Data Type	Description
			Specifies the low power mode of the transceiver: ▪ "0" for the standby mode. Energy saving mode that allows you only to monitor the CAN bus. ▪ "1" for the sleep mode. Puts the transceiver into sleep mode. ▪ "2" for the power on / listen only mode (silent mode). The transmitter is switched off and the bus still receives messages. ▪ "3" for the normal mode. Allows normal bidirectional bus communication.
Autowakeup	0 … 1	Boolean	(Available only if you select ISO11898-6 or ISO11898-5 as the transceiver type) Enables the automatic wake-up feature of the transceiver. If the input is "1", the transceiver switches automatically from standby mode or sleep mode to normal mode if there is a message on the bus or the model wants to send a message. The automatic wake-up feature does not switch from silent mode to normal mode. The transceiver remains in silent mode until you select another mode that allows switching to normal mode.
Bus off	0 … 1	Uint8	(Available only if you select **Bus off** on the Options Page (RTICANMM ControllerSetup) on page 56) Enables the CAN controller: ▪ "0" for enabled CAN controller. ▪ "1" for disabled CAN controller. The CAN controller is disabled with every signal edge. It remains disabled until it is reset via the **Reset** inport.
Baudrate	10 … 1000		(Available only if you select **Baudrate input** on the Options Page (RTICANMM ControllerSetup) on page 56) Defines the baud rate to be used for CAN messages and for the arbitration phase of CAN FD messages on the CAN bus.
BaudrateFD	500 … 8000		(Available only if you select **Data phase baudrate input** on the Options Page (RTICANMM ControllerSetup) on page 56) Defines the baud rate to be used for the data phase of CAN FD messages on the CAN bus.

The table below describes the available block outports:

Simulink Outport	Range	Simulink Data Type	Description
Status	0 … 3	Uint8	(Available only if you select ISO11898-6 or ISO11898-5 as the transceiver type) Outputs the mode of the transceiver: ▪ "0" for the standby mode ▪ "1" for the sleep mode ▪ "2" for the power on / listen only mode (silent mode) ▪ "3" for the normal mode

Simulink Outport	Range	Simulink Data Type	Description
BusFailure	0 … 1	Boolean	(Available only if you select ISO11898-5 as the transceiver type) ▪ Outputs "1" if the transceiver is in normal mode and an error was detected on the CAN bus. ▪ Outputs "0" if the transceiver is in normal mode and no error was detected.
LocalFailure	0 … 1	Boolean	(Available only if you select ISO11898-5 or ISO11898-6 as the transceiver type) ▪ Outputs "1" if the transceiver is in power on / listen only mode and an error was detected on the bus. ▪ Outputs "0" if no error was detected. The transceiver provides an additional meaning of this error state if the transceiver is switched from sleep mode or standby mode to power on / listen only mode. This is not supported.
WakeupFlag	0 … 1	Boolean	(Available only if you select ISO11898-6 or ISO11898-5 as the transceiver type) Outputs "1" if the transceiver is in standby mode or sleep mode and a wake-up is detected. The wake-up flag can be set for one sampling step only.
Status	0 … 2	Uint8	(Available only if you select **CAN controllers status** on the Options Page (RTICANMM ControllerSetup) on page 56) Outputs status information of the CAN controller's Error Management Logic (EML): ▪ "0" (error active): The CAN controller is active. ▪ "1" (error warn): Before turning to the error passive state, the controller sets an error warn (EWRN) bit. ▪ "2" (bus off): The CAN controller disconnects itself from the bus. To recover, an external action is required (bus off recovery).
Stuff errors	0 … 2^{32}–1	Uint32	(Available only if you select **Number of stuff bit errors** on the Options Page (RTICANMM ControllerSetup) on page 56) Counts the number of stuff bits. The counter is incremented each time more than 5 consecutive equal bits occurred in a part of a received message where this is not allowed.
Format errors	0 … 2^{32}–1	Uint32	(Available only if you select **Number of form errors** on the Options Page (RTICANMM ControllerSetup) on page 56) Counts the number of format errors. The counter is incremented each time the format of a received message deviates from the fixed format.
Ack errors	0 … 2^{32}–1	Uint32	(Available only if you select **Number of acknowledge errors** on the Options Page (RTICANMM ControllerSetup) on page 56) Counts the number of acknowledge errors. The counter is incremented each time a message sent by the CAN controller is not acknowledged.
Bit 0 errors	0 … 2^{32}–1	Uint32	(Available only if you select **Number of bit 0 errors** on the Options Page (RTICANMM ControllerSetup) on page 56) Counts the number of bit 0 errors. Each time the CAN controller tries to send a dominant bus level but a recessive level is detected instead, the counter is incremented. During bus off recovery, the counter is

Simulink Outport	Range	Simulink Data Type	Description
			incremented each time a sequence of 11 consecutive recessive bits is detected.
Bit 1 errors	0 … 2^{32}–1	Uint32	(Available only if you select **Number of bit 1 errors** on the Options Page (RTICANMM ControllerSetup) on page 56) Counts the number of bit 1 errors. The counter is incremented each time the CAN controller tries to send a recessive bus level but a dominant level is detected instead.
CRC errors	0 … 2^{32}–1	Uint32	(Available only if you select **Number of CRC errors** on the Options Page (RTICANMM ControllerSetup) on page 56) Counts the number of checksum errors for received messages.
RX lost	0 … 2^{32}–1	Uint32	(Available only if you select **Number of lost RX messages** on the Options Page (RTICANMM ControllerSetup) on page 56) Provides the number of RX messages lost by the CAN controller: If a message cannot be stored in the buffer of the CAN controller, the message is lost and an *RX lost error* is detected.
RX ok	0 … 2^{32}–1	Uint32	(Available only if you select **Number of successfully received RX messages** on the Options Page (RTICANMM ControllerSetup) on page 56) Counts the number of successfully received RX messages.
TX ok	0 … 2^{32}–1	Uint32	(Available only if you select **Number of successfully sent TX messages** on the Options Page (RTICANMM ControllerSetup) on page 56) Counts the number of successfully sent TX messages.

Dialog pages

The dialog settings can be specified on the following pages and dialogs:

- Setup Page (RTICANMM ControllerSetup) on page 48
- Options Page (RTICANMM ControllerSetup) on page 56
- Partial Networking Page (RTICANMM ControllerSetup) on page 60
- Advanced Configuration Dialog (RTICANMM ControllerSetup) on page 63

Related topics

Basics

Lesson 1: Building Your First Model Using the RTI CAN MultiMessage Blockset (RTI CAN MultiMessage Blockset Tutorial 📖)
Lesson 10: Working with Variants of a CAN Controller (RTI CAN MultiMessage Blockset Tutorial 📖)

Setup Page (RTICANMM ControllerSetup)

Purpose

To set up a CAN controller.

> **Note**
>
> When you switch to or from a MicroAutoBox II or MicroLabBox, you have to recreate the ControllerSetup block.

Dialog settings

Controller name Lets you specify the controller name. Controller names are used to identify the different RTICANMM ControllerSetup blocks in an RTI model. Each controller name can therefore be used only once within an RTI model. This also applies to RTI-MP (multiprocessor) models. For SCALEXIO systems, the controller name must be unique within a whole multicore application.

The RTI CAN MultiMessage Blockset automatically creates a default controller name. It is recommended to change the default name to guarantee that the controller name is unique. After that you have to recreate the ControllerSetup block.

> **Note**
>
> The length of the controller name is limited to 16 characters.

Board type (Not available for SCALEXIO systems) Lets you select the type of the dSPACE board you use for connection to the CAN bus. For a MicroAutoBox II and MicroLabBox, select **CAN Type 1**. If you want to use the CAN FD protocol with a MicroAutoBox II, select **IP Type 1**.

Board number (Not available for SCALEXIO systems) Lets you select the number of the dSPACE board you use for connection to the CAN bus. The valid numbers are:

Hardware	Board Number
MicroAutoBox II	1 … 16
DS2202, DS2210, and DS2211	1 … 16
DS4302	1 … 16
DS4505	1 … 16
MicroLabBox	1

Module number (Available only if DS4505 is selected as the board type) Lets you select the number of the CAN FD interface module (1 ... 4) you use for connection to the CAN bus.

Module type (Available only if DS4505 or IP Type 1 is selected as the board type) Lets you select the type of the interface module used for CAN FD support.

Controller number (Not available for SCALEXIO systems) Lets you select the number of the CAN controller. The valid numbers are:

Board	Controller Number
MicroAutoBox II	1 … 2 Mapping between board/controller numbers in RTI and interface connector pin numbers as described in the MicroAutoBox II Hardware Reference 📖 document: ▪ The combination of board 1 and controller 1 is named *CAN 1* in the pin description. ▪ The combination of board 1 and controller 2 is named *CAN 2*. ▪ The combination of board 2 and controller 1 is named *CAN 3*. ▪ The combination of board 2 and controller 2 is named *CAN 4*. ▪ The combination of board 3 and controller 1 is named *CAN 5*. ▪ The combination of board 3 and controller 2 is named *CAN 6*.
DS2202, DS2210, and DS2211	1 … 2
DS4302	1 … 4
DS4505	1 … 2
MicroLabBox	1 … 2 Mapping between board/controller numbers in RTI and interface connector pin numbers as described in the MicroLabBox Hardware Installation and Configuration 📖 document: ▪ The combination of board 1 and controller 1 is named *CAN 1* in the pin description. ▪ The combination of board 1 and controller 2 is named *CAN 2*.

For the mapping of CAN controller numbers to CAN signals and pins of dSPACE hardware, refer to the topics in the table below:

dSPACE Hardware		Topic with Mapping Information
DS2202		Signal Mapping to I/O Pins (PHS Bus System Hardware Reference 📖)
DS2210		Signal Mapping to I/O Pins (PHS Bus System Hardware Reference 📖)
DS2211		Signal Mapping to I/O Pins (PHS Bus System Hardware Reference 📖)
DS4302		Signal Mapping to I/O Pins (PHS Bus System Hardware Reference 📖)
DS4505		Signal Mapping to I/O Pins (DS4505/DS4342) (PHS Bus System Hardware Reference 📖)
MicroAutoBox II	1401/1507	Signal Mapping to I/O Pins (1401/1507) (refer to Interfaces (MicroAutoBox II Hardware Reference 📖))
	1401/1511	Signal Mapping to I/O Pins (1401/1511) (refer to Interfaces (MicroAutoBox II Hardware Reference 📖))
	1401/1511/1514	Signal Mapping to I/O Pins (1401/1511/1514) (refer to Interfaces (MicroAutoBox II Hardware Reference 📖))
	1401/1513	Signal Mapping to I/O Pins (1401/1513) (refer to Interfaces (MicroAutoBox II Hardware Reference 📖))
	1401/1513/1514	Signal Mapping to I/O Pins (1401/1513/1514) (refer to Interfaces (MicroAutoBox II Hardware Reference 📖))
MicroLabBox		Signal Mapping to I/O Pins (refer to CAN Connector (Sub-D) (MicroLabBox Hardware Installation and Configuration 📖))

dSPACE Hardware		Topic with Mapping Information
SCALEXIO	DS2671 Bus Board	Signal Mapping of the DS2671 Bus Board (SCALEXIO Hardware Installation and Configuration 📖)
	DS2672 Bus Module	Signal Mapping of the DS2672 Bus Module (SCALEXIO Hardware Installation and Configuration 📖)
	DS6301 CAN/LIN Board	Signal Mapping of the DS6301 CAN/LIN Board (SCALEXIO Hardware Installation and Configuration 📖)
	DS6341 CAN Board	Signal Mapping of the DS6341 CAN Board (SCALEXIO Hardware Installation and Configuration 📖)
	DS6342 CAN Board	Signal Mapping of the DS6342 CAN Board (SCALEXIO Hardware Installation and Configuration 📖)

Identifier format (Not available for SCALEXIO systems and board types that support CAN FD) Lets you select one of the following message identifier formats:

- STD This is the standard format, containing an 11-bit identifier. It corresponds to the CAN 2.0 A specification.
- EXT This is the extended format, containing a 29-bit identifier. It corresponds to the CAN 2.0 B specification.

Messages of both identifier formats can be transmitted and received. However, messages with the selected identifier format cause less run time on the slave processor and therefore prevent the slave system from losing receive messages. The selected message identifier format applies to all RTICANMM MainBlocks that reference the RTICANMM ControllerSetup block (see General Settings Page (RTICANMM MainBlock) on page 100).

Note

If you want to use J1939 support, you must select the EXT format.

CAN FD support (Available only if DS4505 or IP Type 1 is selected as the board type, and for SCALEXIO with ISO 11898-2 selected as the transceiver type) Lets you activate the CAN FD mode for the CAN transceiver and specify the protocol to be used with CAN FD messages. You can work with the non-ISO CAN FD protocol (Bosch's original CAN FD protocol) or the ISO CAN FD protocol (according to the ISO 11898-1:2015 standard). If you select one of the CAN FD protocols, you can monitor CAN FD messages in addition to classic CAN messages on the CAN bus. To disable the CAN FD mode for the CAN transceiver, select No CAN FD from the list.

After you selected a CAN FD protocol, you can specify further settings for CAN FD messages (e.g., an optional higher baud rate for the CAN FD data phase, see Baud rate FD).

Note

To receive and transmit CAN FD messages on the bus, you must also enable CAN FD support in the MainBlock. Refer to General Settings Page (RTICANMM MainBlock) on page 100.

Transceiver type Lets you select the transceiver type of your CAN bus.

<table>
<tr><th>Hardware</th><th>Setting</th></tr>
<tr><td>MicroAutoBox II</td><td>If CAN Type 1 is selected as the board type, the following transceiver types are supported:
ISO11898-2 (formerly ISO11898)
ISO11898-6[1)]
If IP Type 1 is selected as the board type, the following transceiver types are supported:
ISO11898-2
ISO11898-6[2)]</td></tr>
<tr><td>DS2202, DS2210, and DS2211</td><td>The DS2202, DS2210, and DS2211 support only the ISO11898-2 (formerly ISO11898) transceiver. You cannot change the setting.</td></tr>
<tr><td>DS4302</td><td>The following transceiver types are supported:
ISO11898-2 (formerly ISO11898)
RS485
ISO11898-3 (formerly C252)
ISO11898-5 (formerly TJA1041)
If none of the above transceivers matches your CAN application, select PIGGY-BACK to connect to an individual transceiver.</td></tr>
<tr><td>DS4505</td><td>The following transceiver types are supported:
ISO11898-2
ISO11898-6[2)]</td></tr>
<tr><td>MicroLabBox</td><td>The following transceiver types are supported:
ISO11898-2
ISO11898-6</td></tr>
<tr><td>DS2671[3)]</td><td>The following transceiver types are supported:
ISO11898-2
ISO11898-3
Piggyback PlugOn module</td></tr>
<tr><td>DS2672[3)]</td><td>The following transceiver types are supported:
ISO11898-2
ISO11898-3</td></tr>
<tr><td>DS6301[3)]</td><td>The following transceiver types are supported:
ISO11898-2
ISO11898-3
Piggyback PlugOn module</td></tr>
<tr><td>DS6341[3)]</td><td>The following transceiver types are supported:
ISO11898-2
ISO11898-3
Piggyback PlugOn module</td></tr>
</table>

Hardware	Setting
DS6342[3)]	The following transceiver types are supported: ▪ ISO11898-2 ▪ ISO11898-3 ▪ Piggyback PlugOn module

[1)] Available only for a MicroAutoBox II with DS1513 I/O Board.
[2)] If ISO11898-6 is selected but partial networking is not enabled, the ISO11898-5 transceiver type is used instead (even though ISO11898-6 is still displayed).
[3)] In a SCALEXIO system.

Bus configuration register (Not available for SCALEXIO systems) Lets you configure the bus configuration register in hexadecimal format.

Hardware	Setting
MicroAutoBox II	The MicroAutoBox II does not support a transceiver type other than ISO11898-2 or ISO11898-6. You cannot change the setting.
DS2202, DS2210, and DS2211	The DS2202, DS2210, and DS2211 do not support a transceiver type other than ISO11898-2. You cannot change the setting.
DS4302	Enabled only if you select “PIGGY-BACK” or “ISO11898-5” as the transceiver type. Via the bus configuration register, you can configure the CAN controller for the physical/electrical characteristics of the CAN bus according to the piggyback or the ISO11898-5 transceiver.
DS4505	The DS4505 does not support a transceiver type other than ISO11898-2 or ISO11898-6. You cannot change the setting.
MicroLabBox	The MicroLabBox does not support a transceiver type other than ISO11898-2 or ISO11898-6. You cannot change the setting.

Termination resistance Lets you specify the resistance of the CAN bus terminator.

Hardware	Resistance
MicroAutoBox II	▪ MicroAutoBox II 1401/1513 variant: Select 120 Ω or off. ▪ All other MicroAutoBox II variants: There is no termination resistance installed on the MicroAutoBox II. This means that there is no connection between CAN-Low and CAN-High. This setting therefore has no effect on the hardware.
DS2202, DS2210, and DS2211	Select 120 Ω or off.
DS4302	Select 120 Ω or off. If you select ISO11898-3 as the transceiver type, select 1.6 kΩ or 10 kΩ.
DS4505	Select 120 Ω or off. (The termination resistor is installed on the DS4342.)
MicroLabBox	Select 120 Ω or off.
DS2671	▪ ISO11898-2 High-Speed CAN: Select On or Off. "Off" means that there is no 120-Ω termination resistor between CAN-Low and CAN-High. ▪ ISO11898-3 Fault-Tolerant CAN: Select High or Low. ▪ Piggyback PlugOn module: Select On or Off. "Off" means that there is no 120-Ω termination resistor between CAN-Low and CAN-High.
DS2672	▪ ISO11898-2 High-Speed CAN: Select On or Off. "Off" means that there is no 120-Ω termination resistor between CAN-Low and CAN-High. ▪ ISO11898-3 Fault-Tolerant CAN: Select High or Low.

Hardware	Resistance
DS6301	▪ ISO11898-2 High-Speed CAN: Select On or Off. "Off" means that there is no 120-Ω termination resistor between CAN-Low and CAN-High. ▪ ISO11898-3 Fault-Tolerant CAN: Select High or Low. ▪ Piggyback PlugOn module: Select On or Off. "Off" means that there is no 120-Ω termination resistor between CAN-Low and CAN-High.
DS6341	▪ ISO11898-2 High-Speed CAN: Select On or Off. "Off" means that there is no 120-Ω termination resistor between CAN-Low and CAN-High. ▪ ISO11898-3 Fault-Tolerant CAN: Select High or Low. ▪ Piggyback PlugOn module: Select On or Off. "Off" means that there is no 120-Ω termination resistor between CAN-Low and CAN-High.
DS6342	▪ ISO11898-2 High-Speed CAN: Select On or Off. "Off" means that there is no 120-Ω termination resistor between CAN-Low and CAN-High. ▪ ISO11898-3 Fault-Tolerant CAN: Select High or Low. ▪ Piggyback PlugOn module: Select On or Off. "Off" means that there is no 120-Ω termination resistor between CAN-Low and CAN-High.

Note

If the board is installed at one end of a CAN network, you should select 120 Ω as the termination resistance. In this case, the CAN bus is physically terminated with the 120 Ω-termination resistor installed on the board.

Without termination, signal changes may cause communication failures due to reflections on the CAN network.

Standard configuration (Not available for SCALEXIO systems) Lets you set up the CAN controller by specifying the baud rate.

When you select the standard configuration, an algorithm automatically determines an appropriate configuration for the bit timing registers of the CAN controller.

Tip

No detailed hardware information is required for the standard configuration.

Baud rate Lets you enter the baud rate in kB/s. This value is the standard data rate used for classic CAN messages and for the arbitration phase of CAN FD messages. The valid minimum and maximum baud rates are:

Hardware	Transceiver Type	Minimum Rate (in kB/s)	Maximum Rate (in kB/s)
MicroAutoBox II	ISO11898-2	20	1000
	ISO11898-6	20	1000
DS2202	ISO11898-2	10	1000
DS2210	ISO11898-2	10	1000
DS2211	ISO11898-2	10	1000

Hardware	Transceiver Type	Minimum Rate (in kB/s)	Maximum Rate (in kB/s)
DS4302	ISO11898-2	10	1000
	RS485	10	500
	ISO11898-3	10	125
	PIGGY-BACK	10	1000
	ISO11898-5	10	1000
DS4505/DS4342	ISO11898-2	10	1000
	ISO11898-6	10	1000
MicroLabBox	ISO11898-2	20	1000
	ISO11898-6	1	1000
DS2671	ISO11898-2 High-Speed CAN	40	1000
	ISO11898-3 Fault-Tolerant CAN	40	125
	Piggyback PlugOn module	5	1000
DS2672	ISO11898-2 High-Speed CAN	40	1000
	ISO11898-3 Fault-Tolerant CAN	40	125
DS6301	ISO11898-2 High-Speed CAN	40	1000
	ISO11898-3 Fault-Tolerant CAN	40	125
	Piggyback PlugOn module	5	1000
DS6341	ISO11898-2 High-Speed CAN	40	1000
	ISO11898-3 Fault-Tolerant CAN	40	125
	Piggyback PlugOn module	5	1000
DS6342	ISO11898-2 High-Speed CAN	40	1000
	ISO11898-3 Fault-Tolerant CAN	40	125
	Piggyback PlugOn module	5	1000

Sample point (Available only for dSPACE hardware that supports CAN FD communication, and only if a CAN FD protocol is selected in the **CAN FD support** field) Lets you specify the sample point for the arbitration phase of CAN FD message transmission as a percentage of the CAN bit period. You can enter a value in the range 50 ... 91. However, not each value of this value range can be used with a CAN FD controller. Depending on the baud rate and the number of time quantums available, only certain sample point values are possible. If you specified an unsuitable sample point value, the real-time application will set the next possible sample point value.

The log file provides information on the actual sample point value (this does not apply to SCALEXIO).

Note

All members in a CAN FD network must have the same setting for the arbitration phase sample point.

Baud rate FD (Available only for dSPACE hardware that supports CAN FD communication, and only if a CAN FD protocol is selected in the CAN FD support field) Lets you specify the baud rate in kB/s used for the data phase of CAN FD messages on the CAN bus. The valid minimum and maximum baud rates are:

Hardware	Minimum Rate (in kB/s)	Maximum Rate (in kB/s)
DS4505/DS4342	500	8000
MicroAutoBox II/ IP Type 1	500	8000
DS2671	125	8000
DS2672	125	8000
DS6301	125	8000
DS6341	125	8000
DS6342	125	8000

Sample point FD (Available only for dSPACE hardware that supports CAN FD communication, and only if a CAN FD protocol is selected in the CAN FD support field) Lets you specify the sample point for the data phase of CAN FD message transmission as a percentage of the CAN bit period. You can enter a value in the range 50 ... 91. However, not each value of this value range can be used with a CAN FD controller. Depending on the baud rate and the number of time quantums available, only certain sample point values are possible. If you specified an unsuitable sample point value, the real-time application will set the next possible sample point value.

The log file provides information on the actual sample point value (this does not apply to SCALEXIO).

Note

All members in a CAN FD network must have the same setting for the data phase sample point.

Advanced configuration (Not available for DS4505, MicroAutoBox II with IP Type 1 selected as the board type, MicroLabBox, and SCALEXIO systems) Disables the baud rate setting of the standard configuration, and lets you specify advanced configuration settings.

Activating the advanced configuration automatically disables the Baudrate input option on the Options Page (RTICANMM ControllerSetup), that is, no

Baudrate inport is added to the block. Thus, you cannot change the baud rate during run time as long as the advanced configuration is active.

Note

Detailed knowledge of the CAN controller hardware and the CAN bus hardware is required for advanced configuration.

Edit (Not available for DS4505, MicroAutoBox II with IP Type 1 selected as the board type, MicroLabBox, and SCALEXIO systems) Lets you specify the advanced configuration in the Advanced Configuration Dialog (RTICANMM ControllerSetup) on page 63.

Related topics

Basics

References

Options Page (RTICANMM ControllerSetup)

Purpose

To get status information on the CAN controller and the CAN bus, and to remove a CAN controller from the CAN bus.

Bus statistics period frame

Lets you enter a sample time in seconds. The sample time determines how often the functions of this page are executed. You can specify a sample time larger than the model's sample time to minimize the execution time. Enter –1 to keep the model's base sample time (inherited).

Note

If the model´s sample time is < 10 ms, you should set the sample time to the hundredfold value of the model's sample time. This avoids communication overload and loss of data.

Tip

For most use cases, you can use the default sample time of 1 second.

Read CAN controller status information frame

Lets you enable various status information options. An outport is added to the RTICANMM GeneralSetup block for each selected option.

> **Note**
>
> For optimum performance, you should specify as few status information options as possible.

CAN controller status Lets you add a Status outport to the block to output status information of the CAN controller's Error Management Logic (EML). The following values can be displayed:

Outport Deliver	Error State	Description
0	Error active	The CAN controller is active.
1	Error warn	Before turning to the error passive state, the controller sets an error warn (EWRN) bit.
2	Bus off	The CAN controller disconnects itself from the bus. To recover, an external action is required (bus off recovery).

Number of stuff bit errors Lets you add a Stuff errors outport to the block which counts the number of stuff bits. Each time more than 5 consecutive equal bits occurred in a part of a received message where this is not allowed, the counter is incremented. Possible counter values are 0 … 2^{32}–1.

Number of form errors Lets you add a Format errors outport to the block which counts the number of format errors. Each time the format of a received message deviates from the fixed format, the counter is incremented. Possible counter values are 0 … 2^{32}–1.

Number of acknowledge errors Lets you add an Ack errors outport to the block which counts the number of acknowledge errors. Each time a message sent by the CAN controller is not acknowledged, the counter is incremented. Possible counter values are 0 … 2^{32}–1.

Number of successfully received RX messages Lets you add an RX ok outport to the block which counts the number of successfully received RX messages. Possible counter values are 0 … 2^{32}–1.

Number of successfully sent TX messages Lets you add a TX ok outport to the block which counts the number of successfully sent TX messages. Possible counter values are 0 … 2^{32}–1.

Number of bit 0 errors Lets you add a Bit 0 errors outport to the block which counts the number of bit 0 errors. Each time the CAN controller tries to send a dominant bus level but a recessive level is detected instead, the counter is incremented. During bus off recovery, the counter is incremented each time a sequence of 11 consecutive recessive bits is detected. This enables the controller to monitor the bus off recovery sequence, which indicates that the bus is not permanently interrupted. Possible counter values are 0 … 2^{32}–1.

Number of bit 1 errors Lets you add a Bit 1 errors outport to the block which counts the number of bit 1 errors. Each time the CAN controller tries to send a recessive bus level but a dominant level is detected instead, the counter is incremented. Possible counter values are 0 … 2^{32}–1.

Number of CRC errors Lets you add a CRC errors outport to the block which counts the number of checksum errors of a received message. Possible counter values are 0 … 2^{32}–1.

Number of lost RX messages Lets you add an RX lost outport to the block which provides the number of RX messages lost by the CAN controller. If a message cannot be stored in the buffer of the CAN controller, the message is lost and an RX lost error is detected. Possible counter values are 0 … 2^{32}–1.

Select all Lets you select all the status information options of this frame.

Select none Lets you deselect all the status information options of this frame.

Invert Lets you invert the selection of the status information options of this frame.

Supported features frame

Bus off Lets you add a Bus off inport to the block. You can enable or disable the CAN controller via this inport. If the input is "0", the CAN controller is enabled. If the input is "1", the CAN controller is disabled at every signal edge. It remains disabled until it is reset via the Reset inport.

Baudrate input Lets you add a Baudrate inport to the block. You can set the baud rate to be used for CAN messages and for the arbitration phase of CAN FD messages for the CAN controller via this inport.

Note

You can add a Baudrate inport to the block only if the standard configuration is active. As long as Advanced configuration is selected on the Setup Page (RTICANMM ControllerSetup), the RTI CAN MultiMessage Blockset prevents you from adding a Baudrate inport to the block.

Data phase baudrate input (Available only for platforms that support CAN FD, and only if a CAN FD protocol is selected on the Setup Page (RTICANMM ControllerSetup)) Lets you add a Baudrate FD inport to the block. You can set the baud rate for the CAN FD data phase (for transmitting the payload of a CAN FD message) for the CAN controller via this inport.

Error frame generation (Available only for dSPACE platforms if the DS4342 is selected as the module type, and for SCALEXIO systems) Lets you enable the generation of error messages for the CAN controller. If enabled, each time that a message that matches a specified mask is received on the CAN bus, the CAN controller replies with an error frame.

If error frame generation is enabled, you can activate and configure up to 16 automatic error frame generators for the CAN controller during run time in ControlDesk via entries in the generated TRC file. The following TRC variables are available:

Variable	Description
`<controller name>/EFG_Enable_<1 ... 16>`	To enable the associated error frame generator.
`<controller name>/EFG_IDFormat_<1 ... 16>`	To specify the identifier format of the CAN message that triggers the error frame generator. The following values are possible: ▪ 1: Classic CAN message with standard format (STD) ▪ 2: Classic CAN message with extended format (EXT) ▪ 3: Classic CAN message with either format (STD or EXT) ▪ 4: CAN FD message with standard format (FD STD) ▪ 5: CAN FD message with extended format (FD EXT) ▪ 6: CAN FD message with either format (FD STD or FD EXT) ▪ 7: Any CAN message format
`<controller name>/EFG_IdMask_<1 ... 16>`	To specify the CAN controller identifier mask. For each bit position, you can specify one of the following values: ▪ 0: 'don't care' ▪ 1: 'must match'
`<controller name>/EFG_Id_<1 ... 16>`	To specify a concrete CAN ID that must be received for triggering the error frame generator.
`<controller name>/EFG_DLCMask_<1 ... 16>`	To specify the data length mask.
`<controller name>/EFG_DLC_<1 ... 16>`	To specify how many data bytes are expected in the data length code (DLC) of the message.
`<controller name>/EFG_DataMask_<1 ... 16>_<1 ... 8>`	To specify the data mask of the error frame generator. The data mask is used for masking the relevant data bytes to determine valid messages. When a message that matches this data mask is received, an error frame is transmitted if an appropriate error frame generator is active. For each bit position, you can specify one of the following values: ▪ 0: 'don't care' ▪ 1: 'must match'
`<controller name>/EFG_Data_<1 ... 16>_<1 ... 8>`	To specify concrete CAN data bytes that must be received to trigger the error frame generator.
`<controller name>/EFGs_Kickout_<1 ... 16>`	To transmit an error frame via kickout, i.e., regardless of the bus state.
`<controller name>/EFG_Number_<1 ... 16>`	To specify the number of error frames to be sent. The value '-1' means that the number of transmissions is not limited.
`<controller name>/EFGs_State_<1 ... 16>`	To get state information on the error frame generator. The following values are possible: ▪ 1: Currently, there is no error frame to be sent. ▪ 2: At least one error frame must still be sent. ▪ 3: All requested error frames were transmitted.

You will find the variables for error frame generation in the Variables controlbar below `BusSystems/CAN/<controller name>/ErrorFrameGeneration`.

Calculate busload (Available only for SCALEXIO systems) Lets you add a Busload outport to the block. You can measure the busload on this channel via this outport.

Operation pending outport (Available only for SCALEXIO systems) Lets you add an Operation pending outport to the block. This outport indicates whether a channel modification is pending. If the output is "1", a channel modification is pending and message transmission is not possible.

Related topics

Basics

References

Partial Networking Page (RTICANMM ControllerSetup)

Purpose

To enable partial networking, and to define messages for transceiver wake-up.

Supported dSPACE platforms

Partial networking is possible for the following dSPACE real-time hardware:

- MicroAutoBox II equipped with the DS1513 I/O Board
- MicroLabBox
- dSPACE hardware that supports working with CAN FD messages and that is equipped with DS4342 CAN FD Interface Modules, such as:
  - PHS-bus-based systems (DS1006 or DS1007 modular systems) with DS4505 Interface Board
  - MicroAutoBox II variants with DS1507
  - MicroAutoBox II variants with DS1514

Basics on CAN partial networking

Principle of partial networking With CAN partial networking, selected ECUs in a network can be set to sleep mode or shut down if they do not have to run continuously. Wake-up messages then activate specific ECUs as and when required, and for as long as required.

Specifying wake-up messages The RTI CAN MultiMessage Blockset lets you specify the CAN partial networking wake-up messages by filtering message IDs and message data:

- Filtering message IDs: You can define a message filter to select the messages to use as wake-up messages. The filter uses a bitmask which represents the message. A message passes the filter and is used as wake-up message only if it matches the bitmask.
- Filtering message data: You can mask the data bytes of incoming wake-up messages to determine whether they are valid wake-up messages.

Switching the CAN transceiver to sleep mode The CAN transceiver of the dSPACE real-time hardware is switched to sleep mode via the real-time application.

Tip

(MicroAutoBox II only) You can stop and power down the MicroAutoBox II with the DS1401_POWER_DOWN block from the DS1401 MicroAutoBox Base Board II library. To set the MicroAutoBox II to sleep mode, KL15 (REMOTE) must be disconnected from the power supply. Before the MicroAutoBox II is put to sleep mode via the DS1401_POWER_DOWN block, the CAN transceiver should be put to sleep mode. For more information, refer to DS1401_POWER_DOWN (MicroAutoBox II RTI Reference 🕮).

Waking up dSPACE real-time hardware You can use partial networking messages to wake up dSPACE real-time hardware after its CAN transceiver is switched to sleep mode.

The transceiver configuration is kept as long as the VBAT voltage is connected to the transceiver.

(Relevant for MicroAutoBox II only) If the MicroAutoBox II was powered down via the DS1401_POWER_DOWN block and its CAN transceiver is woken up via a partial networking message, the MicroAutoBox II behaves as if it was powered up manually. Depending on where the real-time application is loaded (flash memory or RAM), the MicroAutoBox II starts the application or waits for further input.

(Relevant for all platforms except for the MicroAutoBox II) Unlike the MicroAutoBox II, the dSPACE real-time hardware cannot be stopped and then woken up via partial networking messages. However, the CAN transceiver of the dSPACE real-time hardware can be set to sleep mode and then woken up via partial networking messages later on.

TRC file and ControlDesk You can enable/disable and configure CAN partial networking in ControlDesk via entries in the generated TRC file. When you open the `<model>.sdf` file in ControlDesk, you will find the variables for partial networking in the Variables controlbar below `BusSystems/CAN/<controller name>/PartialNetworking`.

CAN controller partial networking frame

Enable Lets you enable or disable CAN partial networking for the transceiver.

- Partial networking is possible only if the correct ISO11898-6 transceiver type is selected.
- Partial networking is supported only for the following baud rates:
  - 125 kbit/s
  - 250 kbit/s

- 500 kbit/s
- 1000 kbit/s

Other baud rates can be used for normal CAN operation, but detecting wake-up messages for partial networking is supported only for the baud rates listed above.

Refer to Setup Page (RTICANMM ControllerSetup) on page 48.

CAN controller wake-up mask settings frame	**Identifier format** (Available only if partial networking is enabled) Lets you select the STD or EXT format for the identifier mask used for CAN partial networking. **Identifier mask** (Available only if partial networking is enabled) Lets you specify the bitmask that you want to use as the wake-up message filter. Each bit position can be assigned 0 (must be matched), 1 (must be matched), or X (don't care). A message is used as a partial networking wake-up message only if it matches the bitmask. You can input bitmasks with more than 11 bits even if you selected the standard identifier format for filtering wake-up messages. RTICANMM then uses the bits 10 ... 0 only, and further bits are ignored. If you input bitmasks with fewer bits than required by the selected identifier format, the required bits are set to X. In the TRC file, the identifier mask is represented by the two TRC variables `WakeUp_ID_Code` and `WakeUp_ID_Mask`, which are calculated from the entered bitmask. Example: If you enter the bitmask XXX10100000, the corresponding TRC variables are `WakeUp_ID_Code` = 11110100000 and `WakeUp_ID_Mask` = 00011111111.
Wake-up mask extensions frame	**Data mask used** (Available only if partial networking is enabled) Lets you enable a data mask for filtering valid wake-up messages. A data mask makes it possible to wake up a group (or several groups) of nodes simultaneously via a single wake-up message (see example below). **Data length** (Available only if **Data mask used** is selected) Lets you select how many data bytes are expected in the data length code (DLC) of a wake-up message. Messages with a different data length are ignored. **Data mask** (Available only if **Data mask used** is selected) Lets you specify the data mask value in hexadecimal notation. The data mask is used for masking the relevant data bytes (as specified in the **Data length** field) of the wake-up messages to determine valid wake-up messages. A logical AND operation is used between the data mask and the message. Example: You are working with CAN wake-up messages with a data length of 1 byte (**Data length** = 1). Each wake-up message can wake up several groups of network nodes simultaneously. A bit value of '1' at a certain bit position wakes up the related node group. For example, a wake-up message with the data field '11100010' wakes up group1, group2, group3, and group7. The specified **Data mask** is '0xA8' (which corresponds to the binary value '10101000').

When the wake-up message with data field '11100010' is received, it is evaluated against the specified data mask. Since there are two matching bits (bit 1 and bit 3), the wake-up message is valid and is used to wake up the node.

The table below illustrates this example.

	Group 1	Group 2	Group 3	Group 4	Group 5	Group 6	Group 7	Group 8
Mask	1	0	1	0	1	0	0	0
Received message	1	1	1	0	0	0	1	0
Wake-up result	✓ [1)]	-	✓ [1)]	-	-	-	-	-

[1)] Nodes of this group are woken up.

Related topics

Basics

Block Description (RTICANMM ControllerSetup).. 43

References

Advanced Configuration Dialog (RTICANMM ControllerSetup).. 63
Options Page (RTICANMM ControllerSetup)... 56
Setup Page (RTICANMM ControllerSetup)... 48

Advanced Configuration Dialog (RTICANMM ControllerSetup)

Access

On the **Setup Page (RTICANMM ControllerSetup)** on page 48, select Advanced configuration and click Edit.

Note

The advanced configuration is not available for the MicroLabBox and SCALEXIO systems.

Purpose

To specify the baud rate and either show all valid controller settings or enter the BTR (bit timing register) values.

Dialog settings

The resulting controller settings are displayed in the Controller settings selection frame.

Baud rate Lets you enter the baud rate in kbit/s. The valid minimum and maximum baud rates are:

Hardware	Transceiver Type	Minimum Rate (in kbit/s)	Maximum Rate (in kbit/s)
MicroAutoBox II	ISO11898-2	10	1000
	ISO11898-6	10	1000
DS2202	ISO11898-2	10	1000
DS2210	ISO11898-2	10	1000
DS2211	ISO11898-2	10	1000
DS4302	ISO11898-2	10	1000
	RS485	10	500
	ISO11898-3	10	125
	PIGGY-BACK	10	1000
	ISO11898-5	10	1000

External clock (Available only for MicroAutoBox II and DS2211) Lets you select the external clock rate used for calculating the BTR registers. The available frequencies are 24 MHz, 36 MHz and 64 MHz (MicroAutoBox II only). The bit timing register values for the selected frequency and the alternative frequency are calculated automatically. The selected frequency has no effect on the connected dSPACE board and is only used for calculation.

Show all valid settings frame

Disables the settings from the Enter BTR values frame. The CAN controller frequency is calculated as high as possible according to the specified baud rate and the sample mode. Bit timing values are also calculated. The Controller settings selection frame lists all the combinations of BTRO, BTR1, SP, SJW, TSEG1, TSEG2, BRP and SMPL values that are possible for the calculated frequency. Select one combination from the list. BTR values for the alternative CAN controller frequency are calculated if you use a DS2211 or a MicroAutoBox II.

1 sample per bit Lets the CAN controller sample each bit once to decide whether it is recessive or dominant.

3 samples per bit Lets the CAN controller sample each bit three times and use the majority to decide if a bit is recessive or dominant.

Note

The 3 samples per bit mode is supported by the DS4302 only.

Enter BTR values frame

Disables the settings from the Show all valid settings frame, sets the Baud rate edit field to read-only, and enables the BTR0 and BTR1 edit fields. A baud rate and bit timing values (for SP, SJW, TSEG1, TSEG2, BRP and SMPL) are calculated according to the specified BTR0 and BTR1 values.

BTR0 Lets you enter the value for the Bit Timing Register 0. The value must be 8 bits long and in hexadecimal format.

BTR1 Lets you enter the value for the Bit Timing Register 1. The value must be 8 bits long and in hexadecimal format.

Controller settings selection frame

Summarizes the settings resulting from the advanced configuration. The read-only list displays the controller frequency and the specified or calculated combinations for BTR0, BTR1, SP, SJW, TSEG1, TSEG2, BRP and SMPL.

Frequency Displays the CAN controller frequency. The value is determined automatically.

BTR0, BTR1, ... Displays the possible combinations for BTR0, BTR1, SP, SJW, TSEG1, TSEG2, BRP, and SMPL. If **Show all valid settings** is selected, you have to select one combination.

Related topics

Basics

Block Description (RTICANMM ControllerSetup).. 43

References

Options Page (RTICANMM ControllerSetup).. 56
Partial Networking Page (RTICANMM ControllerSetup).. 60
Setup Page (RTICANMM ControllerSetup).. 48

CAN Communication Block

Introduction

The **dSPACE_Blocksets/rticanmmlib** library provides a CAN communication block that lets you control the CAN communication.

RTICANMM MainBlock

Where to go from here

Information in this section

Information in other sections

General Information on RTICANMM MainBlock

Where to go from here

Information in this section

Block Description (RTICANMM MainBlock)

Block

Purpose

To set up and configure CAN communication on a CAN controller.

Description

The RTICANMM MainBlock allows you to configure the entire communication on a CAN controller of a dSPACE CAN board. You can configure the CAN communication via a database container (DBC) file, FIBEX file, MAT file, or AUTOSAR system description file.

- *Names of CAN signals*: To identify CAN signals, the RTICANMM MainBlock dialog displays their names together with the names of the corresponding messages:

  `MessageName.SignalName`

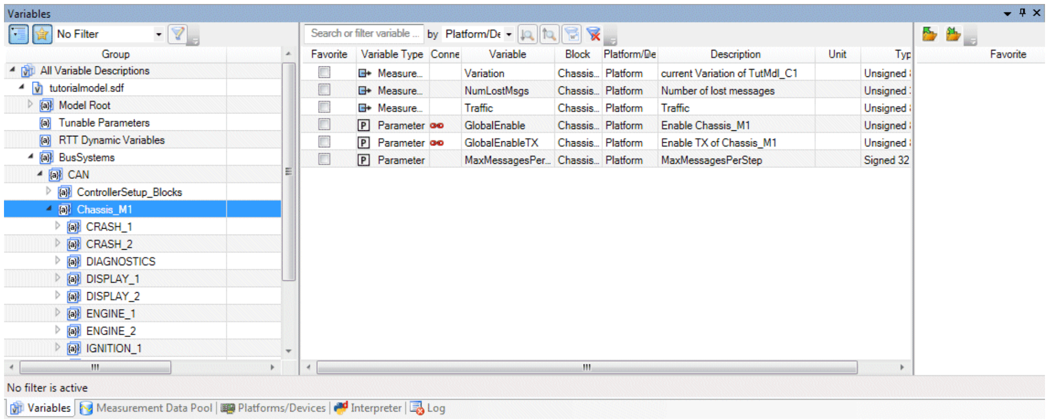
- *S-function*: You have to create a separate S-function for each RTICANMM MainBlock by clicking **Create** in the RTICANMM MainBlock dialog. Together with the S-functions created for the RTICANMM ControllerSetup blocks, the S-functions created for the RTICANMM MainBlocks in the model are included in the model's build process.

  > **Tip**
  >
  > Some pages do not enable **Create** when you change their properties. This applies to all the settings that do not directly affect the generated code. Your changes are stored internally and activated in the next create process. If you want to apply your changes immediately, right-click **Create** and select **Force “Create”**, or select **Force “Create”** from the Options menu.

- *TRC file and ControlDesk*: Whenever you create an S-function for an RTICANMM MainBlock, a TRC file is also automatically created. The TRC file contains entries for all the CAN signals of the CAN controller concerned. The entries allow you to analyze received signals and change the value of signals to be transmitted in ControlDesk.

  When you open the `<model>.sdf` file in ControlDesk, you will find the entries of the RTI CAN MultiMessage Blockset in the **Variables** controlbar of ControlDesk below “BusSystems/CAN”.

The illustration below shows an SDF file in the **Variables** controlbar of ControlDesk:



Note

The CAN controller you want to use and general settings must already have been specified via RTICANMM GeneralSetup and RTICANMM ControllerSetup blocks.

I/O characteristics

The table below describes the available block inports:

Simulink Inport	Range	Simulink Data Type	Description
GlobalEnable	0 … 1	Boolean	(Available only if **GlobalEnable** is selected on the Triggering Options Page (RTICANMM MainBlock)) Activates or deactivates the entire CAN communication of this CAN controller variation.
GlobalEnableTX	0 … 1	Boolean	(Available only if **GlobalEnableTX** is selected on the Triggering Options Page (RTICANMM MainBlock)) Activates or deactivates the transmission of all the messages of this CAN controller variation.
Triggering (struct array)	–	Depends on the struct: ▪ Cyclic: Boolean ▪ Enable: Boolean ▪ Kickout: Boolean ▪ Cycle time: Double ▪ Delay time: Double ▪ ECUEnable: Boolean	(Available only if **Enable**, **Cyclic**, **Kickout** and/or **Network Node Enable** is specified (see Triggering Options Page (RTICANMM MainBlock) on page 136), or if the cycle time or delay time of at least one TX message is specified from within the model: see TX Cycle Time Page (RTICANMM MainBlock) on page 154 and TX Delay Time Page (RTICANMM MainBlock) on page 156) Defines the triggering options **Enable**, **Cyclic** and/or **Kickout** separately for each TX message. Also defines the cycle time and delay time of TX messages. To ensure the input structure is correct, you can let the Triggering to RTICANMM Block be connected to this inport automatically.
TX Data (struct array)	–	Depends on the struct	(Available only if at least one signal is selected as a TX model signal on the Model Signals (TX) Page (RTICANMM MainBlock))

Simulink Inport	Range	Simulink Data Type	Description
			Defines the TX signals to be transmitted from the model. To ensure the input structure is correct, you can let the Mapping to RTICANMM Block be connected to this inport automatically. You can specify the Simulink data type of this inport. Refer to Code Options Page (RTICANMM MainBlock) on page 257.
TX Data Gateway (struct array)	–	Depends on the struct	(Available only if at least one gateway signal is selected on the Gateway Signals Page (RTICANMM MainBlock)) Defines gateway signals from another RTICANMM MainBlock. You can specify the Simulink data type of this inport. Refer to Code Options Page (RTICANMM MainBlock) on page 257.

The table below describes the available block outports:

Simulink Outport	Range	Simulink Data Type	Description
Enable	0 … 1	Boolean	(Available only if **Create enable output port** is selected on the Peripheral Options Page (RTICANMM MainBlock)) Outputs the enable status of the current CAN controller variation. The output is `1` if the **GlobalEnable** on page 137 condition is `true` (see Triggering Options Page (RTICANMM MainBlock) on page 136) and if the variation of this RTICANMM MainBlock is active (see General Settings Page (RTICANMM MainBlock) on page 100).

RX Data (struct array)	–	Depends on the struct:
		▪ RX Status: Boolean
		▪ TX Status: Boolean
			(Available only if one of the following conditions is fulfilled:
			▪ At least one signal is selected as an RX model signal on the Model Signals (RX) Page (RTICANMM MainBlock), or
			▪ At least one capture message is specified on the Capture Messages Page (RTICANMM MainBlock) and the **Create outport for captured messages** option is activated, or
			▪ At least one of the following ports is enabled:
			  ▪ TX Status Ports Page (RTICANMM MainBlock)
			  ▪ RX Status and Time Ports Page (RTICANMM MainBlock)
			  ▪ RX ID Port Page (RTICANMM MainBlock)
			  ▪ RX Raw Data Ports Page (RTICANMM MainBlock)
			  ▪ RX Error Ports Page (RTICANMM MainBlock))
			Puts received RX signals in the model. Outputs status information and errors of TX messages and RX messages.
			You can specify the Simulink data type of this outport. Refer to Code Options Page (RTICANMM MainBlock) on page 257.

TransmittedTime	–	Double	(Available only if a GTS message is selected on the TX Messages Page (RTICANMM MainBlock).) Provides the time that is transmitted via the SYNC and FUP messages via the CAN bus. The value is the sum of the 'seconds' part of the transmitted time, the 'nanoseconds' part of the transmitted time, and the 'overflow of seconds' part transmitted in the FUP message. The value is updated once the FUP message has been transmitted and contains the time that was valid when the SYNC message's end of message (EOF) was on the bus.

Simulink Outport	Range	Simulink Data Type	Description
FupDelay	–	Double	(Available only if a GTS message is selected on the TX Messages Page (RTICANMM MainBlock).) Outputs the time difference between the sending time of the SYNC message and the sending time of the FUP message. The value is updated once the FUP message has been received as a loopback message.

Rx_Error	–		(Available only if a GTS message is selected on the RX Messages Page (RTICANMM MainBlock).)
			Outputs information on the RX error status of a time synchronization message. When a SYNC message is received, the bit values are reset to 0 and then refilled with the current SYNC-message-related bit values (after the SYNC message was received) and the current FUP-message-related bit values (after the FUP message was received). The bits are set as follows:
			▪ Bit 0:
			  ▪ 0: The last message received was a SYNC message.
			  ▪ 1: The last message received was a FUP message.
			▪ Bit 1[1)]:
			  ▪ 0: CRC calculation of SYNC message was successful.
			  ▪ 1: CRC calculation of SYNC message was not successful.
			▪ Bit 2[2)]:
			  ▪ 0: SYNC message counter signal jump is not within the range [0 ... SequenceCounterJumpWidth].
			  ▪ 1: SYNC message counter signal jump is within the range [0 ... SequenceCounterJumpWidth].
			▪ Bit 3:
			  ▪ 0: SYNC message counter signal jump equals 1.
			  ▪ 1: SYNC message counter signal jump is not equal to 1.
			▪ Bit 4[2)]:
			  ▪ 0: SYNC message counter signal jump exceeds the sequence counter jump width.
			  ▪ 1: SYNC message counter signal jump is smaller than or equal to the sequence counter jump width.
			▪ Bit 5[1)]:
			  ▪ 0: CRC calculation of FUP message was successful.
			  ▪ 1: CRC calculation of FUP message was not successful.
			▪ Bit 6:
			  ▪ 0: SYNC counter value and FUP counter values differ.
			  ▪ 1: SYNC counter value and FUP counter value are the same.
			▪ Bit 7:
			  ▪ 0: FUP message arrived after the FUP timeout expired.
			  ▪ 1: FUP message arrived before the FUP timeout expired.

TimeToUpdate	–	Double	(Available only if a GTS message is selected on the RX Messages Page (RTICANMM MainBlock).) Provides the time to be written to the time base manager instance of each time slave of the message. The value is the sum of the

Simulink Outport	Range	Simulink Data Type	Description
			'seconds' part of the time received in the SYNC message, the 'nanoseconds' part of the time received in the FUP message, and the delay time caused by message transmission and CPU processing time, for example.

StbMStatus	–	UInt8	(Available only if a GTS message is selected on the RX Messages Page (RTICANMM MainBlock). One port is generated per slave defined in the GTS message.)
			Provides information on the time base synchronization status for each time base manager instance. The value is updated for each slave in every sampling step. The bits are set as follows:
			▪ Bit 0 (LSB) - TIMEOUT:
			  ▪ 0: No synchronization timeout.
			  ▪ 1: Timeout on receiving synchronization messages.
			▪ Bit 1 - Reserved:
			  ▪ 0: The bit is always set to 0.
			▪ Bit 2 - SYNC_TO_GATEWAY:
			  ▪ 0: The local time base is synchronous to global time master.
			  ▪ 1: The local time base is synchronous to an ECU subordinate to the global time master, i.e., local time base updates are based on a time gateway below the global time master.
			▪ Bit 3 - GLOBAL_TIME_BASE:
			  ▪ 0: The local time base has never been synchronized with the global time master. The local time base is based only on a local time base reference clock.
			  ▪ 1: The local time base has been synchronized with the global time master at least once since the start of the real-time application.
			▪ Bit 4 - TIMELEAP_FUTURE:
			  ▪ 0: No leap into the future within the received time for the time base.
			  ▪ 1: Leap into the future within the received time for the time base exceeds a configured threshold.
			▪ Bit 5 - TIMELEAP_PAST:
			  ▪ 0: No leap into the past within the received time for the time base.
			  ▪ 1: Leap into the past within the received time for the time base exceeds a configured threshold.

TimeBeforeUpdate	–	Double	(Available only if a GTS message is selected on the RX Messages Page (RTICANMM MainBlock). One port is generated per slave defined in the GTS message.) Outputs the time that is currently written in the time base manager instance for a time slave before the time value is updated with the **TimeToUpdate** value. The value is updated if the necessary condition[3)] for updating a time base manager instance is fulfilled or the `ValidationEnable` TRC variable is set to 0.

Simulink Outport	Range	Simulink Data Type	Description
TimeUpdateDelta	–	Double	(Available only if a GTS message is selected on the RX Messages Page (RTICANMM MainBlock). One port is generated per slave defined in the GTS message.) Outputs the time difference between the **TimeToUpdate** value and the **TimeBeforeUpdate** value. The value is updated if the necessary condition[3] for updating a time base manager instance is fulfilled or the `ValidationEnable` TRC variable is set to 0.

[1] The bit value is not updated for unsecured time synchronization messages.

[2] The sequence counter jump width is defined for each slave in a GTS message. However, only the sequence counter jump width value of the *first* slave defined in a GTS message is always used to determine the bit value.

[3] For time slaves, the TRC variable `ValidationEnable` is generated, which lets you override the necessary condition for updating the time in the time base manager instance. The necessary condition is defined by the **Rx_Error** value: For a secured time synchronization message, bits 0, 1, 2, 5, and 6 must be set to 1, and bit 7 must be 0. For an unsecured time synchronization message, bits 0 and 6 must be 1, and bit 7 must be 0. To override the necessary condition and ignore the **Rx_Error** value, the `ValidationEnable` variable must be set to 0.

Dialog menu

Settings global to the RTICANMM MainBlock can be specified on the Settings menu. Refer to Settings Menu (RTICANMM MainBlock) on page 86.

Context menu and buttons of the dialog tree

The dialog settings are structured on several dialog pages and can be accessed via the dialog tree on the left of the dialog.

To improve handling, you can hide dialog pages from the dialog tree or create your own structure of dialog pages. The necessary settings are available from the buttons and context menu of the dialog tree. Refer to Commands of the Dialog Tree (RTICANMM MainBlock) on page 88.

For an overview of all dialog pages, refer to Dialog Tree (RTICANMM MainBlock) on page 75.

Context menu and buttons of the message view

To improve handling, all the settings belonging to a message or a signal can be specified on a separate page. These message and signal pages are located on the message view. The message view can be accessed by clicking **Settings - Open Settings Message View** in the RTICANMM MainBlock dialog. Refer to Message View (RTICANMM MainBlock) on page 79.

You can display the message and/or signal pages according to your needs. The necessary settings are available from the buttons and context menu of the message view. Refer to Commands of the Message View (RTICANMM MainBlock) on page 90.

For an overview of all dialog pages, refer to Pages of RTICANMM MainBlock (Message View) on page 263.

Related topics

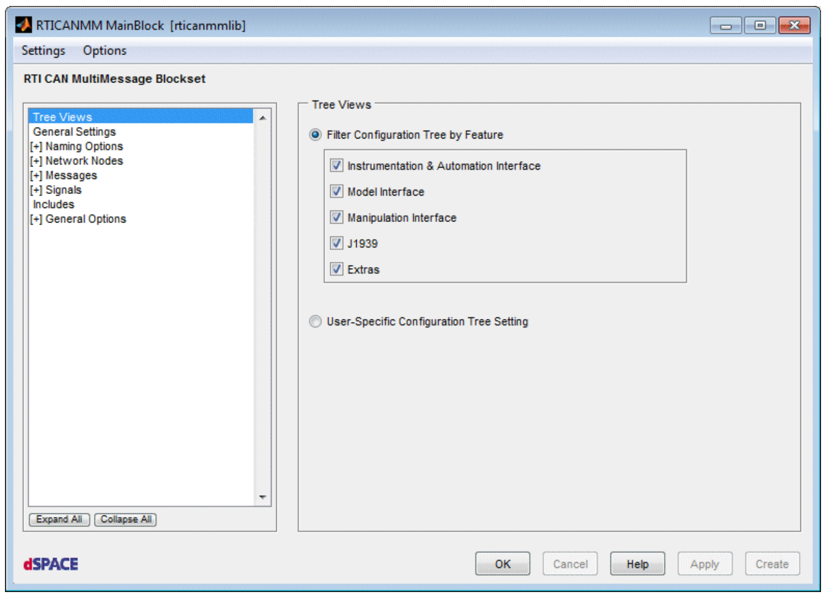
Basics

Lesson 1: Building Your First Model Using the RTI CAN MultiMessage Blockset (RTI CAN MultiMessage Blockset Tutorial)

Dialog Tree (RTICANMM MainBlock)

Dialog tree view

The RTICANMM MainBlock has various dialog pages which can be accessed using the dialog tree.



By default, the dialog pages are collapsed. They can be expanded by clicking Expand All.

Note

If the dialog tree pages are not displayed, the message view is activated. Click OK to close the message view and return to the dialog tree. For information on the message view, refer to Message View (RTICANMM MainBlock) on page 79.

Filter configuration

You do not have to work with all the available dialog pages. For example, if you do not want to define J1939-compliant CAN communication, you can hide the corresponding pages in the dialog tree. The RTICANMM MainBlock provides predefined filters, so you can reduce the size of the dialog tree for easier handling. The following table describes the predefined filter settings.

Number	Name	Description
1	Minimum	The dialog pages are always available.
2	Instrumentation & Automation Interface	The dialog pages allow the access of messages via the TRC file in ControlDesk.
3	Model Interface	The dialog pages are necessary to link RX and TX messages to other blocks of the Simulink model.
4	Manipulation Interface	The dialog pages are necessary to manipulate messages.
5	J1939	The dialog pages are necessary to set up J1939-compliant CAN communication.
6	Extras	Dialog pages for advanced handling of the RTI CAN MultiMessage Blockset

You can set the filter configuration on the Tree Views Page (RTICANMM MainBlock) on page 98.

Note

If you change the filter settings, the parameters of the hidden pages are set to their default values.

Dialog Pages

The dialog settings can be specified on the following pages (listed in the maximum tree). The Filter Configuration column shows which page is affected by which filter configuration.

Dialog Pages of the Maximum Tree	Filter Configuration					
	1	2	3	4	5	6
Tree Views Page (RTICANMM MainBlock)	✓	-	-	-	-	-
General Settings Page (RTICANMM MainBlock)	✓	-	-	-	-	-
File Settings Page (RTICANMM MainBlock)	✓	-	-	-	-	-
Naming Options Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Naming Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Naming Mapping Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Network Nodes Page (RTICANMM MainBlock)	✓	-	-	-	-	-
J1939 Network Node Configuration Page (RTICANMM MainBlock)	-	-	-	-	✓	-
Network Node Selection Page (RTICANMM MainBlock)	-	-	-	-	✓	-
Network Node Identification Page (RTICANMM MainBlock)	-	-	-	-	✓	-
Network Node Interfaces Page (RTICANMM MainBlock)	✓	-	-	-	-	-
Network Node Enable Page (RTICANMM MainBlock)	✓	-	-	-	-	-

Dialog Pages of the Maximum Tree	Filter Configuration					
	1	2	3	4	5	6
Messages Page (RTICANMM MainBlock)	✓	-	-	-	-	-
J1939 Page (RTICANMM MainBlock)	-	-	-	-	✓	-
Source Destination Mapping Page (RTICANMM MainBlock)	-	-	-	-	✓	-
Instance to Container Page (RTICANMM MainBlock)	-	-	-	-	✓	-
Choose Messages Page (RTICANMM MainBlock)	✓	-	-	-	-	-
Network Node Preselection Page (RTICANMM MainBlock)	✓	-	-	-	-	-
TX Messages Page (RTICANMM MainBlock)	✓	-	-	-	-	-
RX Messages Page (RTICANMM MainBlock)	✓	-	-	-	-	-
Free Raw Messages Page (RTICANMM MainBlock)	✓	-	-	-	-	-
Capture Messages Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Triggering Page (RTICANMM MainBlock)	-	✓	✓	-	-	-
Triggering Options Page (RTICANMM MainBlock)	-	✓	✓	-	-	-
Message Enable Page (RTICANMM MainBlock)	-	✓	✓	-	-	-
Message Cyclic Page (RTICANMM MainBlock)	-	✓	✓	-	-	-
Message Kickout Page (RTICANMM MainBlock)	-	✓	✓	-	-	-
Cycle Times Page (RTICANMM MainBlock)	-	-	✓	✓	-	✓
Cycle Time Defaults Page (RTICANMM MainBlock)	-	-	-	-	-	✓
TX Cycle Time Page (RTICANMM MainBlock)	-	-	✓	✓	-	-
TX Delay Time Page (RTICANMM MainBlock)	-	-	✓	✓	-	-
Base/Update Time Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Base/Update Messages Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Base/Update Time Page (RTICANMM MainBlock)	-	-	-	-	-	✓
TX Timeout Enable Page (RTICANMM MainBlock)	-	-	-	✓	-	-
Trigger Reaction Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Mapping Page (RTICANMM MainBlock)	-	-	✓	-	-	✓
Ports & Displays Page (RTICANMM MainBlock)	-	✓	✓	✓	-	✓
TX Status Ports Page (RTICANMM MainBlock)	-	-	✓	-	-	-
RX Status and Time Ports Page (RTICANMM MainBlock)	-	-	✓	-	-	-
RX ID Port Page (RTICANMM MainBlock)	-	-	✓	-	-	-
RX Message Length Port Page (RTICANMM MainBlock)	-	-	✓	-	-	-
RX Message Counter Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Raw Data Page (RTICANMM MainBlock)	-	✓	✓	✓	-	-
TX Raw Data Page (RTICANMM MainBlock)	-	-	✓	✓	-	-
TX Raw Data Display Page (RTICANMM MainBlock)	-	✓	-	-	-	-
RX Raw Data Ports Page (RTICANMM MainBlock)	-	-	✓	-	-	-
RX Raw Data Display Page (RTICANMM MainBlock)	-	✓	-	-	-	-
Errors Page (RTICANMM MainBlock)	-	✓	✓	✓	-	✓
RX Error Ports Page (RTICANMM MainBlock)	-	-	✓	✓	-	-
RX Error Display Page (RTICANMM MainBlock)	-	✓	-	✓	-	-
Cycle Time Error Page (RTICANMM MainBlock)	-	✓	✓	-	-	✓

Dialog Pages of the Maximum Tree	Filter Configuration					
	1	2	3	4	5	6
Message Manipulation Page (RTICANMM MainBlock)	-	-	-	✓	-	✓
Manipulation Options Page (RTICANMM MainBlock)	-	-	-	✓	-	-
TX ID Page (RTICANMM MainBlock)	-	-	-	✓	-	-
J1939 TX ID Page (RTICANMM MainBlock)	-	-	-	-	✓	-
TX Message Length Page (RTICANMM MainBlock)	-	-	-	✓	-	-
Message Defaults Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Message Variations Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Checksum Page (RTICANMM MainBlock)	-	-	-	✓	-	-
Checksum Definition Page (RTICANMM MainBlock)	-	-	-	✓	-	-
Checksum Messages Page (RTICANMM MainBlock)	-	-	-	✓	-	-
Secure Onboard Communication Page (RTICANMM MainBlock)	-	-	-	✓	-	-
Custom Code Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Signals Page (RTICANMM MainBlock)	✓	-	-	-	-	-
TX Page (RTICANMM MainBlock)	✓	-	-	-	-	-
Model Signals (TX) Page (RTICANMM MainBlock)	✓	-	-	-	-	-
Input Manipulation Page (RTICANMM MainBlock)	-	-	-	✓	-	✓
Saturation Options Page (RTICANMM MainBlock)	-	-	✓	✓	-	✓
Saturation Page (RTICANMM MainBlock)	-	-	✓	-	-	-
Signal Defaults Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Signal Ranges Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Signal Errors Page (RTICANMM MainBlock)	-	-	-	✓	-	-
Signal Mappings Page (RTICANMM MainBlock)	-	-	✓	✓	-	-
Signal Manipulation Page (RTICANMM MainBlock)	-	-	-	✓	-	✓
Custom Signal Manipulation Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Dynamic Signal Page (RTICANMM MainBlock)	-	-	-	✓	-	-
Dynamic Signal Values Page (RTICANMM MainBlock)	-	-	-	✓	-	-
Dynamic Signal Defaults Page (RTICANMM MainBlock)	-	-	-	✓	-	-
Toggle Page (RTICANMM MainBlock)	-	-	-	✓	-	-
Parity Page (RTICANMM MainBlock)	-	-	-	✓	-	-
Counter Page (RTICANMM MainBlock)	-	-	-	✓	-	-
Signal Default Manipulation Page (RTICANMM MainBlock)	-	-	-	✓	-	-
Model Signals (RX) Page (RTICANMM MainBlock)	✓	-	-	-	-	-
Gateway Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Gateway Signals Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Gateway Defaults Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Signal Default Manipulation Page (Gateway) (RTICANMM MainBlock)	-	-	-	-	-	✓
Includes Page (RTICANMM MainBlock)	-	-	-	-	-	✓

Dialog Pages of the Maximum Tree	Filter Configuration					
	1	2	3	4	5	6
General Options Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Experimental Software Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Code Options Page (RTICANMM MainBlock)	-	-	-	-	-	✓
TRC Page (RTICANMM MainBlock)	-	-	-	-	-	✓
TRC Options Page (RTICANMM MainBlock)	-	-	-	-	-	✓
TRC Extras Page (RTICANMM MainBlock)	-	-	-	-	-	✓
Peripheral Options Page (RTICANMM MainBlock)	-	-	-	-	-	✓

Related topics

References

Commands of the Dialog Tree (RTICANMM MainBlock)........ 88
Tree Views Page (RTICANMM MainBlock)........ 98

Message View (RTICANMM MainBlock)

Access

If a database file is specified for the RTICANMM MainBlock, you can activate the message view by clicking **Settings - Open Settings Message View** in the RTICANMM MainBlock dialog.

Purpose

Lets you specify all the settings of the messages and their signals sent by an ECU and display information on the specified settings.

Description

The message view is only available if you specify a database file for the RTICANMM MainBlock. The message view lets you specify all the settings of the messages and their signals sent by an ECU. In contrast to the dialog tree and to improve handling, the settings belonging to one message/ one signal are not specified on several pages but on a separate page. For an overview of the specified settings of a message or a signal, you can display an information page (refer to Commands of the Message View (RTICANMM MainBlock) on page 90). General settings of the RTICANMM MainBlock cannot be specified in the message view but only in the dialog tree. To change from the message view to the dialog tree, you have to click **OK**.

Dialog tree and message view The message view and the dialog tree depend on each other. As a consequence, all the pages of the dialog tree must be displayed if you want to specify the settings of a message or signal (refer to Tree Views Page (RTICANMM MainBlock) on page 98).

Note

If you specify settings in the message view while the appropriate pages in the dialog tree are hidden, all the settings that relate to the hidden pages are reset to their defaults.

If you want to reduce the dialog tree after you specified settings in the message view, select User Specific Configuration Tree Settings on the Tree Views Page (RTICANMM MainBlock) on page 98 and choose Delete all default pages from the context menu of the dialog tree (refer to Commands of the Dialog Tree (RTICANMM MainBlock) on page 88). If you reduce the dialog tree by its predefined selection of dialog pages (Filter Configuration View by Feature on the Tree Views Page (RTICANMM MainBlock) on page 98), you are warned if one or more pages have user-defined settings.

Dialog pages

The message/ signal settings can be specified on the following pages (listed in the Infos/Settings - ECU/Message/Signal- view):

- ECU Page (RTICANMM MainBlock Message View) on page 264
- Messages Container Page (RTICANMM MainBlock Message View) on page 265
- Message Page (RTICANMM MainBlock Message View) on page 266
- Message Info Page (RTICANMM MainBlock Message View) on page 271
- Signals Container Page (RTICANMM MainBlock Message View) on page 271
- Signal Page (RTICANMM MainBlock Message View) on page 272
- Signal Info Page (RTICANMM MainBlock Message View) on page 277

Related topics

References

Settings Menu (RTICANMM MainBlock)..86

Configuration File

Introduction

Configuration files are M files. You can use them to store configuration settings of many pages of the RTICANMM MainBlock.

Creating a configuration file

You can create a configuration file in two ways:

Creating a configuration file via GUI You can create a configuration file easily with the Configuration File – Create context menu command on the relevant pages of the RTICANMM MainBlock. The Configuration File – Add context menu command lets you easily add the settings from a page to an existing configuration file.

Creating a configuration file by hand You can also create a configuration file by hand. The first line of the file must be:

```
function configdata = <mfilename>()
```

The list below shows the structure the configuration file must have:

Variable Name	Variable Type	Corresponding Page of the RTICANMM MainBlock
`configdata.<MessageName>.cycletime.cycletime` `configdata.<MessageName>.cycletime.delaytime`	Integer value ≥ `0` (in milliseconds) Integer value ≥ `0` (in milliseconds)	Cycle Time Defaults Page (RTICANMM MainBlock) on page 151
`configdata.<MessageName>.cycletime.ranges.min` `configdata.<MessageName>.cycletime.ranges.max`	Integer value ≥ `0` (in milliseconds) Integer value ≥ `0` (in milliseconds)	Cycle Time Error Page (RTICANMM MainBlock) on page 187

`configdata.<MessageName>.Cyclic.Source`
`configdata.<MessageName>.Cyclic.Logic`
`configdata.<MessageName>.Cyclic.DefaultSource`
`configdata.<MessageName>.Cyclic.Default`
	`'NONE'`, `'TRC'`, `'INPORT'`, or `'BOTH'`
	`'AND'`, `'OR'`, or `'-'`
	`'TRC'`, `'INPORT'`, `'BOTH'`, or `'-'`
	`'CYCLIC'`, `'TRIGGERED'`, or `'-'`
		Message Cyclic Page (RTICANMM MainBlock) on page 144

`configdata.<MessageName>.J1939Mapping = 'ATA-FF;WTA-FF;TCO-FF;K-FF';`	String The syntax of the string is the same as for the DBC file. Refer to Mapping in the DBC file on page 123.	Source Destination Mapping Page (RTICANMM MainBlock) on page 122
`configdata.<MessageName>.timeout` `configdata.<MessageName>.timeoutcounter`	Integer value ≥ `0` `'STOP'` or `'CONTINUE'`	TX Timeout Enable Page (RTICANMM MainBlock) on page 161

`configdata.<MessageName>.triggermapping.ECU_Enable`
`configdata.<MessageName>.triggermapping.Cyclic`
`configdata.<MessageName>.triggermapping.Kickout`
`configdata.<MessageName>.triggermapping.Enable`
`configdata.<MessageName>.triggermapping.CycleTime`
`configdata.<MessageName>.triggermapping.DelayTime`
`configdata.<MessageName>.triggermapping.BaseTime`
`configdata.<MessageName>.triggermapping.UpdateTime`
`configdata.<MessageName>.triggermapping.UpdateNum`
`configdata.<MessageName>.triggermapping.MessageVariation`
	String of Bus Selector structure (e.g., <CANname>.<ECUName>.<MsgName>.Cyclic)
	The following macros are available for defining a mapping structure:
	▪ `'%CAN'`
	▪ `'%Controller'`
	▪ `'%ECU'`
	▪ `'%ReceiveECU'`
	▪ `'%MsgName'`
	▪ `'%MsgID'`
	▪ `'%SigName'`
		Mapping Page (RTICANMM MainBlock) on page 168

`configdata.<MessageName>.<SignalName>.crc.default` `configdata.<MessageName>.<SignalName>.crc.algorithm`	`'Enable'` or `'Disable'` String of algorithm name	Checksum Messages Page (RTICANMM

Variable Name	Variable Type	Corresponding Page of the RTICANMM MainBlock
		MainBlock) on page 205
configdata.<MessageName>.<SignalName>.counter.start configdata.<MessageName>.<SignalName>.counter.step configdata.<MessageName>.<SignalName>.counter.divisor configdata.<MessageName>.<SignalName>.counter.steplength configdata.<MessageName>.<SignalName>.counter.max configdata.<MessageName>.<SignalName>.counter.CounterContinueMode	Integer value Integer value Integer value ≥ 0 Integer value ≥ 0 Integer value '0' for Continue Counting, '1' for Set to Constant, or '2' for Stop Counting	Counter Page (RTICANMM MainBlock) on page 240
configdata.<MessageName>.<SignalName>.dynvalue.value configdata.<MessageName>.<SignalName>.dynvalue.countdown	Any numerical value Integer value ≥ 0	Dynamic Signal Defaults Page (RTICANMM MainBlock) on page 235
configdata.<MessageName>.<SignalName>.parity.parity configdata.<MessageName>.<SignalName>.parity.signals	'odd' or 'even' String of parity signals separated by ;	Parity Page (RTICANMM MainBlock) on page 238
configdata.<MessageName>.<SignalName>.saturation	'Both', 'Input', 'Output', or 'No'	Saturation Page (RTICANMM MainBlock) on page 218
configdata.<MessageName>.<SignalName>.manipulationdefault configdata.<MessageName>.<SignalName>.manipulationfields	String for selected default manipulation Cell array with string for available manipulation options Possible strings: ▪ 'Input' ▪ 'Input/Gateway' ▪ 'Constant' ▪ 'Constant/Gateway' ▪ 'DynValue' ▪ 'Toggle' ▪ 'Counter' ▪ 'IncrementCounter' ▪ 'Parity' ▪ 'InversParity' ▪ 'Error' (The default manipulation options, which can be selected for the concrete signal, are exported in the "manipulationfields" field.)	Signal Default Manipulation Page (RTICANMM MainBlock) on page 244
configdata.<MessageName>.<SignalName>.default	Any numerical value	Signal Defaults Page (RTICANMM MainBlock) on page 221

Variable Name	Variable Type	Corresponding Page of the RTICANMM MainBlock
configdata.<MessageName>.<SignalName>.error	[] for no error, or any numerical value	Signal Errors Page (RTICANMM MainBlock) on page 225
configdata.<MessageName>.<SignalName>.signalbusmapping	String of BusSelector structure	Signal Mappings Page (RTICANMM MainBlock) on page 227
configdata.<MessageName>.<SignalName>.signalrange.min configdata.<MessageName>.<SignalName>.signalrange.max	Any numerical value Any numerical value	Signal Ranges Page (RTICANMM MainBlock) on page 223
configdata.<NetworkNodeName>.ECUEnable.Source configdata.<NetworkNodeName>.ECUEnable.Logic configdata.<NetworkNodeName>.ECUEnable.DefaultSource configdata.<NetworkNodeName>.ECUEnable.Default	'NONE', 'TRC', 'INPORT', or 'BOTH' 'AND', 'OR', or '-' 'TRC', 'INPORT', 'BOTH', or '-' 'ENABLE', 'DISABLE', or '-'	Network Node Enable Page (RTICANMM MainBlock) on page 117
configdata.<MessageName>.instance2container.instance configdata.<MessageName>.instance2container.active configdata.<MessageName>.instance2container.default	String for instance name 'Active' or 'Inactive' 'ENABLE', 'DISABLE', or '-'	Instance to Container Page (RTICANMM MainBlock) on page 125
configdata.<MessageName>.Enable.Source configdata.<MessageName>.Enable.Logic configdata.<MessageName>.Enable.DefaultSource configdata.<MessageName>.Enable.Default	'NONE', 'TRC', 'INPORT', or 'BOTH' 'AND', 'OR', or '-' 'TRC', 'INPORT', 'BOTH', or '-' 'ENABLE', 'DISABLE', or '-'	Message Enable Page (RTICANMM MainBlock) on page 140
configdata.<MessageName>.Kickout.Source	'NONE', 'TRC', 'INPORT', or 'BOTH'	Message Kickout Page (RTICANMM MainBlock) on page 148
configdata.<MessageName>.baseupdate.basetime configdata.<MessageName>.baseupdate.updatetime configdata.<MessageName>.baseupdate.number	Integer value > 0 (in milliseconds) Integer value > 0 (in milliseconds) Integer value > 0	Base/Update Time Page (RTICANMM MainBlock) on page 159
configdata.<MessageName>.rxmessagecounter.destination	'NONE', 'TRC', 'OUTPORT', or 'BOTH'	RX Message Counter Page (RTICANMM MainBlock) on page 176
configdata.<MessageName>.TXRAWData.active configdata.<MessageName>.TXRAWData.Source configdata.<MessageName>.TXRAWData.default	'Active' or 'Inactive' 'Inport' or 'TRC' 'Raw Data' or 'Signals'	TX Raw Data Page (RTICANMM MainBlock) on page 178

Variable Name	Variable Type	Corresponding Page of the RTICANMM MainBlock
configdata.<MessageName>.ID.AdjustOption	'NONE', 'TRC', 'INPORT', 'BOTH (DEFAULT TRC)', or 'BOTH (DEFAULT INPORT)'	TX ID Page (RTICANMM MainBlock) on page 191
configdata.<MessageName>.Priority.AdjustOption configdata.<MessageName>.SrcDest.AdjustOption configdata.<MessageName>.Addressing.AdjustOption	'NONE', 'TRC', 'INPORT', 'BOTH (DEFAULT TRC)', or 'BOTH (DEFAULT INPORT)' 'NONE', 'TRC', 'INPORT', 'BOTH (DEFAULT TRC)', or 'BOTH (DEFAULT INPORT)' 'NODE INDEX' or 'ADDRESS'	J1939 TX ID Page (RTICANMM MainBlock) on page 193
configdata.<MessageName>.MessageLength.AdjustOption configdata.<MessageName>.MessageLength.Default	'None', 'TRC', or 'INPORT' Integer value in the range ▪ 1 ... 8 (standard messages) ▪ 1 ... 64 (CAN FD messages) ▪ 1 ... 1785 (J1939 messages)	TX Message Length Page (RTICANMM MainBlock) on page 195
configdata.<MessageName>.DefaultValue	String for default value (e.g., '0x1234')	Message Defaults Page (RTICANMM MainBlock) on page 197
configdata.<MessageName>.MessageVariation.default configdata.<MessageName>.MessageVariation.source	String for variation name 'ONLY DEFAULT', 'TRC', or 'INPORT'	Message Variations Page (RTICANMM MainBlock) on page 198
configdata.<MessageName>.<SignalName>.gw.manipulationdefault configdata.<MessageName>.<SignalName>.gw.manipulationfields	String for default signal manipulation option Cell array with string for available signal manipulation options Possible strings: ▪ 'Input' ▪ 'Input/Gateway' ▪ 'Constant' ▪ 'Constant/Gateway' ▪ 'DynValue' ▪ 'Toggle' ▪ 'Counter' ▪ 'IncrementCounter' ▪ 'Parity' ▪ 'InversParity' ▪ 'Error' ▪ 'GTS' (The default manipulation options, which can be selected for the concrete signal, are exported in the "manipulationfields" field.)	Signal Default Manipulation Page (Gateway) (RTICANMM MainBlock) on page 251
configdata.<MessageName>.<SignalName>.toggleperiod	Numerical value ≥ 0 (toggle period in seconds)	Toggle Page (RTICANMM

Variable Name	Variable Type	Corresponding Page of the RTICANMM MainBlock
		MainBlock) on page 236

Note

- Incorrect naming of signals and messages in configuration files leads to errors when you load them. Use the correct variable types for the variables.
- Messages whose names begin with an underscore are ignored.

Related topics

References

Commands, Menus and Dialogs of RTICANMM MainBlock

Where to go from here

Information in this section

Settings Menu (RTICANMM MainBlock)

Purpose

To handle the settings of the RTICANMM MainBlock.

Description

The RTICANMM MainBlock stores its settings in a MAT file and/or an intermediate file. The **Settings** menu provides you with commands to work with both files:

- *MAT file*: Whenever you create an S-function for an RTICANMM MainBlock, a MAT file containing the block's settings is also created. The MAT file contains the settings of the RTICANMM MainBlock and also the settings relating to the database file specified for the RTICANMM MainBlock. Whenever you open an RTICANMM MainBlock again, its settings are automatically loaded from the MAT file.
- *Intermediate file*: If you close an RTICANMM MainBlock without creating an S-function for it, an intermediate file is created. This contains the settings of the RTICANMM MainBlock and also the settings relating to the database file specified for the RTICANMM MainBlock, but it may not always correspond to the S-function generated for the block.

Menu commands

Load Settings Lets you load the settings for the current RTICANMM MainBlock. Specify the MAT file with the desired settings. Note that the structure of the dialog tree is not updated. Because all parameters on hidden dialog pages are set to their default values, you are recommended to display the maximum tree before loading the settings. Alternatively, you can load a specific dialog tree structure. To do so, use the **Load tree** command from the context menu of the dialog tree (refer to Commands of the Dialog Tree (RTICANMM MainBlock) on page 88).

Save Settings Lets you save the settings of the RTICANMM MainBlock to a MAT file.

Load Current Page Settings Lets you load the settings for the current page in the RTICANMM MainBlock. Specify the MAT file with the desired settings.

Save Current Page Settings Lets you save the settings of the current page in the RTICANMM MainBlock to a MAT file.

Reset Current Page to Defaults Lets you reset the current page settings to the default.

Reset to Default Settings Lets you reset the RTICANMM MainBlock settings to the default.

Undo Settings Lets you undo the most recent changes to the settings of the current RTICANMM MainBlock. All settings since the last **Close**, **Apply**, or **Create** are undone.

Undo Intermediate Settings Lets you delete the intermediate file, and load the settings of the MAT file. This ensures that the settings correspond to the S-function generated for the block.

Open Settings Message View (Available only if a database file is specified for the RTICANMM MainBlock) Lets you display the message view. For details, refer to Message View (RTICANMM MainBlock) on page 79.

OK Lets you apply the changes in the dialog as intermediate settings and close the dialog. You can also click the OK button of the dialog.

Apply Lets you apply the changes in the dialog as intermediate settings. You can also click the Apply button of the dialog.

Cancel Lets you close the dialog without applying the changes in the dialog. You can also click the Cancel button of the dialog.

Create Lets you create the S-function for the RTICANMM MainBlock.

Tip

Some pages do not enable the Create button of the dialog when you change their properties. Your changes are stored internally and activated in the next create process. If you want to apply your changes immediately, select Create in the settings menu or right-click the Create button and select Force "Create".

Related topics

References

Options Menu (RTICANMM MainBlock)

Purpose

To create the whole MainBlock.

Description

Some pages do not enable Create when you change their properties. This applies to all the settings that do not directly affect the generated code. Your changes are stored internally and activated in the next create process. If you want to apply your changes immediately, select Force "Create", or right-click the Create button and select Force "Create".

Menu commands	**Force "Create"**	Lets you create the whole MainBlock immediately.

Related topics

References

Commands of the Dialog Tree (RTICANMM MainBlock)

Purpose

To create your own structure of dialog pages for the dialog tree.

The commands are available only if you work with the user-specific configuration view settings (see Tree Views Page (RTICANMM MainBlock) on page 98).

Description

The dialog settings are structured on several dialog pages and can be accessed via the dialog tree on the left of the dialog. To improve handling, you can hide dialog pages from the dialog tree or create your own structure of dialog pages.

When adding dialog pages to the dialog tree, you can choose between User pages and RTICANMM pages. The RTICANMM pages contain the actual configuration settings, while user pages let you group the RTICANMM pages to achieve the desired structure.

It is also possible to load preconfigured dialog trees, such as the minimum tree or maximum tree.

Note

If new features are available for the RTICANMM Main Block, they are added automatically only to the maximum tree. You are recommended to use the maximum tree for this reason.

Tip

The structure documented in this reference is that of the maximum tree.

Context menu

Delete page Lets you remove the selected dialog page from the dialog tree.

Note

All the settings in the deleted dialog page are reset to their default values.

Delete all default pages Lets you remove all the dialog pages whose settings are set to their default values.

Add page on current level Lets you place a dialog page on the same level as the selected page. The new dialog page is inserted above the selected one.

- Add user page prompts you for the name of the new dialog page.
- Add RTICANMM page lets you select an RTICANMM dialog page from the list of available dialog pages, i.e., the maximum tree.

  You can only add RTICANMM dialog pages that are not already contained in the dialog tree.

Add page below current level Lets you place a dialog page below the level of the selected page.

- Add user page prompts you for the name of the new dialog page.
- Add RTICANMM page lets you select an RTICANMM dialog page from the list of available dialog pages, i.e., the maximum tree.

  You can only add RTICANMM dialog pages that are not already contained in the dialog tree.

Move selected page Lets you modify the structure of the dialog tree. If a certain move is not possible for the selected dialog page, it is ignored.

- Up/ Down lets you move the selected dialog page vertically in the dialog tree.
- Level up/ Level down lets you move the selected dialog page horizontally in the dialog tree.

Preconfigured trees Lets you select one of the preconfigured dialog trees:

- Minimum tree provides you with the dialog pages that are absolutely essential for configuring the Communication Block.
- Maximum tree provides you with all the available dialog pages.

Direct Link Lets you directly select a page located anywhere in the dialog tree. The Direct Link menu structure always corresponds to the maximum tree.

Load tree Lets you load a previously saved structure of the dialog tree from a MAT file.

Save tree Lets you save the current structure of the dialog tree to a MAT file.

Buttons

Expand All Lets you expand all the nodes of the dialog tree.

Collapse All Lets you collapse all the nodes of the dialog tree.

Related topics

References

Dialog Tree (RTICANMM MainBlock).. 75
Tree Views Page (RTICANMM MainBlock).. 98

Commands of the Message View (RTICANMM MainBlock)

Purpose

To display the pages of the message view according to your needs. The message view can be accessed by clicking Settings - Open Settings Message View in the RTICANMM MainBlock dialog.

Description

The message view displays the relevant information on different pages:

- All ECUs which send messages. ECUs which only receive messages are not displayed.
- All messages which are sent by the ECUs.
- All signals belonging to the messages.
- Information on the settings of the messages and the signals.

You can specify the settings belonging to one message/ one signal and/or display information on the specified settings. You can show or hide the pages of the message view according to your needs.

Context menu

Settings - ECU/Message/Signal Lets you display the messages sent by an ECU and the settings of these messages and their signals.

Settings - Message/Signal Lets you display the settings of the messages and their signals. This is the default view.

Settings - Signal Lets you display the settings of the signals.

Infos - ECU/Message/Signal Lets you display the messages sent by an ECU and information on the specified settings of these messages and their signals. The information is view-only.

Infos - Message/Signal Lets you display information on the specified settings of the messages and their signals. The information is view-only.

Infos - Signal Lets you display information on the specified settings of the signals. The information is view-only.

Infos/Settings - ECU/Message/Signal Lets you display the messages sent by an ECU, the settings, and information on the specified settings of these messages and their signals. The information on the specified settings is view-only.

Infos/Settings - Message/Signal Lets you display the settings and information on the specified settings of the messages and their signals. The information on the specified settings is view-only.

Infos/Settings - Signal Lets you display the settings and information on the specified settings of the signals. The information on the specified settings is view-only.

Buttons

Expand All Lets you expand all the nodes of the message view.

Collapse All Lets you collapse all the nodes of the message view.

Related topics

References

Filter Dialog (RTICANMM MainBlock)

Access

This dialog appears if you click **Filter** on one of the following pages:

- Checksum Messages Page (RTICANMM MainBlock) on page 205
- Toggle Page (RTICANMM MainBlock) on page 236
- Counter Page (RTICANMM MainBlock) on page 240

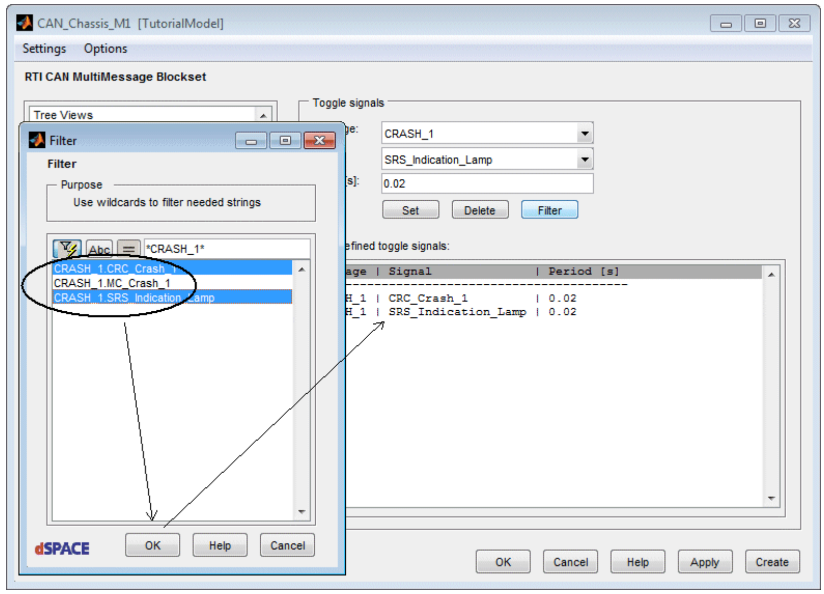
Purpose

To filter and select signals.

Description

This dialog lists the signals of all TX messages and, if you open the dialog from the **Counter Page (RTICANMM MainBlock)** or **Checksum Messages Page (RTICANMM MainBlock)**, of all RX messages.

- *Filtering signals*: You can perform a wildcard search in the list via filter buttons. Enter the search string in the dialog's edit field. You can also enter * as a wildcard. Press the `Enter` key to apply the wildcard filter setting.
- *Selecting signals*: You can select any signals from the (filtered) list. When you click **OK**, your signal selection will be transferred to the list on the page on which you clicked the **Filter** button.



Dialog settings

Filter buttons Let you specify the wildcard filter:

State	Description
	The wildcard filter is not active.
	The wildcard filter is active.
Abc	The wildcard filter is case-sensitive.
ABC	The wildcard filter is case-insensitive.
=	Search results contain the specified wildcard string.
≠	Search results do not contain the specified wildcard string.

OK Transfers the selected signals to the list on the page on which you clicked the Filter button.

Related topics

References

Pages of RTICANMM MainBlock (Dialog Tree)

Access

> **Note**
>
> If the dialog tree pages are not displayed, the message view is activated. Click OK to close the message view and return to the dialog tree. For information on the message view, refer to Message View (RTICANMM MainBlock) on page 79.

Where to go from here

Information in this section

Tree Views Page (RTICANMM MainBlock)

Access

Located at the top level of the dialog tree.

Purpose

To select the dialog pages for working with the RTICANMM MainBlock.

Description

On the **Tree Views** page, you can select dialog pages to display in the dialog tree. All the parameters on the hidden dialog pages are set to the default values. You can use predefined selections of dialog pages (**Filter Configuration View by Feature** option) or work with your own selection of dialog pages (**User Specific Configuration View Setting** option).

> **Note**
>
> If a dialog page is hidden, all its parameters are set to their default values.

Dialog settings

Filter Configuration View by Feature Lets you select the dialog pages necessary for your use case and structure the dialog tree by feature. By default all the features are selected and the maximum tree is displayed.

Feature	Description
Instrumentation & Automation Interface	The dialog tree contains all the dialog pages necessary to set up CAN communication with the **Bus Navigator** of ControlDesk. In this case the RX and TX messages are accessed via the TRC file. The TRC file contains entries for all the CAN signals of the CAN controller concerned. The entries allows you to analyze received signals and change the values of signals to be transmitted. You can enable error displays for RX messages or specify kickouts via entries in the TRC file. This is the fastest method to build the real-time application, because no mapping of inports is necessary. It should therefore be used in large real-time application, for example, applications for hardware-in-the-loop simulation.
Model Interface	The dialog tree contains all the dialog pages necessary to set up CAN communication for a model interface. All the messages are accessed in the Simulink model. The RX messages are available at an outport of the RTICANMM MainBlock block. The TX messages can be read by an inport of the Pages of RTICANMM MainBlock (Dialog Tree) block. You should use the Naming Mapping Page (RTICANMM MainBlock) to map the inports of the block.
Manipulation Interface	The dialog tree contains all the dialog pages necessary to manipulate RX and TX messages.
J1939	The dialog tree contains all the dialog pages necessary to set up J1939-compliant CAN communication.
Extras	The dialog tree contains dialog pages for advanced handling of the RTI CAN MultiMessage Blockset.

You can select one or more features depending on your use case. For a list of the features and the corresponding dialog pages, refer to Dialog Tree (RTICANMM MainBlock) on page 75.

> **Note**
>
> If you change the filter settings, the parameters of the hidden pages are set to their default values.

User Specific Configuration View Setting Lets you specify your own dialog tree. You can add or delete dialog pages using the context menu of the dialog tree. Additionally you can save or load your selection of dialog pages. Refer to Commands of the Dialog Tree (RTICANMM MainBlock) on page 88.

Related topics

References

General Settings Page (RTICANMM MainBlock)

Access

Located on the top level of the dialog tree.

Purpose

To specify a database file and select the CAN controller variation for which you want to set up CAN communication.

Description

Via this page you can:

- Specify the DBC file, FIBEX file, AUTOSAR system description file, or MAT file to be used as the database for this RTICANMM MainBlock.
- Specify to work without a database but only with free raw and/or capture messages.
- Select the CAN controller and controller variation for which you want to set up CAN communication.
- Enable CAN FD mode for this RTICANMM MainBlock.

DBC file as the database

You can use a DBC file as the database for an RTICANMM MainBlock.

- *DBC file parser and DBC file attributes*: The RTI CAN MultiMessage Blockset provides a DBC file parser that allows you to specify a DBC file as the database. The DBC file format was developed by Vector Informatik GmbH, Stuttgart, Germany.

  DBC files may contain default values for message-specific attributes such as the cycle time. Some of these default settings are supported by the RTI CAN MultiMessage Blockset's DBC file parser.
- *Consistency check for DBC files*: The RTI CAN MultiMessage Blockset's DBC file parser automatically performs a consistency check on the DBC file you specify. This check is similar to the consistency check of Vector Informatik's CANdb++ Editor. In other words: For the RTI CAN MultiMessage Blockset, you can use any DBC file that passes the check of the CANdb++ Editor.

- *DBC file with standard and extended messages*: You can use DBC files as database which contain both standard and extended messages. On the Setup Page (RTICANMM ControllerSetup) you can specify which message identifier format (standard or extended) is transmitted with higher priority. In the case of communication overflow, the message identifier format with the higher priority will be transmitted preferentially, only messages of the lower prioritized identifier format might be lost.

FIBEX file as the database

The Field Bus Exchange (FIBEX) format is an XML exchange file format developed by ASAM e. V. It is used for data exchange between different tools that work with message-oriented bus communication.

- *FIBEX file parser*: The RTI CAN MultiMessage Blockset provides a FIBEX file parser that allows you to specify a FIBEX file as the database. Since a FIBEX file usually describes more than one bus system, you have to select one of the available bus systems if you use a FIBEX file as the database.
- *Consistency check for FIBEX files*: The RTI CAN MultiMessage Blockset's FIBEX file parser automatically performs a consistency check on the FIBEX file you specify. A log file is generated which provides information on the bus systems defined in the FIBEX file. Errors, warnings and general information on the messages and signals specified in the bus systems are displayed.

The RTI CAN MultiMessage Blockset supports FIBEX 3.1, FIBEX 4.1, FIBEX 4.1.1, and FIBEX 4.1.2 files.

AUTOSAR system description file as the database

You can use an AUTOSAR system description file as the database for an RTICANMM MainBlock.

AUTOSAR system description files are files of AUTOSAR XML file type that describe a system according to AUTOSAR. A system is a combination of a hardware topology, a software architecture, a network communication, and information on the mappings between these elements. AUTOSAR system description files are instances of the AUTOSAR System Template. The AUTOSAR System Template contains a description of the network communication and hardware topology according to the FIBEX standard defined by ASAM e.V.

The RTI CAN MultiMessage Blockset supports the AUTOSAR System Template based on AUTOSAR Release 3.1.4, 3.2.1, 3.2.2, 4.0.3, 4.1.1, 4.1.2, 4.2.1, 4.2.2, 4.3.0, 4.3.1, and 4.4.0, and AUTOSAR Classic Platform Release R19-11 and R20-11.

- *AUTOSAR XML file parser*: The RTI CAN MultiMessage Blockset provides an AUTOSAR XML file parser that allows you to import an AUTOSAR system description file as the database. Since an AUTOSAR system description file usually describes more than one bus system, you have to select one of the available bus systems if you use an AUTOSAR system description file as the database.
- *Consistency check for AUTOSAR system description files*: The RTI CAN MultiMessage Blockset's AUTOSAR XML file parser automatically performs a consistency check on the AUTOSAR system description file you specify.

You can also import a single AUTOSAR ECU Extract as the database. An AUTOSAR ECU Extract is the extract of a system description that contains information about signals and messages concerning a single ECU.

MAT file as the database

You can use a MAT file generated by the M-script as the database for an RTICANMM MainBlock.

- *Using other database file formats via MAT file*: You can also specify other database file formats as the database via the MAT file format. You must convert your specific database files into the MAT file format for this purpose. The MAT file structure must be like this:

Structure	Range	Type	Description
`dbcdata`	-	Struct	Complete structure of CAN data
`dbcdata.Message(m).`	m = 1 …	Struct	Complete structure of CAN message
`MsgName`	-	String	Message name
`MsgId`	-	Numerical value	Message identifier in decimal format
`MsgIdFormat`	0 … 1	Numerical value	Message identifier format: ▪ 0: standard message identifier ▪ 1: extended message identifier
`MsgLength`	0 … 8	Numerical value	Number of bytes of current message
`ECU`	-	String	ECU sending this message
`MsgComment`	-	Cellarray of strings	Comments for this message
`MsgCycleTime`	0 …	Numerical value	Value of message cycle time
`MsgDelayTime`	0 …	Numerical value	Value of message delay time
`MsgBaseTime`	0 …	Numerical value	Value of message base time
`MsgUpdateTime`	0 …	Numerical value	Value of message update time
`MsgSendType`	-	String	Send type ▪ cyclic ▪ triggered
`MsgStartValue`	'0000000000000000h' … 'ffffffffffffffffh'	Formatted string	Start value of message bits (as a hexadecimal number)
`J1939active`	0 … 1	Numerical value	▪ 0: no J1939 message ▪ 1: J1939 message
`J1939PGPrio`	0 … 7	Numerical value	Priority of a J1939 message
`J1939PGDest`	0 … 255	Numerical value	Destination ECU address of a J1939 message
`J1939PGSrc`	0 … 255	Numerical value	Source ECU address of a J1939 message
`J1939Mapping`	-	String	Mapping to source and destination ECU (for the syntax of the mapping string refer to Source Destination Mapping Page (RTICANMM MainBlock) on page 122)
`SendECU`	-	Cellarray of strings	Source ECUs
`ReceiveECU`	-	Cellarray of strings	Destination ECUs

Structure	Range	Type	Description
`Variations(v).`	-	Struct	Structure of message variations
`Name`	-	String	Name of the variant – must be unique per message
`Signals(n).SgnIdx`	n = 1 … y	Integer	Index of the signals that belong to this variant. ▪ SgnIdx corresponds to s ▪ y is the total number of signals in this variant
`Signal(s).`	s = 1 … 64	Struct	Struct of signal data
`SgnName`	-	String	Name of current signal
`SgnStartBit`	0 … 63	Numerical value	Signal start bit
`SgnLength`	1 … 64	Numerical value	Signal length
`ByteLayout`	1 … 2	Numerical value	1: little endian, LSB first, Intel Standard 2: big endian, MSB first, Motorola Backward (The default value is 1.)
`Factor`	1 …	Numerical value	Scaling factor of current signal
`Offset`	-	Numerical value	Scaling offset of current signal
`SgnComment`	-	String	Comment for current signal
`SgnUnit`	-	String	Signal unit
`Min`	-	Numerical value	Physical minimum value
`Max`	-	Numerical value	Physical maximum value
`ReceiveECUs`	-	Cellarray of strings	ECUs that receive this signal
`SgnType`	0 … 2	Numerical value	0: normal 1: mode signal 2: mode-dependent signal
`DataType`	0 … 1	Numerical value	0: unsigned int 1: signed int
`ModeValue`	-1 …	Numerical value	-1: no mode signal ~= -1: mode value
`InitialValue`	-	Numerical value	Initial value of signal
`SgnErrorValue`	▪ [ ] or ▪ error value	Integer	Signal error value ▪ [ ]: no error value ▪ error value: any value
`ValueNamePairs(p)`	p = 1 …	Struct	Structure of value/name pair
`Name`	-	String	Name of current value
`Value`	-	Numerical value	Current Value

Tip

For an example on working with a MAT file as the database, refer to:
- Lesson 14 (Advanced): Defining CAN Communication via a User-Defined Database (RTI CAN MultiMessage Blockset Tutorial 📖).
- `<RCP_HIL_InstallationPath>\MATLAB\RTICANMM\M\rticanmm_dbcdata_tmpl.m`.

Working without a database

Instead of working with a database, you can use free raw messages and/or capture messages. These messages are independent of database files. To work without a database, you have to select at least one message on the Free Raw Messages Page (RTICANMM MainBlock) and/or on the Capture Messages Page (RTICANMM MainBlock).

Tip

For an example on working without a database file, refer to Lesson 11: Working Without a Database (RTI CAN MultiMessage Blockset Tutorial 📖).

CAN communication variants

Working with different CAN communication variants on one CAN controller allows you to use the same CAN controller in different CAN buses, and to easily switch between the buses and their respective communication variants.

To work with different CAN communication variants, you have to implement them in the model.
- The RTICANMM ControllerSetup block of the RTI CAN MultiMessage Blockset provides a Variation inport. You can specify a CAN communication variant via this inport. A corresponding entry in the TRC file generated for the model allows you to switch variant during run time.
- You have to specify the CAN communication for each variant in a separate MainBlock. You can specify the corresponding Variation index on this page.

Dialog settings

CAN name Lets you specify the name of the CAN bus. Each CAN bus name can be used only once in an RTI model. If you work with a real-time application of a dSPACE multiprocessor (MP) or multicore (MC) system, the CAN bus name must be unique in the whole application.

Note

The length of the CAN bus name is limited to 20 characters.

Select the ControllerSetup block Lets you select the CAN controller in the model for which you want to set up CAN communication.

CAN FD support Lets you enable or disable CAN FD mode for the RTICANMM MainBlock. If the checkbox is selected, CAN FD messages can be transmitted and received. If the checkbox is cleared, CAN FD messages are not sent/received.

Note

- To ensure that CAN FD messages are properly transmitted and received during run time, CAN FD mode must also be enabled for the selected CAN controller (see Setup Page (RTICANMM ControllerSetup) on page 48).
  If you specified dSPACE hardware that does not support CAN FD communication for the selected CAN controller, or if CAN FD support is not yet activated for the CAN controller in the ControllerSetup block, the **CAN FD support** checkbox for the RTICANMM MainBlock is unavailable and you cannot enable CAN FD mode.
- To monitor CAN FD messages, it is sufficient to enable CAN FD support in the ControllerSetup block.
- Activating CAN FD support can be processing-time-consuming. So you should enable CAN FD support for the MainBlock only if you want to include CAN FD messages in the application.

Variation index Lets you specify the CAN controller variation for which you want to set up CAN communication. Range: 1 … 20.

The RTICANMM MainBlock that corresponds to the selected RTICANMM ControllerSetup block and the selected variation currently is active. You can specify the CAN controller variation during run time via the **Variation** inport of the RTICANMM ControllerSetup block.

Sample time [s] Lets you enter the sample time (the intervals for reading the data).

Sample time	Meaning
- 1	Lets you use the sample time inherited from the model.
> 0	Lets you specify the sample time in milliseconds. Use any multiple of the "Fixed step size" chosen for the model.

Database Lets you specify a DBC, MAT, FIBEX, or AUTOSAR system description file as the database. If you specify a database file other than a MAT file, the **Cluster** field is enabled.

Use relative path Lets you set the folder relative to the model root folder or as an absolute path.

Tip

Using a relative path allows you to easily move the model and all the files belonging to it to another location on your file system without having to change any path settings.

Show Check Report Lets you view a report (HTML file) containing warnings and errors on the selected database file.

- Errors indicate, for example, overlapping signals.
- Warnings indicate, for example, that the database file contains the same value for the minimum and maximum values of a signal, or messages without signals.
- Information indicates, for example, mode-dependent signals.

Note

Messages containing errors will be ignored by the RTI CAN MultiMessage Blockset.

Update Lets you update the settings of the RTICANMM MainBlock according to the database file used.

Cluster (Not enabled if a MAT file is specified as the database) Lets you select the relevant bus for the RTICANMM MainBlock. Since a database file can contain several buses, you have to specify which bus is relevant for the RTICANMM MainBlock.

Import J1939 message with shortest name as a container Lets you select the J1939 container type to be used. When working with a J1939-compliant DBC file, the RTI CAN MultiMessage Blockset distinguishes between container and instance messages. There is one container for each PGN, which receives all the messages (except for messages that are specified on the Instance to Container Page (RTICANMM MainBlock) on page 125) that contain the associated PGN. However, several messages might be defined with the same PGN. Via this setting, you can select to use either the message with the shortest name or an additionally created message as the container for the PGN:

- If the checkbox is selected, the message with the shortest name is used as the container message for a PGN.
- If the checkbox is cleared, an additional container message is created for each PGN during database import. The new message is named `CONT_<shortest_message_name>`. If several messages fulfill the condition of the shortest name, the one that is listed first in the DBC file is used.

Change overlapping signals to message variations Lets you enable changing overlapping signals to message variations. Message variations are used to generate overlapping signals into n variants of a message. Make sure that the variants are defined via MAT file.

To specify several variants of a message, refer to Message Variations Page (RTICANMM MainBlock) on page 198.

Only free raw messages and/or capture Lets you work without a database but with free raw messages or capture messages. You have to select at least one message on the Free Raw Messages Page (RTICANMM MainBlock) and/or on the Capture Messages Page (RTICANMM MainBlock).

This option influences the following pages:

- Capture Messages Page (RTICANMM MainBlock) on page 133
- Network Node Enable Page (RTICANMM MainBlock) on page 117
- Network Node Preselection Page (RTICANMM MainBlock) on page 127
- RX Messages Page (RTICANMM MainBlock) on page 130
- Signal Defaults Page (RTICANMM MainBlock) on page 221
- Signal Errors Page (RTICANMM MainBlock) on page 225
- Signal Ranges Page (RTICANMM MainBlock) on page 223
- Toggle Page (RTICANMM MainBlock) on page 236
- TX Messages Page (RTICANMM MainBlock) on page 128

Related topics

Basics

References

File Settings Page (RTICANMM MainBlock)

Access

Located on the top level of the dialog tree.

Purpose

This page is obsolete. The settings located on this page in earlier RTI CAN MultiMessage Blockset versions are now available on the General Settings Page (RTICANMM MainBlock) on page 100.

Dialog settings	None

Related topics

References

Naming Options Page (RTICANMM MainBlock)

Access

Located on the top level of the dialog tree.

Dialog pages

You can specify the naming options on the following pages:

- Naming Page (RTICANMM MainBlock) on page 108
- Naming Mapping Page (RTICANMM MainBlock) on page 109

Related topics

References

Naming Page (RTICANMM MainBlock)

Access

Located in Naming Options Page (RTICANMM MainBlock) on page 108.

Purpose

To specify the naming of TRC file attributes.

Description

You can influence how dynamic attributes of your model such as message names and signal names will be named in the generated variable description (TRC) file.

You can also use macros to name these attributes. In this case, these macros will be replaced by their actual values during the generation of the TRC file. The following macros are available depending on the selected attribute:

- `%CAN`
- `%Controller`
- `%ECU`

- `%MsgName`
- `%MsgID`
- `%MessageDesc`
- `%SigName`
- `%SignalDesc`

Note

You cannot use macros for some names such as the CAN bus name.

For more information on working with generated TRC variables, refer to Functions with Activated TRC Variable Generation on page 303.

Dialog settings

Function naming list Lists the various functions with their names.

Name Lets you edit the name of the function selected from the Function naming list. Click Set to assign the name to the selected function.

Description Lets you edit a description for the function selected from the Function naming list. Click Set to assign the description to the selected function.

Set Lets you assign the specified name and description to the function selected from the Function naming list.

Macros Lets you open a list of the macros that you can use for the function selected from the function naming list.

Related topics

Basics

References

Naming Mapping Page (RTICANMM MainBlock)

Access

Located in Naming Options Page (RTICANMM MainBlock) on page 108.

Purpose

To specify the naming of specific strings in TRC files.

Description

The Naming Page (RTICANMM MainBlock) on page 108 allows you to influence how dynamic attributes of your model such as message names and signal names will be named in the generated variable description (TRC) file.

The Naming Mapping page allows you to rename specific strings before the TRC file is generated. The settings of this page are applied *after* the settings of the Naming Page (RTICANMM MainBlock) on page 108.

Dialog settings

String to replace Lets you enter a string to be renamed.

New string Lets you enter the new string. This string replaces the old string.

Macro Lets you select macros to specify which attributes are effected by the new string.

- `%CAN`
- `%Controller`
- `%ECU`
- `%MsgName`
- `%MsgID`
- `%MessageDesc`
- `%SigName`
- `%SignalDesc`

Set Lets you add the string to the list of Strings to be replaced.

Delete Lets you delete the string selected from the list of Strings to be replaced.

Strings to be replaced Displays all the available mapping items. You can select mapping item(s) to change already specified settings and sort the items by String to replace, New string or Macro via context menu.

Tip

You can use **`Shift`** or **`Ctrl`** for multiselection.

Related topics

References

Network Nodes Page (RTICANMM MainBlock)

Access	Located on the top level of the dialog tree.
Dialog pages	You can specify the settings specific to network nodes on the following pages: ▪ J1939 Network Node Configuration Page (RTICANMM MainBlock) on page 111 ▪ Network Node Interfaces Page (RTICANMM MainBlock) on page 117
Related topics	References RTICANMM MainBlock.. 68

J1939 Network Node Configuration Page (RTICANMM MainBlock)

Access	Located in Network Nodes Page (RTICANMM MainBlock) on page 111.
Dialog pages	You can specify the settings specific to J1939 network node configuration on the following pages: ▪ Network Node Selection Page (RTICANMM MainBlock) on page 111 ▪ Network Node Identification Page (RTICANMM MainBlock) on page 113
Related topics	References RTICANMM MainBlock.. 68

Network Node Selection Page (RTICANMM MainBlock)

Access	Located in J1939 Network Node Configuration Page (RTICANMM MainBlock) on page 111.

Purpose

To specify which network nodes are to be simulated by the RTI CAN MultiMessage Blockset and which are external network nodes whose network management has to be observed, and to specify the network node indices.

Description

All the network nodes defined in the J1939 database file are displayed with their network node indices, network node names and simulation modes. The list is sorted alphabetically according to the network node names.

The *network node index* is a numerical value assigned to a network node. It is used to address a network node as a source or destination node for J1939 messages. The network node index must be unique. By default, the network node indices are automatically assigned to the network nodes in the order in which they appear in the database file. You can change the network node index by hand, for example, if you want to use existing tests with special addressing.

The *network node names* are specified in the database file.

The *simulation type* of a network node specifies whether the network node is to be completely simulated by the RTI CAN MultiMessage Blockset, or is an external network node whose network management you want to observe and you want to communicate with, or is not relevant.

Dialog settings

Set simulation type of network nodes Lets you select network nodes to specify their node indices and/or simulation types.

> **Tip**
>
> Press the **`Shift`** or **`Ctrl`** key to select several network nodes simultaneously.

Node index Lets you specify the node index for the selected network node in the range 0 ... 253. The network node index must be unique. The node index must be specified individually for each network node. You can therefore edit the node index for only one network node at a time. Your node index setting is ignored if several network nodes are selected in the list.

Simulation type Lets you specify the simulation type for the selected network nodes.

Simulation Type	Description
Simulated	A *simulated* network node is completely simulated by the RTI CAN MultiMessage Blockset. RTICANMM performs network management for this network node, that is, it performs the network address claim procedure for the network node and sends/receives messages for the node. You can switch the simulation of each simulated network node on and off during run time without recompiling the application.

Simulation Type	Description
External	An *external* network node is a real network node that is externally connected to the J1939 network. The network node is not simulated by the RTI CAN MultiMessage Blockset, but RTICANMM detects the network address of the network node and can exchange J1939 messages with it.
Non-relevant	A network node with *non-relevant* simulation type is neither really existent in the J1939 network nor simulated by the RTI CAN MultiMessage Blockset. However, RTICANMM detects the network address of the network node and can exchange J1939 messages with it.

Set Lets you assign the specified settings to the selected network node(s).

Related topics

References

Network Node Identification Page (RTICANMM MainBlock)

Access

Located in J1939 Network Node Configuration Page (RTICANMM MainBlock) on page 111.

Purpose

To specify the node addresses and 64-bit names of network nodes for identification purposes.

Description

J1939 network management is based on the address claiming process. Address claiming means that addresses are automatically assigned to network nodes during network initialization. This dynamic node address assignment ensures unique identification of network nodes and their primary functions. Each network node in a J1939 network must therefore hold one name and one associated address for identification purposes.

Address The 8-bit network node *address* defines the source or destination for J1939 messages in the network. The address of a network node must be unique. If there is an address conflict, the network nodes try to perform dynamic network node addressing (address claiming) to ensure unique addresses, if this is enabled for the network nodes.

The J1939 standard reserves the following addresses:

- Address 0xFE (254) is reserved as the 'null address' that is used as the source address by network nodes that have not yet claimed an address or have failed to claim an address.

- Address 0xFF (255) is reserved as the 'global address' and is exclusively used as a destination address in order to support message broadcasting (for example, for address claims).

The RTI CAN MultiMessage Blockset does not allow J1939 messages to be sent from the null or global addresses.

Note

The RTI CAN MultiMessage Blockset interprets attributes in the DBC file like this:

- In a DBC file created with CANalyzer 5.1 or earlier, the *name* network node attributes and the *J1939PGSrc* and *J1939PGDest* message attributes are read in. The J1939PGSrc attribute is interpreted as the address of the node that sends the message, the J1939PGDest attribute is interpreted as the address of the node that receives the message.
- In a DBC file created with CANalyzer 5.2 or later, the *name* and *NMStationAddress* network node attributes are read in. The NMStationAddress attribute is interpreted as the network node address.

Name The J1939 standard defines a 64-bit *name* to identify each network node. The name indicates the main function of the network node with the associated address and provides information on the manufacturer.

Arbitrary Address Capable	Industry Group	Vehicle System Instance	Vehicle System	Reserved	Function	Function Instance	ECU Instance	Manufacturer Code	Identity Number
1 bit	3 bit	4 bit	7 bit	1 bit	8 bit	5 bit	3 bit	11 bit	21 bit

For more information on the address claiming process, refer to Basics on Working with a J1939-Compliant DBC File on page 30.

Dialog settings

Settings from database Lets you specify to use the network node address and 64-bit name values specified in the database file.

User-defined settings Lets you specify your own network node identification values. Select this option to enable the network node list and all the other settings on this page. The list shows all the network nodes defined in the J1939 database file with their network node indices and simulation types (see Network Node Selection Page (RTICANMM MainBlock) on page 111), network node names (specified in the database file), and address and 64-bit name values. The list is sorted alphabetically according to network node names, grouped by the simulation type. Select a network node to define its address and 64-bit name for identification purposes. The following commands are available via context menu:

- Get all defaults from database
- Get selected defaults from database
- Change all 64-bit names based on node index (unique)

Tip

User-defined network node identification settings are preserved when the database file is updated.

Get all defaults from database (Available via context menu) Lets you set the addresses and 64-bit name values for all network node data fields according to the specification in the database file.

Get selected defaults from database (Available via context menu) Lets you set the addresses and 64-bit name values in the network node data field selected from the list according to the specification in the database file.

Change all 64-bit names based on node index (unique) (Available via context menu) Lets you set unique 64-bit name values automatically for all network nodes. This is useful, for example, if the database file does not contain 64-bit names for all network nodes and the configuration has to be done very quickly.

Note

Previously specified 64-bit name values are overwritten.

Initial network address Displays the initial J1939 network address for the selected network node or lets you assign another free J1939 network address to it. You can enter the new address as a decimal or hexadecimal value in the range 0 ... 253 (0x00 ... 0xFD).

Arbitrary address capable (1 bit) Lets you specify whether the selected network node is arbitrary-address-capable (value is 1) or not (value is 0). An arbitrary-address-capable network node is a self-configurable address network node which can dynamically change its network address. If an arbitrary-address-capable network node encounters an address conflict, the network node with the 64-bit name of higher priority (= lower numerical value) claims the address, and the other network node claims another free address.

Industry group (3 bit) Lets you select the industry group code for the selected network node. Industry group codes are associated with particular industries, for example, On-Highway Equipment or Agricultural and Forestry Equipment. The possible industry group codes are defined and assigned in the SAE J1939 standard.

Vehicle system (7 bit) Lets you select the vehicle system for the selected network node. The vehicle system provides a common name for a group of functions within a connected network. Vehicle systems are closely associated with industry groups, for example, "tractor" in the Global industry group and "trailer" in the On-Highway Equipment industry group. The vehicle systems are defined and assigned in the SAE J1939 standard.

Vehicle system instance (4 bit) Lets you specify the vehicle system instance for the selected network node. The vehicle system instance works in combination with the vehicle system field. A J1939 network can contain several network nodes of the same vehicle system. The vehicle system instance value is used to identify a particular instance of a vehicle system within the network. It assigns a numerical value to each instance of the vehicle system.

Function (8 bit) Lets you specify the function code for the selected network node. The function codes are defined and assigned in the SAE J1939 standard. Function codes in the range 0 ... 127 are independent of other field entries, and the interpretation of codes ≥ 128 depends on the associated industry group entries. (For example, function code 133 means "Product Flow" in combination with the "Agricultural and Forestry Equipment" industry group, but "Land Leveling System Display" in connection with the "Construction Equipment" industry group.)

Function instance (5 bit) Lets you specify the function instance value for the selected network node. The function instance works in combination with the function field. A J1939 network can contain several network nodes (controller applications) with the same function. The function instance value is used to identify a particular instance of a function on a vehicle system within the network.

ECU instance (3 bit) Lets you specify the ECU instance value for the selected network node. A J1939 network can contain several network nodes (controller applications) with the same function. (For example, a single vehicle can contain two identical ECUs, the first measuring the road speed of the vehicle and the second measuring the speed of the attached trailer.) The ECU instance value is used to identify a particular ECU instance of a vehicle system within the network, i.e., the ECU instance indicates which particular network node of the group of network nodes associated with the same function is referenced.

Manufacturer code (11 bit) Lets you specify the manufacturer code of the selected network node. The manufacturer code indicates which manufacturer produced this particular network node. The manufacturer codes are defined in the SAE J1939 document.

Identity number (21 bit) Lets you specify the identity number for the selected network node as a decimal or hexadecimal value. The identity number is assigned by the ECU manufacturer. This field should be unique to guarantee unique names within a product line. The manufacturer is free to encode any other information in the identity number, for example, the serial number or date of manufacture. The identity number must be in the range 0 ... 2,097,151 (0x000000 ... 0x1FFFFF).

64 bit name Displays the 64-bit name of the selected network node consisting of the single name fields (see above), or lets you enter a 64-bit name as a decimal or hexadecimal value. If you enter a new 64-bit name value, the entries of the single name fields are changed accordingly.

Set Lets you assign the specified settings to the selected network node.

Info Displays the network node address and 64-bit name settings made for the currently selected network node. The individual fields of the 64-bit name are listed separately, using the codes and definitions assigned by the SAE and displaying the specified information in plain text, where possible.

Related topics

Basics

References

Network Node Interfaces Page (RTICANMM MainBlock)

Access

Located in Network Nodes Page (RTICANMM MainBlock) on page 111.

Dialog pages

You can specify the settings specific for enabling/disabling network nodes on the following page:

- Network Node Enable Page (RTICANMM MainBlock) on page 117

Related topics

References

Network Node Enable Page (RTICANMM MainBlock)

Access

Located in Network Node Interfaces Page (RTICANMM MainBlock) on page 117.

Purpose

To set the network node-specific conditions for the transmission of messages for particular network nodes.

The network node enable interface allows you to enable or disable the transmission of all the messages transmitted by a network node via a single inport or TRC variable.

Description

You can select one or more network nodes from a list of all the available network nodes and specify their enable conditions for message transmission. You can enable or disable the transmission of all messages of a network node by default. You can let entries be included in the TRC file (to modify the default settings via ControlDesk), and/or generate a Triggering inport for the block (to modify the default settings via your model).

The following table shows which default values are set for the TRC file entries and for the block's Triggering inport when modification via model *and* via ControlDesk is selected:

Specified Dialog Settings				**Resulting Default Value for ...**	
Source	**Logic**	**Default Source**	**Default**	**TRC File Entry**	**Constant Inport**
BOTH	AND	INPORT	DISABLE	1	0
			ENABLE	1	1
		TRC	DISABLE	0	1
			ENABLE	1	1
		BOTH	DISABLE	0	0
			ENABLE	1	1
	OR	INPORT	DISABLE	0	0
			ENABLE	0	1
		TRC	DISABLE	0	0
			ENABLE	1	0
		BOTH	DISABLE	0	0
			ENABLE	1	1

For a better understanding of the trigger conditions for message transmission, refer to the illustration on the Triggering Options Page (RTICANMM MainBlock) on page 136.

This page is disabled if you selected Only free raw messages and/or capture on page 107.

Dialog settings

Specify enables for all network nodes Lets you select network nodes to specify their operation conditions for message transmission. The network node list is disabled if Use these options for all is selected. The following commands are available via context menu:

- Select according to selected Network Node Selection (simulated)
- Select according to selected Network Node Selection (external)
- Select according to selected Network Node Selection (non-relevant)
- Configuration File

Tip

Press the **`Shift`** or **`Ctrl`** key to select several network nodes simultaneously.

Select according to selected Network Node Selection (simulated) (Available via context menu) Lets you select all the network nodes of the *simulated* simulation type. You can specify the simulation type on the Network Node Selection Page (RTICANMM MainBlock) on page 111.

Select according to selected Network Node Selection (external) (Available via context menu) Lets you select all the network nodes of the *external* simulation type. You can specify the simulation type on the Network Node Selection Page (RTICANMM MainBlock) on page 111.

Select according to selected Network Node Selection (non-relevant) (Available via context menu) Lets you select all the network nodes of the *non-relevant* simulation type. You can specify the simulation type on the Network Node Selection Page (RTICANMM MainBlock) on page 111.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Source Lets you specify the condition source. You can let an entry be included in the TRC file and/or generate an Enable inport for the block as the condition source.

Setting	Description
None	Neither a TRC file entry nor an Enable inport is created for messages of the selected network node(s). This is the default value.
TRC	One entry is included in the TRC file for each selected network node. The entry lets you enable or disable message transmission. No Enable inport is generated. If an inport already exists, its structure remains unchanged.
Inport	An Enable inport is generated if the inport does not yet exist. If you select network node-specific triggering, this is added to the inport structure. See I/O characteristics on page 70. No TRC file entry is generated.
Both	For each selected network node, one entry is included in the TRC file and an Enable inport is generated (if the inport does not yet exist). If you select network node-specific triggering, this is added to the inport structure. See I/O characteristics on page 70. You can evaluate this option by a logical AND or OR.

Setting	Description
*	The source value of the selected network node(s) remains unchanged. This allows you to change other options without changing this one. If **Use these options for all** is enabled, this option is not available.

Logic Lets you select how the **Source** values are combined logically if **Both** is selected.

Settings	Description
AND	The source value is evaluated by an **AND** operation.
OR	The source value is evaluated by an **OR** operation. This is the default value.
*	The logic value of the selected network node(s) remains unchanged. This allows you to change other options without changing this one. If **Use these options for all** is enabled, this option is not available.

Use these options for all Select the checkbox to apply the specified settings to all network nodes. If the checkbox is selected, **Set** and **Specify enables for all network nodes** are disabled and * is not available for the drop-down lists. The checkbox is selected by default.

Default source Lets you select the condition source to specify a **Default** enable condition for. The available options are determined by the **Source** value.

Setting	Description
TRC	The **Default** value is specified only for the TRC file, the **Enable** inport remains unchanged.
Inport	The **Default** value is specified only for the **Enable** inport, the TRC file remains unchanged.
Both	The **Default** value is specified for both the TRC file and the **Enable** inport.
*	The default source value of the selected network node(s) remains unchanged. This allows you to change other options without changing this one. If **Use these options for all** is enabled, this option is not available.

Default Lets you specify the default value for the enable condition for the selected **Default source**.

Settings	Description
Enable	Message transmission for the selected network node(s) is enabled by default for the specified **Default source**.
Disable	Message transmission for the selected network node(s) is disabled by default for the specified **Default source**.
*	The default value of the selected network node(s) remains unchanged. This allows you to change other conditions without changing this one. If **Use these options for all** is enabled, this option is not available.

For information on which default values are set if **Both** is specified as the condition source, refer to Description on page 117.

Set Lets you assign the specified conditions to the selected network node(s). This button is disabled if **Use these options for all** is selected.

Related topics

References

RTICANMM MainBlock.. 68
Triggering Options Page (RTICANMM MainBlock).. 136

Messages Page (RTICANMM MainBlock)

Access

Located on the top level of the dialog tree.

Dialog pages

You can specify the settings specific to TX and RX messages on the following pages:

- J1939 Page (RTICANMM MainBlock) on page 122
- Choose Messages Page (RTICANMM MainBlock) on page 127
- Free Raw Messages Page (RTICANMM MainBlock) on page 131
- Capture Messages Page (RTICANMM MainBlock) on page 133
- Triggering Page (RTICANMM MainBlock) on page 136
- Ports & Displays Page (RTICANMM MainBlock) on page 171
- Message Manipulation Page (RTICANMM MainBlock) on page 190

Related topics

References

RTICANMM MainBlock.. 68

J1939 Page (RTICANMM MainBlock)

Access

Located in Messages Page (RTICANMM MainBlock) on page 121.

Dialog page

You can specify the settings specific to J1939 communication on the following page:

- Source Destination Mapping Page (RTICANMM MainBlock) on page 122
- Instance to Container Page (RTICANMM MainBlock) on page 125

Related topics

References

Source Destination Mapping Page (RTICANMM MainBlock)

Access

Located in J1939 Page (RTICANMM MainBlock) on page 122.

Purpose

To specify the mapping of source network node and destination network node for J1939 messages.

Description

Depending on its PGN, a message defined in a J1939-compliant DBC file can be either a broadcast message or a peer-to-peer message (see Basics on Working with a J1939-Compliant DBC File on page 30). In a CAN-J1939 network, a broadcast message is transmitted to any network node connected to the network. A peer-to-peer message is transmitted only to one destination network node.

For each message defined in a J1939-compliant DBC file, the **Source Destination Mapping** page allows you to specify the:

- Source (transmitting network node)
- Destination (receiving network node) for peer-to-peer messages

Without source/destination mapping, if two or more network nodes sent the same PGN in the same sampling step, these messages would be placed in a common container message. As a result only the J1939 message received last in that sampling step is accessible via model and/or TRC file. With source/destination mapping, the two J1939 messages can be treated individually as separate instances of the message. Additionally, you can define instances to be shown in the container. In this case, the container contains the last received

J1939 instances. Refer to Instance to Container Page (RTICANMM MainBlock) on page 125.

The name of a user-defined instance is generated automatically and has the syntax `<container name>_<source node>_<destination node>` (destination node only for peer-to-peer messages). J1939 peer-to-peer messages can use the global address 255 (0xFF) as the destination. In this case, the name of the peer-to-peer message does not include the destination node and matches the name of a broadcast message.

Tip

There exists one container for each PGN (except for proprietary PGNs). If you work with a J1939-compliant DBC file created with CANalyzer 5.2 or later, several messages with the same PGN might be defined. The log file provides information on which instance is used as the container message.

For more information on how the RTI CAN MultiMessage Blockset determines a container, refer to Basics on Working with a J1939-Compliant DBC File on page 30.

Mapping in the DBC file The source/destination mapping can be defined in the DBC file. However, the network node information in the DBC file might be incomplete or erroneous. To define the mapping for a message in the DBC file, add the RTICANMM-specific `J1939Mapping` attribute to the message, specifying the desired mapping string. The syntax of the mapping string is as follows:

- The individual mappings are separated by semicolons.
- For peer-to-peer messages, the source and destination network nodes are separated by a hyphen.

For instructions on modifying DBC files, refer to the documentation of the DBC++ editor.

Example In the mapping string, the source and destination network nodes are described by their network node names.

- The mapping string for a broadcast message could look like this: `CSS;CTS;CUV`
- The mapping string for a peer-to-peer message could look like this: `APS-ATA;Door3-DualBatt`

Dialog settings

Source/destination mapping Displays the source/destination mappings you defined so far. Only the user-defined source/destination mappings are displayed, not mappings that are imported from the DBC file. The following commands are available via context menu:

- Delete mappings (current selected)
- Delete all mappings
- Filter
- Sort
- Configuration File

Delete mappings (current selected) (Available via context menu) Lets you remove the currently selected source/destination mappings from the mapping list.

Delete all mappings (Available via context menu) Lets you remove all source/destination mappings from the mapping list. As a consequence, all user-defined mappings are deleted. Mappings that are imported from the DBC file are not shown in the mapping list and are not affected by this option.

Filter (Available via context menu) Lets you filter and select PGNs in the mapping list using the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort - Sort PGN (Available via context menu) Lets you sort the mapping list alphabetically according to PGNs.

Sort - Sort Source (Available via context menu) Lets you sort the mapping list alphabetically according to source network node names.

Sort - Sort Destination (Available via context menu) Lets you sort the mapping list alphabetically according to destination network node names.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

PGN Lets you specify a message whose source/destination mapping you want to configure. The message is specified by its associated PGN. The list contains the containers of all the PGNs contained in the J1939-compliant DBC file.

Source Lets you select the network node to be added as the source (transmitting network node) to the message whose PGN is currently selected from the PGN drop-down list. All network nodes defined in the DBC file are available.

Destination (Available only if the PGN selected from the PGN drop-down list belongs to a peer-to-peer message) Lets you select the network node to be added as the destination (receiving network node) to the message whose PGN is currently selected from the PGN drop-down list. All network nodes defined in the DBC file are available.

Add Lets you add the source/destination mapping specified by the selected PGN, Source and Destination values to the mapping list.

Set (Available only if a single source/destination mapping is selected in the mapping list) Lets you replace the source network node and destination network node (only for peer-to-peer message) of the currently selected source/destination mapping with the values currently selected from the Source and Destination drop-down lists.

Keep in mind that previously made settings for an instance (for example, counter, parity, trigger and cycle time settings) are lost when a source/destination mapping is reconfigured. In the RTI CAN MultiMessage Blockset, the configuration is connected to the name of an instance, and the name changes if the source or destination node is replaced.

Delete Lets you delete the currently selected mappings from the mapping list.

Related topics

Basics

References

Instance to Container Page (RTICANMM MainBlock)

Access

Located in J1939 Page (RTICANMM MainBlock) on page 122.

Purpose

To define an instance to be shown in the container.

Description

For J1939 messages, the source/destination mapping is already specified in the DBC file or you can specify it on the Source Destination Mapping Page (RTICANMM MainBlock) on page 122.

With RTICANMM, a J1939 message is received as follows: If the configuration contains an RX instance with matching PGN, source node and destination node, the message is received in that instance. If the configuration does not contain such an instance, the message is received in a matching container (if existent).

The RTI CAN MultiMessage Blockset lets you specify to place an instance in its associated J1939 container.

Tip

There exists one container for each PGN (except for proprietary PGNs). If you work with a J1939-compliant DBC file created with CANalyzer 5.2 or later, several messages with the same PGN might be defined. The log file provides information on which instance is used as the container message. For more information on how the RTI CAN MultiMessage Blockset determines a container, refer to Basics on Working with a J1939-Compliant DBC File on page 30.

You can disable the instance to container option during run time. This can be useful, for example, if you want to connect a large number of messages and are interested in the last parameter group number (PGN) only, or if the communication design guarantees that two messages with the same PGN are definitely not received in the same sampling step.

Dialog settings

Define instance to be shown in container Lets you select instances of mapped messages to specify their container conditions for transmission. All messages that are currently selected on the RX Messages Page (RTICANMM MainBlock) on page 130 are displayed in the list.

> **Tip**
>
> Press the **`Shift`** or **`Ctrl`** key to select several instances simultaneously.

You can create or load a configuration file or add these page settings to an existing configuration file via context menu.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Active Lets you specify whether an instance is placed in the container or not.

Setting	Description
Active	The message is still mapped, and the instance is placed in the container. You can change this option during run time.
Inactive	The message is still mapped, but the instance is not placed in the container. If you select Inactive, the Default options are not available.
*	The default behavior remains unchanged. This allows you to change other options without changing this one.

Default Lets you specify the default behavior for the instance to container option.

Setting	Description
Enable	Instance to container is enabled by default. You can disable the option during run time.
Disable	Instance to container is disabled by default.
*	The default behavior of the selected instance(s) remains unchanged. This allows you to change other conditions without changing this one.

Set Lets you assign the specified conditions to the selected instance(s).

Use these options for all Select the checkbox to apply the specified settings to all instances. If the checkbox is selected, Set and Define instance to be shown in container are disabled. The checkbox is selected by default.

Related topics

Basics

References

Choose Messages Page (RTICANMM MainBlock)

Access

Located in Messages Page (RTICANMM MainBlock) on page 121.

Dialog pages

You can specify the messages to be transmitted or received on the following pages:

- Network Node Preselection Page (RTICANMM MainBlock) on page 127
- TX Messages Page (RTICANMM MainBlock) on page 128
- RX Messages Page (RTICANMM MainBlock) on page 130

Related topics

References

Network Node Preselection Page (RTICANMM MainBlock)

Access

Located in Choose Messages Page (RTICANMM MainBlock) on page 127.

Purpose

To select a set of network nodes for further message setup.

Description

This page lets you make a selection of the network nodes referenced in the DBC file. The network nodes you select on this page can be used in the context menu on the TX Messages Page (RTICANMM MainBlock) and RX Messages Page (RTICANMM MainBlock), for example. This allows you to specify message settings for specific network nodes.

This page is disabled if you selected Only free raw messages and/or capture on page 107.

Dialog settings

All Lets you select all the network nodes displayed.

Selected Lets you specify one or more network nodes.

Available Network Nodes Displays all the network nodes that are available in the DBC file. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select network nodes in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

Selected Network Nodes Displays the selection of network nodes that is active if you select the Selected option. The following commands are available via context menu:

- All
- None

All (Available via context menu) Lets you add all the available network nodes to the list.

None (Available via context menu) Lets you remove all network nodes from the list.

Related topics

References

TX Messages Page (RTICANMM MainBlock)

Access

Located in Choose Messages Page (RTICANMM MainBlock) on page 127.

Purpose

To specify the messages to be transmitted by the CAN controller.

Description

The **Available messages** list of this page lists all the messages contained in the database file. If you work with a J1939-compliant DBC file, all the instances found in the database and defined by the source/destination mappings are listed. You can select any of these messages for transmission. This page is disabled if you selected Only free raw messages and/or capture on page 107.

Dialog settings

All Lets you specify all messages as TX messages.

None Lets you select none of the available messages for transmission.

Selected Lets you specify a set of the available messages for transmission.

Available messages Displays all the messages that are available in the database file. The following commands are available via context menu:

- Filter
- Sort
- J1939

Filter (Available via context menu) Lets you filter and select messages in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically **Up** or **Down**.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Selected messages Displays the selection of messages that is active if you select the **Selected** option. You can select specific messages via context menu:

- All
- None
- Choose the Same Messages as RX
- Choose Messages Not Chosen as RX
- Choose Messages Received by Selected Network Nodes
- Choose Messages Not Received by Selected Network Nodes
- Choose Messages Transmitted by Selected Network Nodes
- Choose Messages Not Transmitted by Selected Network Nodes
- Choose Messages Not Transmitted by Any Network Node
- Choose Messages Not Received by Any Network Node
- Choose Messages Received but Not Transmitted by Selected Network Nodes

You can specify the *Selected Network Nodes* on the Network Node Preselection Page (RTICANMM MainBlock) on page 127.

Related topics

References

RX Messages Page (RTICANMM MainBlock)

Access

Located in Choose Messages Page (RTICANMM MainBlock) on page 127.

Purpose

To specify the messages to be received by the CAN controller.

Description

The **Available messages** list of this page lists all the messages contained in the database file. You can select any of these messages for reception. This page is disabled if you selected Only free raw messages and/or capture on page 107.

Loopback messages: Each message you specify to be transmitted by the CAN controller on the TX Messages Page (RTICANMM MainBlock) on page 128 will also automatically be received by the same CAN controller. Since these messages are transmitted and then received by the same CAN controller, they are called *loopback messages*.

Dialog settings

All Lets you specify all messages as RX messages.

None (just loopback) Lets you specify that none of the available messages are received. Only loopback messages will be received.

Selected Lets you specify a set of the available messages for reception.

Available messages Displays all the messages that are available in the database file. The following commands are available via context menu:

- Filter
- Sort
- J1939

Filter (Available via context menu) Lets you filter and select messages in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Selected messages Displays the selection of messages that is active if you select the **Selected** option. You can select specific messages via context menu:

- All
- None
- Choose the Same Messages as TX
- Choose Messages Not Chosen as TX
- Choose Messages Received by Selected Network Nodes
- Choose Messages Not Received by Selected Network Nodes
- Choose Messages Transmitted by Selected Network Nodes
- Choose Messages Not Transmitted by Selected Network Nodes
- Choose Messages Not Transmitted by Any Network Node
- Choose Messages Not Received by Any Network Node

You can specify the *Selected Network Nodes* on the Network Node Preselection Page (RTICANMM MainBlock) on page 127.

Related topics

References

Free Raw Messages Page (RTICANMM MainBlock)

Access

Located in Messages Page (RTICANMM MainBlock) on page 121.

Purpose

To process messages independently of the database file.

Description

Free raw messages are independent of the database file. They always consist of 8 (CAN) or up to 64 (CAN FD) 8-bit signals. You can completely modify these messages, including their identifiers, message formats and message lengths. They are TX and RX messages. However, if you change an ID during run time, make sure that no other messages (also no other free raw messages) use this ID at the same time.

When you use free raw messages, you might need to modify IDs, message formats and message lengths during run time. For details, refer to TX ID Page (RTICANMM MainBlock) on page 191 and TX Message Length Page (RTICANMM MainBlock) on page 195.

When you work without a database file, you have to select at least one free raw message and/or one capture message. For details, refer to General Settings Page (RTICANMM MainBlock) on page 100 and Capture Messages Page (RTICANMM MainBlock) on page 133.

Dialog settings

Number of free raw messages Lets you specify how many free raw messages you want to use and adds them to the list of **Free raw messages**.

Free raw messages Displays all the messages that you specified as free raw messages. You can display a list of the IDs already specified in the database file via context menu (**Show IDs in database**).

Default identifier Lets you specify the default identifier for the message selected from the list of **Free raw messages**. Make sure that no other messages (also no free raw messages) use this ID.

Format Lets you select the default format for the message selected from the list of **Free raw messages**. The selected format affects the number of raw bytes to be generated for the selected message, and specifies whether the message's data bits are transferred with the standard or the optional higher bit rate (CAN FD only).

The following message formats are available:

Format	Description
STD	Standard format (11-bit identifier) for classic CAN message (8 data bytes are generated, BRS = 0)
EXT	Extended format (29-bit identifier) for classic CAN message (8 data bytes are generated, BRS = 0)
FD STD	Standard format (11-bit identifier) for CAN FD message *without* bit rate switch for data phase (i.e., the standard bit rate is used for the CAN FD data phase) (64 data bytes are generated, BRS = 0)
FD EXT	Extended format (29-bit identifier) for CAN FD message *without* bit rate switch for data phase (i.e., the standard bit rate is used for the CAN FD data phase) (64 data bytes are generated, BRS = 0)
FD STD BRS	Standard format (11-bit identifier) for CAN FD message *with* bit rate switch for data phase (i.e., the optional higher bit rate is used for the CAN FD data phase) (64 data bytes are generated, BRS = 1)
FD EXT BRS	Extended format (29-bit identifier) for CAN FD message *with* bit rate switch for data phase (i.e., the optional higher bit rate is used for the CAN FD data phase) (64 data bytes are generated, BRS = 1)

> **Note**
>
> - There are some points to note concerning the manipulation of message formats and message lengths during run time. Refer to:
>   - TX ID Page (RTICANMM MainBlock) on page 191
>   - TX Message Length Page (RTICANMM MainBlock) on page 195
> - If you work without a database file and if you specify free raw messages with the CAN FD format only, ensure that CAN FD support is enabled for the RTICANMM MainBlock (refer to General Settings Page (RTICANMM MainBlock) on page 100). Otherwise, building an application for your model will fail.

Set Lets you assign the specified default identifier and format to the message selected from the **Free raw messages** list.

Delete Lets you remove the selected message from the **Free raw messages** list.

Related topics

Basics

References

Capture Messages Page (RTICANMM MainBlock)

Access Located in Messages Page (RTICANMM MainBlock) on page 121.

Purpose To capture messages on the CAN bus whose identifiers match the specified filter.

Description

If you work without a database file, you have to select at least one capture message and/or one free raw message.

> **Tip**
>
> For details on working without a database file, refer to General Settings Page (RTICANMM MainBlock) on page 100, Free Raw Messages Page (RTICANMM MainBlock) on page 131 and Lesson 11: Working Without a Database (RTI CAN MultiMessage Blockset Tutorial 📖).

By default, only messages are captured that pass the specified identifier filter and that are not specified on the TX Messages Page (RTICANMM MainBlock) on page 128, RX Messages Page (RTICANMM MainBlock) on page 130, and/or Free Raw Messages Page (RTICANMM MainBlock) on page 131. However, you can enable to also capture messages specified on these pages.

Messages are captured in every sampling step, so the value of a capture message often changes within milliseconds.

Number of messages to be captured You have to specify the number of messages to be captured in one sampling step. If you want to ensure that all the messages of one sampling step are captured, you can calculate the maximum number of messages on the bus:

```
max_message_number = baud_rate [Kb/s] · sample_time [ms] /
message_length [bit]
```

The baud rate and the sample time are specified on the Setup Page (RTICANMM ControllerSetup) on page 48 and in the Solver Dialog (Model Configuration Parameters Dialogs) (RTI and RTI-MP Implementation Reference 📖), respectively. The message length is the sum of data bits, protocol bits, and stuff bits.

> **Tip**
>
> You can calculate a realistic number of stuff bits using a stuff bit probability of 35% of the maximum number of stuff bits.

Example of calculating the relevant message_length:

Suppose you have a classic CAN message of STD identifier format (47 protocol bits) with 64 data bits. For classic CAN messages of STD identifier format, only 34 bits of the protocol bits can cause stuff bits. Thus, the maximum stuff bit number is: (34 protocol bits + 64 data bits) / 5 = 19.6.

With a stuff bit probability of 35%, the relevant message_length value is: message_length = 47 + 64 + (19.6 · 35%).

> **Tip**
>
> For classic CAN messages of EXT identifier format (67 protocol bits), only 54 of the protocol bits can cause stuff bits.

Dialog settings

Number of messages to capture in one sample step Lets you specify how many messages are captured at the most. If more messages match the acceptance filter, they are not captured.

Capture messages included in TX/RX list and also FreeRawMessages Lets you enable to also capture messages specified on the TX Messages Page (RTICANMM MainBlock), RX Messages Page (RTICANMM MainBlock) and Free Raw Messages Page (RTICANMM MainBlock).

Acceptance filter for captured standard identifiers Lets you specify the bits to be filtered if you use standard identifiers. For bits that you do not want to filter, enter `X`.

For example, to filter for messages with ID 34 (i.e., 0x22), enter `00000100010`, to filter for messages with IDs between 32 and 63, enter `000001XXXXX`.

Acceptance filter for captured extended identifiers Lets you specify the bits to be filtered if you use extended identifiers. For bits that you do not want to filter, enter `X`.

Create outport for captured messages Lets you make captured messages available as outports.

Add captured messages to TRC file Lets you make captured messages available in the TRC file.

No RX status for loopback capture Lets you specify the RX status for the reception of loopback capture. The RX status is provided via the RTICANMM MainBlock's RX Data outport if this outport is enabled (see RX Status and Time Ports Page (RTICANMM MainBlock) on page 173).

- If selected, the RX status remains `0` whenever loopback captures are received.
- If not selected, the RX status is set to `1` whenever loopback captures are received.

Related topics

Basics

Lesson 11: Working Without a Database (RTI CAN MultiMessage Blockset Tutorial 📖)

References

Free Raw Messages Page (RTICANMM MainBlock) 131
General Settings Page (RTICANMM MainBlock) 100
RTICANMM MainBlock 68
RX Messages Page (RTICANMM MainBlock) 130
Setup Page (RTICANMM ControllerSetup) 48
Solver Dialog (Model Configuration Parameters Dialogs) (RTI and RTI-MP Implementation Reference 📖)
TX Messages Page (RTICANMM MainBlock) 128

Triggering Page (RTICANMM MainBlock)

Access	Located in Messages Page (RTICANMM MainBlock) on page 121.

Dialog pages

You can specify the various aspects of triggering on the following pages:

- Triggering Options Page (RTICANMM MainBlock) on page 136
- Message Enable Page (RTICANMM MainBlock) on page 140
- Message Cyclic Page (RTICANMM MainBlock) on page 144
- Message Kickout Page (RTICANMM MainBlock) on page 148
- Cycle Times Page (RTICANMM MainBlock) on page 151
- TX Timeout Enable Page (RTICANMM MainBlock) on page 161
- Trigger Reaction Page (RTICANMM MainBlock) on page 163
- Mapping Page (RTICANMM MainBlock) on page 168

Related topics

References

RTICANMM MainBlock.. 68

Triggering Options Page (RTICANMM MainBlock)

Access

Located in Triggering Page (RTICANMM MainBlock) on page 136.

Purpose

To specify the conditions for triggering the transmission of messages.

Description

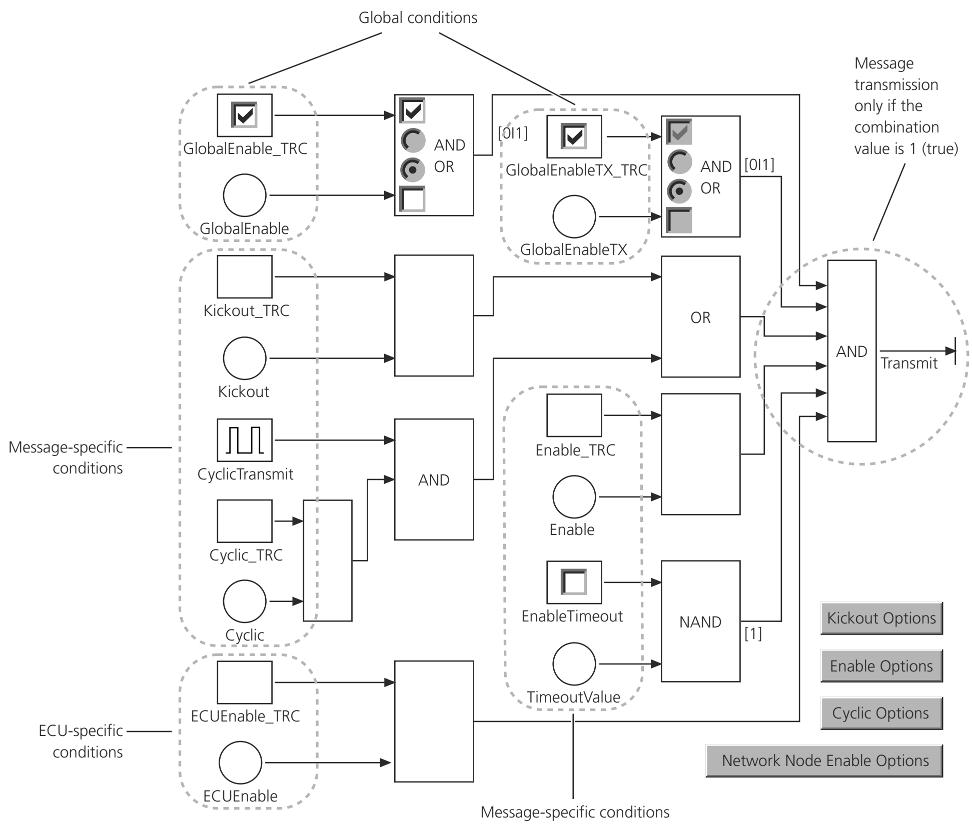
- *Trigger conditions*: You can specify trigger conditions for message transmission of the current CAN controller variation. The following conditions are available:

Type	Condition
Global conditions	GlobalEnable
	GlobalEnableTX
Network node-specific conditions	Network Node Enable Options
Message-specific conditions	Enable Options
	Timeout
	Kickout Options
	Cyclic Options

To trigger message transmission, these conditions are combined logically so that a message is transmitted only if the combination of the condition values is 1 (true).

- *Global and message-specific conditions*: Global conditions let you specify trigger conditions that apply to the current CAN controller variation and cannot be set for individual messages. Message-specific conditions can be set for individual messages.

  The illustration below shows all the options and the way they are logically combined:

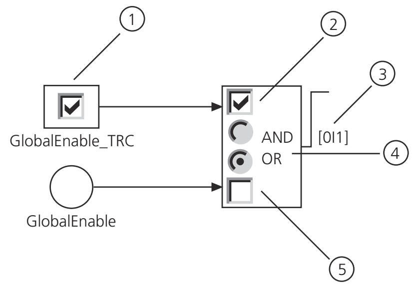


- *TRC file entries and block inports*: For each trigger condition, you can let an entry be included in the TRC file and an inport be generated for the corresponding block.

Dialog settings

GlobalEnable Lets you enable or disable the entire CAN communication of this CAN controller variation. If the CAN communication is disabled, almost no turnaround time is required.

To set the GlobalEnable condition, you can let an entry be included in the TRC file and/or a GlobalEnable inport be generated for a block. If you select both the TRC file entry and the block inport, you can let the condition value be evaluated by an AND or OR operation.

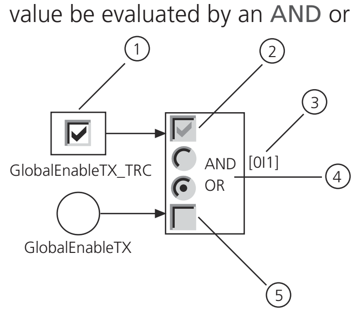


The following settings are possible for this condition:

Setting	Description
1	(Available only if setting 2 is selected) ▪ If selected, the default value is 1. ▪ If not selected, the default value is 0.
2	▪ If selected, an entry specific to this condition will be included in the TRC file. ▪ If not selected, no entry specific to this condition will be included in the TRC file.
3	Displays the possible condition values for this condition. If neither 2 nor 5 is selected, the condition value always is `1`. Otherwise the value can be `0` (false) or `1` (true).
4	(Available only if settings 2 and 5 are selected) Lets you select whether conditions 1 and GlobalEnable are combined via AND or OR.
5	▪ If selected, the current RTICANMM MainBlock gets a GlobalEnable inport. See I/O characteristics on page 70. ▪ If not selected, this block does not get a GlobalEnable inport.

GlobalEnableTX Lets you enable or disable the transmission of messages of this CAN controller variation.

To set the GlobalEnableTX condition, you can let an entry be included in the TRC file and/or a GlobalEnableTX inport be generated for the block. If you select both the TRC file entry and the block inport, you can let the condition value be evaluated by an AND or OR operation.



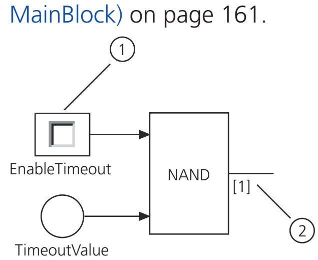
The following settings are possible for this condition:

Setting	Description
1	(Available only if setting 2 is selected) ▪ If selected, the default value is 1.

Setting	Description
	▪ If not selected, the default value is 0.
2	▪ If selected, an entry specific to this condition will be included in the TRC file. ▪ If not selected, no entry specific to this condition will be included in the TRC file.
3	Displays the possible condition values for this condition. If neither 2 nor 5 is selected, the condition value always is `1`. Otherwise the value can be `0` (false) or `1` (true).
4	(Available only if settings 2 and 5 are selected) Lets you select whether conditions 1 and GlobalEnableTX are combined via AND or OR.
5	▪ If selected, the current RTICANMM MainBlock gets a **GlobalEnableTX** inport. See I/O characteristics on page 70. ▪ If not selected, this block does not get a **GlobalEnableTX** inport.

Timeout Lets you enable or disable the timeout mechanism for the transmission of TX messages.

If the timeout mechanism is enabled, entries for the timeout values will be included in the TRC file. For details on the timeout mechanism and on specifying default timeout values, refer to the TX Timeout Enable Page (RTICANMM MainBlock) on page 161.



The following settings are possible for this condition:

Setting	Description
1	▪ If selected, the timeout mechanism for the transmission of TX messages is enabled. ▪ If not selected, the timeout mechanism for the transmission of TX messages is disabled.
2	Displays the possible condition values for this condition. If setting 1 is not selected, the condition value always is `1`. Otherwise the value can be `0` (false) or `1` (true).

Kickout Options Lets you specify the kickout conditions. Kickout lets you transmit messages immediately. The Message Kickout Page (RTICANMM MainBlock) on page 148 opens for you to specify the kickout conditions.

Enable Options Lets you enable or disable the transmission of individual messages of this CAN controller variation. The Message Enable Page (RTICANMM MainBlock) on page 140 opens for you to specify the enable conditions.

Cyclic Options Lets you enable or disable the cyclic transmission of messages of this CAN controller variation. The Message Cyclic Page (RTICANMM MainBlock) on page 144 opens for you to specify the cyclic transmission conditions.

Network Node Enable Options Lets you enable or disable the transmission of messages for particular network nodes. The Network Node Enable Page (RTICANMM MainBlock) on page 117 opens for you to specify the network node-specific conditions.

Related topics

References

Message Enable Page (RTICANMM MainBlock)

Access

Located in Triggering Page (RTICANMM MainBlock) on page 136.

Purpose

To set the message-specific conditions for the transmission of particular messages.

Description

You can select one or more messages and specify the trigger conditions for transmitting them. You can enable or disable message transmission by default. You can let entries be included in the TRC file (to modify the default settings via ControlDesk), and/or generate a Triggering inport for the block (to modify the default settings via your model).

The following table shows which default values are set for the TRC file entries and for the block's Triggering inport when modification via model *and* via ControlDesk is selected:

Specified Dialog Settings				Resulting Default Value for ...	
Source	**Logic**	**Default Source**	**Default**	**TRC File Entry**	**Constant Inport**
BOTH	AND	INPORT	DISABLE	1	0
			ENABLE	1	1
		TRC	DISABLE	0	1
			ENABLE	1	1
		BOTH	DISABLE	0	0
			ENABLE	1	1
	OR	INPORT	DISABLE	0	0
			ENABLE	0	1
		TRC	DISABLE	0	0
			ENABLE	1	0
		BOTH	DISABLE	0	0
			ENABLE	1	1

For a better understanding of the trigger conditions for message transmission, refer to the illustration on the Triggering Options Page (RTICANMM MainBlock) on page 136.

Dialog settings

Specify enables for all messages Lets you select messages from all the available messages to specify their trigger conditions for transmission. The message list is disabled if Use these options for all is selected. The following commands are available via context menu:

- Select according to selected Network Nodes
- J1939
- Configuration File

Tip

Press the **`Shift`** or **`Ctrl`** key to select several messages simultaneously.

Select according to selected Network Nodes - Messages received by Network Nodes (Available via context menu) Lets you select messages that are received by the selected network nodes. You can specify the *selected network nodes* on the Network Node Preselection Page (RTICANMM MainBlock) on page 127.

Select according to selected Network Nodes - Messages transmitted by Network Nodes (Available via context menu) Lets you select messages that are transmitted by the selected network nodes. You can specify the *selected network nodes* on the Network Node Preselection Page (RTICANMM MainBlock) on page 127.

Select according to selected Network Nodes - Messages not received by Network Nodes (Available via context menu) Lets you select messages that are not received by the selected network nodes. You can specify the *selected network nodes* on the Network Node Preselection Page (RTICANMM MainBlock) on page 127.

Select according to selected Network Nodes - Messages not transmitted by Network Nodes (Available via context menu) Lets you select messages that are not transmitted by the selected network nodes. You can specify the *selected network nodes* on the Network Node Preselection Page (RTICANMM MainBlock) on page 127.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Source Lets you specify the condition source. You can let an entry be included in the TRC file and/or generate a Triggering inport for the block as the condition source.

Setting	Description
None	Neither a TRC file entry nor a Triggering inport is generated for the selected message(s).
TRC	One entry is included in the TRC file for each selected TX message. The entry lets you enable or disable message transmission. No Triggering inport is generated. If an inport already exists, its structure remains unchanged. This is the default value.
Inport	A Triggering inport is generated if the inport does not yet exist. The entry lets you enable or disable message transmission. If you select cyclic triggering, this is added to the inport structure. See I/O characteristics on page 70. No TRC file entry is generated.
Both	For each selected TX message, one entry is included in the TRC file and a Triggering inport is generated (if the inport does not yet exist). If you select cyclic triggering, this is added to the inport structure. See I/O characteristics on page 70. You can evaluate this option by a logical AND or OR.

Setting	Description
*	The source value of the selected message(s) remains unchanged. This allows you to change other options without changing this one. If **Use these options for all** is enabled, this option is not available.

Logic Lets you select how the source values are combined logically if **Both** is selected.

Settings	Description
AND	The source is evaluated by an **AND** operation.
OR	The source is evaluated by an **OR** operation. This is the default value.
*	The logic value of the selected message(s) remains unchanged. This allows you to change other options without changing this one. If **Use these options for all** is enabled, this option is not available.

Use these options for all Select the checkbox to apply the specified settings to all messages. If the checkbox is selected, **Set** and **Specify enables for all messages** are disabled and * is not available for the drop-down lists. The checkbox is selected by default.

Default source Lets you select the condition source to specify a **Default** triggering condition for. The available options are determined by the **Source** settings.

Setting	Description
TRC	The **Default** value is specified only for the TRC file, the **Triggering** inport remains unchanged.
Inport	The **Default** value is specified only for the **Triggering** inport, the TRC file remains unchanged.
Both	The **Default** value is specified for both the TRC file and the **Triggering** inport.
*	The default source value of the selected message(s) remains unchanged. This allows you to change other options without changing this one. If **Use these options for all** is enabled, this option is not available.

Default Lets you specify the default value for the triggering condition for the selected **Default source**.

Settings	Description
Enable	Message transmission for the specified **Default source** is enabled by default.
Disable	Message transmission for the specified **Default source** is disabled by default.
*	The default value of the selected message(s) remains unchanged. This allows you to change other conditions without changing this one. If **Use these options for all** is enabled, this option is not available.

For information on which default values are set if Both is specified as the condition source, refer to Description on page 140.

Set Lets you assign the specified conditions to the selected message(s). This button is disabled if Use these options for all is selected.

Related topics

Basics

Lesson 3: Triggering Message Transmission via the Model and ControlDesk (RTI CAN MultiMessage Blockset Tutorial 📖)

References

Message Cyclic Page (RTICANMM MainBlock)

Access

Located in Triggering Page (RTICANMM MainBlock) on page 136.

Purpose

To specify cyclic message transmission.

Description

Messages can be transmitted cyclically or triggered. If you enable the cyclic transmission of messages, the messages are transmitted according to the specified cycle time. If you set the cycle time to zero, the message transmission is stopped. You can specify the default behavior on the Cycle Time Defaults Page (RTICANMM MainBlock). If you enable triggered transmission, the messages are transmitted by kickout. You can specify the kickout conditions on the Message Kickout Page (RTICANMM MainBlock).

On this page, you can select one or more messages and specify cyclic or triggered message transmission by default for them. You can let entries be included in the TRC file (to modify these default settings via ControlDesk), and/or generate a Triggering inport for the block (to modify these default settings via your model). The following table shows which default values are set for the TRC file entries and for the block's Triggering inport when modification via model *and* via ControlDesk is selected:

Specified Dialog Settings				Resulting Default Value for ...	
Source	**Logic**	**Default Source**	**Default**	**TRC File Entry**	**Constant Inport**
BOTH	AND	INPORT	TRIGGERED	1	0
			CYCLIC	1	1
		TRC	TRIGGERED	0	1
			CYCLIC	1	1
		BOTH	TRIGGERED	0	0
			CYCLIC	1	1
	OR	INPORT	TRIGGERED	0	0
			CYCLIC	0	1
		TRC	TRIGGERED	0	0
			CYCLIC	1	0
		BOTH	TRIGGERED	0	0
			CYCLIC	1	1

For a better understanding of the trigger conditions for message transmission, refer to the illustration on the Triggering Options Page (RTICANMM MainBlock) on page 136.

Dialog settings

Specify triggering for all messages Lets you select messages from all the available messages to specify transmission. The message list is disabled if Use these options for all is selected. The following commands are available via context menu:

- Set selected to database defaults
- Select according to selected Network Nodes
- J1939
- Configuration File

> **Tip**
>
> Press the **`Shift`** or **`Ctrl`** key to select several messages simultaneously.

Set selected to database defaults (Available via context menu) Lets you specify the transmission of the selected message(s) according to the specification in the database file.

Select according to selected Network Nodes - Messages received by Network Nodes (Available via context menu) Lets you select messages that are received by the selected network nodes. You can specify the *selected network nodes* on the Network Node Preselection Page (RTICANMM MainBlock) on page 127.

Select according to selected Network Nodes - Messages transmitted by Network Nodes (Available via context menu) Lets you select messages that are transmitted by the selected network nodes. You can specify the *selected network nodes* on the Network Node Preselection Page (RTICANMM MainBlock) on page 127.

Select according to selected Network Nodes - Messages not received by Network Nodes (Available via context menu) Lets you select messages that are not received by the selected network nodes. You can specify the *selected network nodes* on the Network Node Preselection Page (RTICANMM MainBlock) on page 127.

Select according to selected Network Nodes - Messages not transmitted by Network Nodes (Available via context menu) Lets you select messages that are not transmitted by the selected network nodes. You can specify the *selected network nodes* on the Network Node Preselection Page (RTICANMM MainBlock) on page 127.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Source Lets you specify the condition source. You can let an entry be included in the TRC file and/or generate a Triggering inport for the block as the condition source.

Setting	Description
None	Neither a TRC file entry nor a Triggering inport is generated for the selected message(s).
TRC	One entry is included in the TRC file for each selected TX message. The entry lets you enable or disable message transmission. No Triggering inport is generated. If an inport already exists, its structure remains unchanged. This is the default value.
Inport	A Triggering inport is generated if the inport does not yet exist. The entry lets you enable or disable message transmission. If you select cyclic triggering, this is added to the inport structure. See I/O characteristics on page 70. No TRC file entry is generated.
Both	For each selected TX message, one entry is included in the TRC file and a Triggering inport is generated (if the inport does not yet exist). If you select cyclic triggering, this is added to the inport structure. See

Setting	Description
	I/O characteristics on page 70. You can evaluate this option by a logical AND or OR.
*	The source value of the selected message(s) remains unchanged. This allows you to change other options without changing this one. If Use these options for all is enabled, this option is not available.

Logic Lets you select how the source values are combined logically if Both is selected.

Settings	Description
AND	The source is evaluated by an AND operation.
OR	The source is evaluated by an OR operation. This is the default value.
*	The logic value of the selected message(s) remains unchanged. This allows you to change other options without changing this one. If Use these options for all is enabled, this option is not available.

Use these options for all Select the checkbox to apply the specified settings to all messages. If the checkbox is selected, Set and Specify triggering for all messages are disabled and * is not available for the drop-down lists. The checkbox is selected by default.

Default source Lets you select the condition source to specify a Default triggering condition for. The available options are determined by the Source settings.

Setting	Description
TRC	The Default value is specified only for the TRC file, the Triggering inport remains unchanged.
Inport	The Default value is specified only for the Triggering inport, the TRC file remains unchanged.
Both	The Default value is specified for both the TRC file and the Triggering inport.
*	The default source value of the selected message(s) remains unchanged. This allows you to change other options without changing this one. If Use these options for all is enabled, this option is not available.

Default Lets you specify the default value for the triggering condition for the selected Default source.

Settings	Description
Cyclic	Message transmission for the specified Default source is cyclic by default.
Triggered	Message transmission for the specified Default source is triggered by default.

Settings	Description
*	The default value of the selected message(s) remains unchanged. This allows you to change other conditions without changing this one. If **Use these options for all** is enabled, this option is not available.

For information on which default values are set for the TRC file entries and for the inport if **Both** is specified as the condition source, refer to Description on page 144.

Set Lets you assign the specified conditions to the selected message(s). This button is disabled if **Use these options for all** is selected.

Related topics

Basics

Lesson 3: Triggering Message Transmission via the Model and ControlDesk (RTI CAN MultiMessage Blockset Tutorial 📖)

References

Message Kickout Page (RTICANMM MainBlock)

Access

Located in Triggering Page (RTICANMM MainBlock) on page 136.

Purpose

To transmit single messages via kickout.

Description

You can select one or more messages and specify the kickout conditions for transmitting them. **Kickout** lets you transmit messages immediately. However, message transmission via kickout also depends on the priority of the message identifier: If there are other messages to be transmitted with a higher priority, they will be transmitted before the kickout is carried out.

If you trigger a kickout for a message, it is transmitted once. After each message kickout, the corresponding TRC file entry is automatically reset so that you can immediately trigger the next message kickout.

You can enable or disable transmission via kickout, let entries be included in the TRC file, and/or generate a **Triggering** inport for the block. If you select both the TRC file entry and the block inport, the condition is evaluated by an OR operation.

For a better understanding refer to the illustration on the Triggering Options Page (RTICANMM MainBlock) on page 136.

Dialog settings

Specify Kickout for all messages Lets you select messages from all the available messages to specify their kickout conditions for transmission. The message list is disabled if Use these options for all is selected. The following commands are available via context menu:

- Select according to selected Network Nodes
- J1939
- Configuration File

Tip

Press the **`Shift`** or **`Ctrl`** key to select several messages simultaneously.

Select according to selected Network Nodes - Messages received by Network Nodes (Available via context menu) Lets you select messages that are received by the selected network nodes. You can specify the *selected network nodes* on the Network Node Preselection Page (RTICANMM MainBlock) on page 127.

Select according to selected Network Nodes - Messages transmitted by Network Nodes (Available via context menu) Lets you select messages that are transmitted by the selected network nodes. You can specify the *selected network nodes* on the Network Node Preselection Page (RTICANMM MainBlock) on page 127.

Select according to selected Network Nodes - Messages not received by Network Nodes (Available via context menu) Lets you select messages that are not received by the selected network nodes. You can specify the *selected network nodes* on the Network Node Preselection Page (RTICANMM MainBlock) on page 127.

Select according to selected Network Nodes - Messages not transmitted by Network Nodes (Available via context menu) Lets you select messages that are not transmitted by the selected network nodes. You can specify the *selected network nodes* on the Network Node Preselection Page (RTICANMM MainBlock) on page 127.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Source Lets you specify the condition source. You can let an entry be included in the TRC file and/or generate a Triggering inport for the block as the condition source.

Setting	Description
None	Neither a TRC file entry nor a Triggering inport is generated for the selected message(s).
TRC	One entry is included in the TRC file for each selected TX message. The entry lets you enable or disable message transmission. No Triggering inport is generated. If an inport already exists, its structure remains unchanged. This is the default value.
Inport	A Triggering inport is generated if the inport does not yet exist. The entry lets you enable or disable message transmission. If you select cyclic triggering, this is added to the inport structure. See I/O characteristics on page 70. No TRC file entry is generated.
Both	For each selected TX message, one entry is included in the TRC file and a Triggering inport is generated (if the inport does not yet exist). If you select cyclic triggering, this is added to the inport structure. See I/O characteristics on page 70. You can evaluate this option by a logical AND or OR.
*	The source value of the selected message(s) remains unchanged. This allows you to change other options without changing this one. If Use these options for all is enabled, this option is not available.

Use these options for all Select the checkbox to apply the specified settings to all messages. If the checkbox is selected, Set and Specify kickout for all messages are disabled. The checkbox is selected by default.

Set Lets you assign the specified conditions to the selected message(s). This button is disabled if Use these options for all is selected.

Related topics

Basics

Lesson 3: Triggering Message Transmission via the Model and ControlDesk (RTI CAN MultiMessage Blockset Tutorial)

References

Cycle Times Page (RTICANMM MainBlock)

Access

Located in Triggering Page (RTICANMM MainBlock) on page 136.

Dialog pages

You can specify the settings specific to the cycle times and delay times of TX messages on the following pages:

- Cycle Time Defaults Page (RTICANMM MainBlock) on page 151
- TX Cycle Time Page (RTICANMM MainBlock) on page 154
- TX Delay Time Page (RTICANMM MainBlock) on page 156
- Base/Update Time Page (RTICANMM MainBlock) on page 157

Related topics

References

RTICANMM MainBlock.. 68

Cycle Time Defaults Page (RTICANMM MainBlock)

Access

Located in Cycle Times Page (RTICANMM MainBlock) on page 151.

Purpose

To specify the default values for cycle times and delay times.

Description

You can specify default values for the cycle times of TX and RX messages and for the delay times of TX messages.

Cycle times For TX and RX messages, you can specify cycle times.

- *Cycle times for TX messages*: The TX messages are transmitted on the bus according to the specified cycle times. To enable the cyclic transmission of TX messages, you have to specify the **Cyclic** option on the Message Cyclic Page (RTICANMM MainBlock) on page 144.
- *Cycle times for RX messages*: The cycle time is needed as a reference value for calculating cycle time errors. The cycle times specify the time interval in which the reception of RX messages is expected. If an RX message is not received within the specified time interval, a cycle time error can be generated (refer to Cycle Time Error Page (RTICANMM MainBlock) on page 187).

Delay times For TX messages, you can specify delay times. Delay times are used in the following cases:

- When you enable the transmission of a specific TX message, transmission starts after the delay time has elapsed. For details on enabling message transmission, see Message Enable Page (RTICANMM MainBlock) on page 140.
- When you switch the TX message transmission method from **triggered** to **cyclic**, cyclic message transmission starts after the delay time has elapsed (refer to Message Cyclic Page (RTICANMM MainBlock) on page 144).

Specifying cycle and delay times Cycle and delay times can be specified in the database file or they can be specified during run time.

- *Database file*: In some cases, database files contain default values for message-specific attributes such as the cycle time or delay time. These can be used in the RTI CAN MultiMessage Blockset. For details, refer to DBC file as the database on page 100.
- *Run time*: During run time, cycle times and delay times can be changed either from within the model or via entries in the TRC file, refer to TX Cycle Time Page (RTICANMM MainBlock) on page 154 or TX Delay Time Page (RTICANMM MainBlock) on page 156, respectively.

> **Note**
>
> Cycle times and delay times must be multiples of the model's sample time.

Dialog settings

Option Lets you select default values for the cycle times and delay times:

- **Defaults to zero** Lets you set the default values for the cycle times and delay times of all TX messages to zero. While the cycle time is zero, the transmission of the messages is stopped.
- **Defaults from database** Lets you set the default values for the cycle times and delay times of all TX messages according to the specification in the database file.
- **Cycle time from database (delay to zero)** Lets you set the default values for the cycle times of all TX messages according to the specification in the database file.
- **Delay from database (cycle time to zero)** Lets you set the default values for the delay times of all TX messages according to the specification in the database file.
- **User-defined defaults** Lets you set the default values for the cycle times and delay times individually for each TX message.

Edit delay time online Lets you specify to change values for the delay times during run time.

- If the default values for all delay times are set to zero, you have to select this option if you want to change the values for the delay times during run time.

Note

Selecting the **Edit delay time online** option reduces performance since additional code has to be generated.

- If the default values for all delay times are set to zero and if this option is not selected, no code specific to delay times will be generated. This improves performance.

Cycle/delay time defaults Displays the messages that are available for user-defined cycle time defaults. The following commands are available via context menu:

- Set All to Database Default
- Set Selected to Database Default
- J1939
- Configuration File

Set All to Database Default (Available via context menu) Lets you set the default values for the cycle times and delay times of all TX messages according to the specification in the database file.

Set Selected to Database Default (Available via context menu) Lets you set the default values for the cycle time and delay time of the message(s) selected from the list according to the specification in the database file.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Cycle time Lets you specify the cycle time for the selected TX message(s). Enter the time (in ms), then click **Set**.

Delay time Lets you specify the delay time for the selected TX message(s). Enter the time (in ms), then click **Set**.

Set Lets you assign the specified cycle time and delay time to the message(s) selected from the list.

Related topics

References

TX Cycle Time Page (RTICANMM MainBlock)

Access

Located in Cycle Times Page (RTICANMM MainBlock) on page 151.

Purpose

To specify whether the cycle time of TX and RX messages can be changed via the model or via TRC file entries during run time.

Description

Cycle times are used for the cyclic transmission of TX messages and to calculate cycle time errors for RX messages. You can specify default values for cycle times for TX and RX messages (for details, refer to Cycle Time Defaults Page (RTICANMM MainBlock) on page 151). You can change the specified default cycle times during run time.

Changing cycle times during run time During run time, cycle times can be specified either from within the model or via entries in the TRC file.

- *Cycle times from the model*: If you specify to take the cycle times from the model, you have to provide the cycle times via the Triggering inport of the RTICANMM MainBlock. Refer to I/O characteristics on page 70. There will be no entry in the TRC file.
- *Cycle times via entries in the TRC file*: The behaviors for cycle times via TRC file entries are different for TX and RX messages.
  - *TX messages*: An entry in the TRC file for each TX message lets you change the cycle time via ControlDesk during run time.
  - *RX messages*: There will be an entry in the TRC file for each RX message if you enable error display or error ports for the message (refer to RX Error Display Page (RTICANMM MainBlock) on page 185 and RX Error Ports Page (RTICANMM MainBlock) on page 184, respectively). The TRC file entry lets you change the cycle time for the RX message via ControlDesk during run time.

Note

Cycle times must be a multiple of the model's sample time.

Dialog settings

All Lets you specify to take the cycle times of all messages from the model. No TRC file entries are generated.

None Lets you specify the cycle times of all messages via entries in the TRC file. One entry for each TX message is included in the TRC file immediately. Entries for each RX message are included in the TRC file if error display or error ports are enabled.

Selected The cycle times for the selected messages are taken from the model. There will be entries in the TRC file only for the messages that are not selected.

Available messages Displays all the messages that are available in the database file. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select messages in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

Selected messages Displays the selection of messages that is active if you select the Selected option. The following commands are available via context menu:

- All
- None
- J1939 - All Container
- J1939 - All Instances

All (Available via context menu) Lets you add all the available messages to the list.

None (Available via context menu) Lets you remove all messages from the list.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Related topics

References

RTICANMM MainBlock.. 68

TX Delay Time Page (RTICANMM MainBlock)

Access

Located in Cycle Times Page (RTICANMM MainBlock) on page 151.

Purpose

To specify the delay times of TX messages.

Description

For TX messages, you can specify delay times. You can change the specified default delay times during run time. For details on delay times and on specifying default values, refer to Cycle Time Defaults Page (RTICANMM MainBlock) on page 151.

Changing delay times during run time During run time, the delay times of TX messages can be specified either from within the model or via entries in the TRC file.

- *Delay times from the model*: If you specify to take the delay times from the model, you have to provide the delay times via the **Triggering** inport of the RTICANMM MainBlock. Refer to I/O characteristics on page 70. There will be no entry in the TRC file.
- *Delay times via entries in the TRC file*: An entry in the TRC file for each message lets you change the delay time via the **Bus Navigator** in ControlDesk during run time.

On this page, you can specify to change delay times via the model or via TRC file entries during run time.

Note

The delay time must be a multiple of the model's sample time.

Dialog settings

All Lets you specify to take the delay times of all TX messages from the model. No TRC file entries are made.

None Lets you specify the delay times of all TX messages via entries in the TRC file. One entry is included in the TRC file for all the messages.

Selected The delay times for the selected TX messages are taken from the model. There will be entries in the TRC file for the TX messages that are not selected.

Available messages Displays all the messages that are available in the database file. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select messages in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

Selected messages Displays the selection of messages that is active if you select the Selected option. The following commands are available via context menu:

- All
- None
- J1939

All (Available via context menu) Lets you add all the available messages to the list.

None (Available via context menu) Lets you remove all messages from the list.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Related topics

References

RTICANMM MainBlock.. 68

Base/Update Time Page (RTICANMM MainBlock)

Access

Located in Cycle Times Page (RTICANMM MainBlock) on page 151.

Dialog pages

You can specify the settings specific to the base times and update times of TX messages on the following pages:

- Base/Update Messages Page (RTICANMM MainBlock) on page 158
- Base/Update Time Page (RTICANMM MainBlock) on page 159

Related topics

References

RTICANMM MainBlock.. 68

Base/Update Messages Page (RTICANMM MainBlock)

Access

Located in Base/Update Time Page (RTICANMM MainBlock) on page 157.

Purpose

To specify the base and update times of cyclically transmitted TX messages.

Description

If your message transmission is cyclic, you can specify the base and update times. To specify cyclic message transmission, refer to Message Cyclic Page (RTICANMM MainBlock) on page 144.

Base time The base time is used instead of the cycle time. The message will be sent with the base time until a signal of this message changes its value.

Update time If a signal of the message changes its value, the cycle time of the message will be set to the update time for a number of times.

Update number If a signal has changed its value, the update time will be used as cycle time for the number of times specified by the Update number.

During run time, the base and update times of TX messages can be specified either from within the model or via entries in the TRC file.

- *Base and update times from the model*: If you specify to take the base and update times from the model, you have to provide the base and update times via the Triggering inport of the RTICANMM MainBlock. Refer to I/O characteristics on page 70. There will be no entry in the TRC file.
- *User-defined base and update times*: You can specify values for the cycle times on the Base/Update Time Page (RTICANMM MainBlock) on page 157.

Dialog settings

All Lets you specify to take all the base and update times for all TX messages from the model.

None Lets you specify to take the base and update times for all TX messages via entries in the TRC file.

Selected The base and update times for the selected TX messages are taken from the model. There will be entries in the TRC file for the TX messages that are not selected.

Available messages Displays all the messages that are available in the database file. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select messages in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

Selected messages Displays the selection of messages that is active if you select the Selected option. The following commands are available via context menu:

- All
- None
- J1939

All (Available via context menu) Lets you add all the available messages to the list.

None (Available via context menu) Lets you remove all messages from the list.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Related topics

References

Base/Update Time Page (RTICANMM MainBlock) 157
Message Cyclic Page (RTICANMM MainBlock) 144
RTICANMM MainBlock 68

Base/Update Time Page (RTICANMM MainBlock)

Access

Located in Base/Update Time Page (RTICANMM MainBlock) on page 157.

Purpose

To specify the values for base times and update times, and specify the number of times the update time will be used as cycle time.

Description

Using the base/update times functionality of the RTI CAN MultiMessage Blockset, a TX message is transmitted cyclically with the base time until a signal of the message changes its value. If a signal has changed its value, the update time is used as the cycle time for a specified number of times.

You can specify the values for the base and update times of cyclically transmitted TX message.

- *Enabling the cyclic transmission of messages*: To enable the cyclic transmission of messages, you have to specify the **Cyclic** option on the Message Cyclic Page (RTICANMM MainBlock).

- *Specifying base times and update times during run time*: During run time, the base and update times can be specified either from within the model or via an entry in the TRC file. Refer to Base/Update Messages Page (RTICANMM MainBlock) on page 158
- *Default settings in the database file*: In some cases, database files contain default values for message-specific attributes such as the base and update times. These can be used in the RTI CAN MultiMessage Blockset. For details, refer to DBC file as the database on page 100.

Dialog settings

Option Lets you specify values for the base and update times:

- Times from database Lets you set the default values for the base times, update times and number of transmissions of all TX messages according to the specification in the database file.
- User-defined times Lets you set the values for the base times, update times and number of transmissions individually for each TX message.

User-defined base and update times Displays the messages that are available for user-defined base and update times. The following commands are available via context menu:

- Set all to database defaults
- Set selected to database defaults
- Configuration File

Set all to database defaults (Available via context menu) Lets you set the values for the base times, update times and number of transmissions of all TX messages according to the specification in the database file.

Set selected to database defaults (Available via context menu) Lets you set the values for the base times, update times and number of transmissions of the message(s) selected from the list according to the specification in the database file.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Base Time Lets you specify the base time for the selected TX message(s). Enter the time (in ms), then click Set.

Update Time Lets you specify the update time for the selected TX message(s). Enter the time (in ms), then click Set.

Number Lets you specify how often the message is transmitted with the specified update time after the message has changed. Enter the number, then click Set.

Set Lets you assign the specified base time, update time and number of transmissions to the message(s) selected from the list.

Related topics

References

TX Timeout Enable Page (RTICANMM MainBlock)

Access

Located in Triggering Page (RTICANMM MainBlock) on page 136.

Purpose

To specify default values for the timeout of TX messages.

Description

You can specify the default number of timeouts for each TX message.

- *Timeout mechanism*: The timeout value is the number of possible transmissions of a TX message that are ignored. It specifies how often the TX message will not be transmitted even if transmission is triggered by the other trigger conditions. For example, if the timeout value is 5, and you trigger message kickout 6 times, only the sixth message kickout will be carried out. The timeout mechanism is independent of the transmission triggering method.

  If you enable message transmission for many messages at the same time, this can block the CAN bus. In this case, specifying a timeout is useful to avoid blocking the CAN bus. You can also specify a timeout to simulate failures on the CAN bus.

  Note

  Using the timeout mechanism reduces performance since additional code has to be generated.

- *Enabling the timeout mechanism*: To specify default values for the timeout, the timeout mechanism must be enabled on the Triggering Options Page (RTICANMM MainBlock) on page 136.

Dialog settings

Default for timeout enable value Displays the messages that are available for user-defined default timeout enable values. The following commands are available via context menu:

- J1939
- Configuration File

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Countdown Lets you enter the default for the number of possible transmissions of the selected message(s). The valid values are in the range 0 … 100. Then click Set.

Counter Lets you choose the behavior of the counter during timeout. You can specify whether the transmissions of a TX message is counted during timeout.

Value	Description
STOP	During timeout, the counter stops. After timeout, the counter continues counting, starting with the counter value of the last message transmission before the timeout started.
CONTINUE	During timeout, the counter continues counting. The counter value is incremented according to the number of ignored messages.

Set Lets you assign the specified default value to the message(s) selected from the list.

Related topics

Basics

Lesson 8: Customizing Signals (RTI CAN MultiMessage Blockset Tutorial 📖)

References

Trigger Reaction Page (RTICANMM MainBlock)

Access

Located in Triggering Page (RTICANMM MainBlock) on page 136.

Purpose

To trigger reactions on the reception of specific RX messages.

Description

You can trigger the transmission of any TX message whenever a specific RX message is received.

- *Trigger conditions for received RX message*: You can limit the triggering of message transmission by specifying conditions that the received RX message has to fulfill. This allows you to let message transmission be triggered only if a signal in the RX message has a specific value, or if the signal value has changed since the last reception of the message. You can combine trigger conditions logically via AND / OR operations. Nesting trigger conditions, however, is not possible.
- *Options for TX message to be transmitted*: You can specify different options for the TX message to be transmitted:
  - You can let the TX message be transmitted immediately. In this case, it is transmitted at once regardless of the current RTICANMM transmit queue. The message is routed directly to the CAN controller via the real-time interface and has the highest priority in the controller transmit queue. This allows you to let the TX message be transmitted in the current sampling step.
  - You can let the TX message be transmitted via kickout. In this case, it is transmitted at once. If there are other messages to be transmitted with a higher priority, however, these messages will be transmitted before the kickout is carried out.
  - You can let the state of the **Enable** condition and/or **GlobalEnableTX** condition be changed on the reception of a specific RX message. The **Enable** condition enables or disables the transmission of individual messages of this CAN controller variation. The **GlobalEnableTX** condition enables or disables the transmission of messages of this CAN controller variation.
  - You can let the cycle time of the TX message be set/changed on the reception of a specific RX message.

Note

- A high bus load can delay the transmission of triggered TX messages. In this case, a triggered TX message might not be transmitted in the current sampling step.
- These settings affect only the TRC file entries for GlobalEnableTX, Enable and CycleTime.

Dialog settings

Add Lets you add a new reaction to the list of triggered reactions. The Trigger Reaction page (TriggerReaction dialog) is opened.

Edit Lets you edit the reaction selected from the list of triggered reactions. The Trigger Reaction page (TriggerReaction dialog) is opened.

Delete Lets you delete the reaction selected from the list of triggered reactions.

Configured trigger reactions Displays the specified trigger reactions.

Trigger Reaction page (TriggerReaction dialog)

To specify the trigger reaction name.

Name Lets you specify the name of the trigger reaction to be defined. The name must be unique.

Messages page (TriggerReaction dialog)

To specify the message whose reception triggers a reaction, and to specify the TX message that is to be transmitted as the reacting message.

Received message (Can be changed only if you add a new reaction to the list of triggered reactions.) Lets you select the RX message to be used as a trigger for the transmission of the reacting TX message. The list contains all the messages that have been selected on the RX Messages Page (RTICANMM MainBlock) on page 130.

Signal values (Received message) Displays the trigger conditions for the selected received RX message on which message transmission is triggered.

You can change the trigger conditions for the selected received message by specifying different signal values and/or logical operators. Use the edit field and the drop-down list below the **Signal values** list for this. To assign the specified values to the selected signal(s), click **Apply**.

The RTI CAN MultiMessage Blockset provides different options for signal value specification (see table below). They allow you to specify different types of trigger conditions.

Value	Description
`Value`	Message transmission is triggered only if the signal has the specified (numerical) value.
`!Value`	Message transmission is triggered only if the signal does not have the specified numerical value (numerical value with ! in front).
`*`	Message transmission is triggered for any signal value.
`changed`	Message transmission is triggered only if the signal value has changed since the last reception of the message.

Value	Description
`notchanged`	Message transmission is triggered only if the signal value has remained unchanged since the last reception of the message.

Sort - Sort Signal (Available via context menu) Lets you sort the Signal values list alphabetically according to signal names.

Sort - Sort Value (Available via context menu) Lets you sort the Signal values list in ascending order by the Value column.

Sort - Sort Logic (Available via context menu) Lets you sort the Signal values list in ascending order by the Logic column.

Apply (Received message) Lets you assign the specified signal value and the selected logical operator (AND or OR) to the selected signal(s) of the received message.

Options (Received message) Displays the signal value options you can use when you specify the trigger conditions for the received RX message.

Reacting message Lets you select the TX message that is to be transmitted as the reaction to reception of the trigger message. The list contains all the messages that have been selected on the TX Messages Page (RTICANMM MainBlock) on page 128.

Signal values (Reacting message) Displays the signal values of the selected reacting TX message.

You can change the signal values of the reacting message and specify whether the signal values are to be transmitted only once or are also to be used for subsequent message transmissions. Use the edit field and the drop-down list below the Signal values list for this. To assign the specified value to the selected signal(s), click Apply.

When you specify the signal values of the reacting TX message, the following options are available:

Value	Description
`Value`	The value of the signal is set to the specified (numeric) value.
`*`	The value of the signal is evaluated according to the specified signal manipulation options.

Sort - Sort Signal (Available via context menu) Lets you sort the signal values list alphabetically according to signal names.

Sort - Sort Value (Available via context menu) Lets you sort the signal values list in ascending order by the Value column.

Sort - Sort Switch (Available via context menu) Lets you sort the signal values list in ascending order by the Switch column.

Apply (Reacting message) Lets you assign the specified signal value and the selected signal option (Constant/*) to the selected signal(s) of the reacting message.

Options (Reacting message) Displays the signal value options you can use when you specify the signal values of the reacting message.

Options page (TriggerReaction dialog)

To specify trigger reaction options for the transmission of the selected and/or all the TX messages.

Send immediately Lets you transmit the reacting TX message at once regardless of the current RTICANMM transmit queue. It is routed directly to the CAN controller via the real-time interface and has the highest priority in the current controller transmit queue.

Kickout Lets you transmit the reacting TX message once according to its priority in the current RTICANMM transmit queue.

> **Note**
>
> A trigger reaction can be executed only if Send immediately and/or Kickout is selected. If both checkboxes are cleared no reacting messages are triggered at all.

Enable – No changes Lets you specify to leave the Enable condition of the reacting TX message unchanged.

Enable – Off Lets you set the state of the Enable condition to off, i.e., message transmission of the reacting message is disabled. To ensure that this option has an effect on the reacting message, it is recommended to select TRC as Source on the Message Enable Page (RTICANMM MainBlock) on page 140.

Enable – On Lets you set the state of the Enable condition to on, i.e., message transmission of the reacting message is enabled. To ensure that this option has an effect on the reacting message, it is recommended to select TRC as Source on the Message Enable Page (RTICANMM MainBlock) on page 140.

Enable – Toggled Lets you switch the state of the Enable condition of the reacting message: If message transmission of the reacting message was disabled, it is enabled when a trigger event occurs and vice versa. To ensure that this option has an effect on the reacting message, it is recommended to select TRC as Source on the Message Enable Page (RTICANMM MainBlock) on page 140.

Global TX enable – No changes Lets you specify to leave the GlobalEnableTX condition unchanged.

Global TX enable – Off Lets you set the state of the GlobalEnableTX condition to off, i.e., message transmission of all the TX messages of this RTICANMM MainBlock is disabled. To ensure that this option has an effect on the reacting message, it is recommended to select only GlobalEnableTX_TRC on the Triggering Options Page (RTICANMM MainBlock) on page 136.

Global TX enable – On Lets you set the state of the GlobalEnableTX condition to on, i.e., message transmission of all the TX messages of this RTICANMM MainBlock is enabled. To ensure that this option has an effect on the reacting message, it is recommended to select only GlobalEnableTX_TRC on the Triggering Options Page (RTICANMM MainBlock) on page 136.

Global TX enable – Toggled Lets you switch the state of the GlobalEnableTX condition: If transmission of a TX message was disabled, it is enabled when a trigger event occurs and vice versa. To ensure that this option has an effect on the reacting message, it is recommended to select only GlobalEnableTX_TRC on the Triggering Options Page (RTICANMM MainBlock) on page 136.

Cycle Time [ms] Lets you set the cycle time of the reacting TX message. The cycle time is set when the trigger reaction occurs.

> **Tip**
>
> Using this edit field in connection with the Reset switch immediately switch option allows you, for example, to change the cycle time of the reacting message once. The cycle time is changed when the defined trigger reaction occurs.

Send messages with defined signal values just once (Available only if at least one numerical value is specified for the Signal values (Reacting message) on page 165) Lets you transmit the TX message with the numeric signal values you specify in the Signal values (Reacting message) list only once. For all subsequent message transmissions, the signal values are evaluated according to the specified signal manipulation options.

Set defined signal values to constants (Available only if at least one numerical value is specified for the Signal values (Reacting message) on page 165) Lets you take the numeric signal values you specify in the Signal values (Reacting message) list over to the Bus Navigator in ControlDesk. The signal values you specify in the list will then be transmitted for the first and all subsequent message transmissions.

Create activate switch for this trigger reaction Lets you enable or disable the trigger reaction. If the checkbox is selected, an additional TRC file entry for the received message is generated that allows you to switch the trigger reaction on or off. You can specify further configuration options for the switch (see below).

Reset switch immediately (Available only if the Create activate switch for this trigger reaction checkbox is selected) Lets you specify whether the trigger reaction is deactivated after execution. If the checkbox is selected, the defined trigger reaction is executed once and is then deactivated. If the checkbox is cleared, the trigger reaction remains active.

Activate switch by default (Available only if the Create activate switch for this trigger reaction checkbox is selected) Lets you specify whether the trigger reaction switch is activated by default or whether you must activate the trigger reaction by setting the TRC file entry of the trigger reaction switch.

Related topics

References

RTICANMM MainBlock.. 68

Mapping Page (RTICANMM MainBlock)

Access

Located in Triggering Page (RTICANMM MainBlock) on page 136.

Purpose

To specify the mapping structure for TX message triggering signals.

Description

On the Triggering Options Page (RTICANMM MainBlock) on page 136, you can specify the conditions for triggering the transmission of TX messages. To set the **Enable**, **Cyclic**, and **Kickout** conditions, you can let a **Triggering** inport be generated for the RTICANMM MainBlock, and set the conditions individually for each TX message by trigger signals from the model. For details on the **Triggering** inport, refer to I/O characteristics on page 70. For details on the above triggering options, refer to Message Enable Page (RTICANMM MainBlock) on page 140, Message Kickout Page (RTICANMM MainBlock) on page 148 and Message Cyclic Page (RTICANMM MainBlock) on page 144.

- Signal mapping of the **Triggering** inport: The RTICANMM MainBlock has an internal mapping structure for the trigger signals provided via the **Triggering** inport. To connect signals to this inport correctly, you have to connect them according to the RTICANMM MainBlock's internal mapping structure. You can let the Triggering to RTICANMM Block be created automatically for this purpose (see Create triggering block on page 262). This block allows you to easily connect triggering signals to the **Triggering** inport according to the RTICANMM MainBlock's internal mapping structure.
- Specifying your own mapping structure: On this page, you can specify your own mapping structure for triggering signals. You have to specify one triggering signal for each triggering option of each TX message.
  - If you do not specify a triggering signal for each triggering option of each TX message, a Triggering to RTICANMM Block is created. This block internally connects the triggering option inputs for which no corresponding triggering signal is specified to Simulink Constant blocks. Connect triggering signals to the Triggering to RTICANMM Block.
  - If you specify one triggering signal for each triggering option of each TX message, no Triggering to RTICANMM Block is required. Connect triggering signals directly to the RTICANMM MainBlock.

> **Note**
>
> If mapping is used in the RTICANMM MainBlock, all errors involving incorrect structure will appear inside the RTICANMM MainBlock.

Dialog settings

No mapping Lets you specify not to use your own mapping structure for TX message triggering signals.

User-defined mapping Lets you specify your own mapping structure via the User-defined mapping list.

Mapping direct to MainBlock (Available only if you specified a mapping structure for each triggering option of each TX message) If you select this checkbox, the RTICANMM MainBlock uses the specified mapping structure directly, without creating a Triggering to RTICANMM Block.

User-defined mapping Displays the messages that are available for user-defined mapping. The following commands are available via context menu:

- Select from Bus Selector
- Gateway (Selected)
- Gateway (All)
- Gateway ECU hierarchy (Selected)
- Gateway ECU hierarchy (All)
- J1939
- Configuration File
- Renew mapping block

Select from Bus Selector (Available via context menu) Lets you specify your own mapping structure using an existing Simulink Bus Selector block. The structure of the selected Bus Selector block is used to specify the mapping structure of the selected message(s).

Gateway (Selected) (Available via context menu) Lets you specify the RX_Status of a message as the trigger signal for the selected TX message.

When another RTICANMM MainBlock connected to the current RTICANMM MainBlock receives the messages, it sets the RX_Status outputs to 1. This triggers the transmission of the TX messages by the current RTICANMM MainBlock.

Gateway (All) (Available via context menu) Lets you specify the RX_Status of a message as the trigger signal for all the TX messages displayed in the User-defined mapping list.

When another RTICANMM MainBlock connected to the current RTICANMM MainBlock receives the messages, it sets the RX_Status outputs to 1. This triggers the transmission of the TX messages by the current RTICANMM MainBlock.

Gateway ECU hierarchy (Selected) (Available via context menu) Lets you specify an ECU-oriented mapping structure to trigger the transmission of the selected TX messages.

If another RTICANMM MainBlock connected to the current RTICANMM MainBlock provides an output port bus structure with an ECU hierarchy, you can map the selected TX messages to this bus structure (see also Peripheral Options Page (RTICANMM MainBlock) on page 262). When the connected RTICANMM MainBlock receives messages, it sets the RX_Status outputs to 1. This triggers the transmission of the TX messages by the current RTICANMM MainBlock.

Gateway ECU hierarchy (All) (Available via context menu) Lets you specify an ECU-oriented mapping structure to trigger the transmission of all the TX messages displayed in the User-defined mapping list.

If another RTICANMM MainBlock connected to the current RTICANMM MainBlock provides an output port bus structure with an ECU hierarchy, you can map all TX messages to this bus structure (see also Peripheral Options Page (RTICANMM MainBlock) on page 262). When the connected RTICANMM MainBlock receives messages, it sets the RX_Status outputs to 1. This triggers the transmission of the TX messages by the current RTICANMM MainBlock.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Renew mapping block Lets you quickly connect your application to the RTICANMM MainBlock after you modify the block's mappings. You do not need to create the whole MainBlock after modifying mappings, but if you want to, you can select Force “Create” from the Options menu.

Mapping structure Lets you enter the mapping structure for the selected message(s). Then click Set.

Set Lets you assign the specified mapping structure to the message(s) selected from the list.

Macros Opens a dialog that informs you which macros are available for defining a mapping structure. The following macros are available:

- %CAN
- %Controller
- %ECU
- %ReceiveECU
- %MsgName
- %MsgID
- %SigName

Related topics

Basics

References

Ports & Displays Page (RTICANMM MainBlock)

Access

Located in Messages Page (RTICANMM MainBlock) on page 121.

Dialog pages

You can specify the settings specific to ports and displays on the following pages:

- TX Status Ports Page (RTICANMM MainBlock) on page 171
- RX Status and Time Ports Page (RTICANMM MainBlock) on page 173
- RX ID Port Page (RTICANMM MainBlock) on page 174
- RX Message Length Port Page (RTICANMM MainBlock) on page 175
- RX Message Counter Page (RTICANMM MainBlock) on page 176
- Raw Data Page (RTICANMM MainBlock) on page 178
- Errors Page (RTICANMM MainBlock) on page 183

Related topics

References

TX Status Ports Page (RTICANMM MainBlock)

Access

Located in Ports & Displays Page (RTICANMM MainBlock) on page 171.

Purpose

To enable status ports for TX messages.

Description

You can enable status ports for one or more TX messages. In this case, the current RTICANMMM MainBlock gets the RX Data outport if the outport does not yet exist.

The RX Data outport provides the following status information relating to the transmission of TX messages.

TX_Status	Description
0	TX message not yet transmitted
1	TX message transmitted

Dialog settings

All Lets you enable status ports for all TX messages.

None Lets you disable status ports for all TX messages.

Selected Lets you enable status ports for a set of TX messages.

Available messages Displays the available TX messages. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select messages in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

Selected messages Displays the selection of messages that is active if you select the Selected option. The following commands are available via context menu:

- All
- None
- J1939

All (Available via context menu) Lets you add all the available messages to the list.

None (Available via context menu) Lets you remove all messages from the list.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Related topics

References

RX Status and Time Ports Page (RTICANMM MainBlock)

Access

Located in Ports & Displays Page (RTICANMM MainBlock) on page 171.

Purpose

To enable status ports and time ports for RX messages.

Description

You can enable status ports and time ports for one or more RX messages. The current RTICANMMM MainBlock gets the **RX Data** outport if the outport does not yet exist.

- *Status information*: The **RX Data** outport provides the following status information relating to the reception of RX messages.

RX_Status	Description
0	New RX message not yet received
1	New RX message received in current sampling step

- *Time information*: The **RX Data** outport provides the following information:
  - RX_Time – the point in time (in seconds) when the message was received.
  - RX_DeltaTime – the time difference (in seconds) between the points in time when the current message and the previous message were received.

Dialog settings

All Lets you enable status ports and time ports for all RX messages.

None Lets you disable status ports and time ports for all RX messages.

Selected Lets you enable status ports and time ports for a set of RX messages.

Available messages Displays the available RX messages. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select messages in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically **Up** or **Down**.

Selected messages Displays the selection of messages that is active if you select the **Selected** option. The following commands are available via context menu:

- All
- None
- J1939

All (Available via context menu) Lets you add all the available messages to the list.

None (Available via context menu) Lets you remove all messages from the list.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Related topics

References

RTICANMM MainBlock... 68

RX ID Port Page (RTICANMM MainBlock)

Access

Located in Ports & Displays Page (RTICANMM MainBlock) on page 171.

Purpose

To provide the IDs of selected messages and their identifier formats as outports. The source and destination addresses are output for selected J1939 messages.

Description

This creates the RX Data outport as a bus for the RTICANMM MainBlock (if it does not already exist) and adds the IDs and the identifier formats of the selected messages, and the source and destination addresses of the selected J1939 messages, to it.

Dialog settings

All Lets you enable ports for all RX messages.

None Lets you disable ports for all RX messages.

Selected Lets you enable ports for a set of RX messages.

Available messages Displays the available RX messages. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select messages in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

Selected messages Displays the selection of RX messages that is active if you select the Selected option. The following commands are available via context menu:

- All
- None
- J1939

All (Available via context menu) Lets you add all the available messages to the list.

None (Available via context menu) Lets you remove all messages from the list.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Related topics

References

RX Message Length Port Page (RTICANMM MainBlock)

Access

Located in Ports & Displays Page (RTICANMM MainBlock) on page 171.

Purpose

To provide the message length of selected messages as an outport.

Description

This creates the RX Data outport as a bus for the RTICANMM Main block (if it does not already exist) and adds the message lengths of the selected messages to it.

This is especially important when free raw messages are processed.

Dialog settings

All Lets you enable Message Length ports for all RX messages.

None Lets you disable Message Length ports for all RX messages.

Selected Lets you enable Message Length ports for a set of RX messages.

Available messages Displays the available RX messages. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select messages in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

Selected messages Displays the selection of messages that is active if you select the Selected option. The following commands are available via context menu:

- All
- None
- J1939

All (Available via context menu) Lets you add all the available messages to the list.

None (Available via context menu) Lets you remove all messages from the list.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Related topics

References

RTICANMM MainBlock 68

RX Message Counter Page (RTICANMM MainBlock)

Access

Located in Ports & Displays Page (RTICANMM MainBlock) on page 171.

Purpose

To specify an RX message counter for RX and/or TX messages.

Description

You can select one or more messages and specify an RX message counter for them. The RX message counter counts the number of messages received. You can enable or disable the RX message counter, let entries be included in the TRC file, and/or generate an RX Data outport for the block.

Dialog settings

Specify RX counter messages Lets you select messages to specify an RX message counter for. The message list is disabled if **Use these options for all** is selected. The following commands are available via context menu:

- J1939
- Configuration File

Tip

Press the **`Shift`** or **`Ctrl`** key to select several messages simultaneously.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Destination Lets you specify the counter destination. You can let an entry be included in the TRC file and/or generate an **RX Data** outport for the block as the counter destination.

Setting	Description
None	No RX message counter is created for the selected messages. This is the default value.
TRC	One entry is included in the TRC file for each selected message. The entry lets you enable or disable the RX message counter. No **RX Data** outport is generated. If an outport already exists, its structure remains unchanged.
Outport	An **RX Data** outport is generated if it does not yet exist. To set the message-specific condition, the outport structure is extended accordingly. See I/O characteristics on page 70. No TRC file entry is generated.
Both	For each message, one entry is included in the TRC file and/or an **RX Data** outport is generated if it does not yet exist. To set the message-specific condition, the outport structure is extended accordingly. See

Setting	Description
	I/O characteristics on page 70. This condition is evaluated by an OR operation.

Set Lets you assign the specified conditions to the selected message(s). This button is disabled if Use these options for all is selected.

Use these options for all Select the checkbox to apply the specified settings to all messages. If the checkbox is selected, Set and Specify RX counter messages are disabled.

Related topics

References

Raw Data Page (RTICANMM MainBlock)

Access

Located in Ports & Displays Page (RTICANMM MainBlock) on page 171.

Dialog pages

You can specify the settings specific to raw data on the following pages:

- TX Raw Data Page (RTICANMM MainBlock) on page 178
- TX Raw Data Display Page (RTICANMM MainBlock) on page 180
- RX Raw Data Ports Page (RTICANMM MainBlock) on page 181
- RX Raw Data Display Page (RTICANMM MainBlock) on page 182

Related topics

References

TX Raw Data Page (RTICANMM MainBlock)

Access

Located in Raw Data Page (RTICANMM MainBlock) on page 178.

Purpose

To let you manipulate TX messages with raw data.

Description

The messages can be manipulated either via TRC file or via inport. During run time, you can select whether raw data or signals are used. The default setting for this switch can also be specified on this dialog page.

Dialog settings

Messages for manipulation with raw data from inport or TRC Displays the messages that can be manipulated with raw data from inport or TRC. The following commands are available via context menu:

- J1939
- Configuration File

> **Tip**
>
> Press the **`Shift`** or **`Ctrl`** key to select several messages simultaneously.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Activate raw data option Lets you activate or deactivate the raw data option for the selected message(s).

Source of raw data Lets you select the source of raw data for the selected message(s). This can be either an Inport or the TRC file.

Default Lets you specify the default for the selected message(s). This can be either Signals or Raw Data.

Set Lets you assign the specified settings to the message(s) selected from the list.

Related topics

References

RTICANMM MainBlock 68

TX Raw Data Display Page (RTICANMM MainBlock)

Access

Located in Raw Data Page (RTICANMM MainBlock) on page 178.

Purpose

To enable raw data displays for TX messages.

Description

You can specify to let entries for TX raw data be included in the generated TRC file. In this case, you can display TX raw data in ControlDesk.

> **Note**
>
> The TRC file entries for TX raw data are generated only if the manipulation option is activated for the TX message(s) on the TX Raw Data Page (RTICANMM MainBlock) on page 178.

Dialog settings

All Lets you enable raw data displays for all TX messages.

None Lets you disable raw data displays for all TX messages.

Selected Lets you enable raw data displays for a set of TX messages.

Available messages Displays the available TX messages. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select messages in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

Selected messages Displays the selection of messages that is active if you select the Selected option. The following commands are available via context menu:

- All
- None
- J1939

All (Available via context menu) Lets you add all the available messages to the list.

None (Available via context menu) Lets you remove all messages from the list.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Related topics

References

RX Raw Data Ports Page (RTICANMM MainBlock)

Access

Located in Raw Data Page (RTICANMM MainBlock) on page 178.

Purpose

To enable raw data ports for RX messages.

Description

You can enable raw data ports for one or more RX messages. The current RTICANMMMM MainBlock gets the **RX Data** outport if the outport does not yet exist.

The **RX Data** outport provides direct access to the unchanged bytes (raw data) of an RX message. This allows you to manipulate a message byte-wise.

Tip

Connect a Simulink Bus Selector to the RX Data outport for easy access to the raw data of RX messages.

Dialog settings

All Lets you enable raw data ports for all RX messages.

None Lets you disable raw data ports for all RX messages.

Selected Lets you enable raw data ports for a set of RX messages.

Available messages Displays the available RX messages. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select messages in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

Selected messages Displays the selection of messages that is active if you select the Selected option. The following commands are available via context menu:

- All
- None
- J1939

All (Available via context menu) Lets you add all the available messages to the list.

None (Available via context menu) Lets you remove all messages from the list.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Related topics

References

RX Raw Data Display Page (RTICANMM MainBlock)

Access

Located in Raw Data Page (RTICANMM MainBlock) on page 178.

Purpose

To enable raw data displays for RX messages.

Description

You can specify to let entries for RX raw data be included in the generated TRC file. You can then display RX raw data in ControlDesk.

Dialog settings

All Lets you enable raw data displays for all RX messages.

None Lets you disable raw data displays for all RX messages.

Selected Lets you enable raw data displays for a set of RX messages.

Available messages Displays the available RX messages. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select messages in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

Selected messages Displays the selection of messages that is active if you select the Selected option. The following commands are available via context menu:.

- All
- None
- J1939

All (Available via context menu) Lets you add all the available messages to the list.

None (Available via context menu) Lets you remove all messages from the list.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Related topics

References

Errors Page (RTICANMM MainBlock)

Access

Located in Ports & Displays Page (RTICANMM MainBlock) on page 171.

Dialog pages

You can specify the settings specific to CAN communication errors on the following pages:

- RX Error Ports Page (RTICANMM MainBlock) on page 184
- RX Error Display Page (RTICANMM MainBlock) on page 185
- Cycle Time Error Page (RTICANMM MainBlock) on page 187

Related topics

References

RX Error Ports Page (RTICANMM MainBlock)

Access

Located in Errors Page (RTICANMM MainBlock) on page 183.

Purpose

To enable error ports for RX messages.

Description

You can enable error ports for one or more RX messages. The current RTICANMM MainBlock gets the **RX Data** outport if the outport does not yet exist.

RX_Error bits: The **RX Data** outport provides two RX_Error bits relating to the reception of RX messages. If an error bit is **1**, the corresponding error has occurred.

RX_Error Bit Position	Description
0	If the cycle time for receiving an RX message exceeds the specified range, the first RX_Error bit is set to **1**. You can specify the range on the Cycle Time Error Page (RTICANMM MainBlock) on page 187.
1	If a checksum algorithm is applied to an RX message, a checksum is calculated for the message and compared to the checksum within the received RX message. The second RX_Error bit is set to **1** if these checksums differ. You can specify the algorithm on the Checksum Definition Page (RTICANMM MainBlock) on page 200.
2	The third RX_Error bit shows whether the counter value is correct. Each time the counter value is incorrect, the internal counter is automatically synchronized with the current value. This is also done whenever the CAN communication is started, i.e. when this message is received the first time. Therefore, the first counter value is always incorrect and causes an error bit. You can specify the counter settings on the Counter Page (RTICANMM MainBlock) on page 240.

Dialog settings

All Lets you enable error ports for all RX messages.

None Lets you disable error ports for all RX messages.

Selected Lets you enable error ports for a set of RX messages.

Available messages Displays the available RX messages. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select messages in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

Selected messages Displays the selection of messages that is active if you select the Selected option. The following commands are available via context menu:

- All
- None
- J1939

All (Available via context menu) Lets you add all the available messages to the list.

None (Available via context menu) Lets you remove all messages from the list.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Related topics

References

RX Error Display Page (RTICANMM MainBlock)

Access

Located in Errors Page (RTICANMM MainBlock) on page 183.

Purpose

To enable error displays for RX messages.

Description

You can specify to let entries for RX errors be included in the generated TRC file. You can then display RX errors in ControlDesk. You have to assign a message's error variable to an instrument for this.

The possible error values are:

RX Error Value	Description
1	The cycle time for receiving an RX message exceeds the specified range. You can specify the range on the Cycle Time Error Page (RTICANMM MainBlock) on page 187.
2	If a checksum algorithm is applied to an RX message, a checksum is calculated for the message and compared to the checksum within the received RX message. These checksums differ. You can specify the algorithm on the Checksum Definition Page (RTICANMM MainBlock) on page 200.
4	The counter value is not correct. Each time the counter value is incorrect, the internal counter is automatically synchronized with the current value. This is also done whenever the CAN communication is started, i.e., when this message is received the first time. Therefore, the first counter value is always incorrect and causes an error. You can specify the counter settings on the Counter Page (RTICANMM MainBlock) on page 240.

If several RX errors relating to the reception of an RX message occur, the corresponding error values are added.

Dialog settings

All Lets you enable error displays for all RX messages.

None Lets you disable error displays for all RX messages.

Selected Lets you enable error displays for a set of RX messages.

Available messages Displays the available RX messages. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select messages in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

Selected messages Displays the selection of messages that is active if you select the Selected option. The following commands are available via context menu:

- All
- None
- J1939

All (Available via context menu) Lets you add all the available messages to the list.

None (Available via context menu) Lets you remove all messages from the list.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Related topics

Basics

Lesson 7: Customizing Messages (RTI CAN MultiMessage Blockset Tutorial)

References

Cycle Time Error Page (RTICANMM MainBlock)

Access

Located in Errors Page (RTICANMM MainBlock) on page 183.

Purpose

To specify the ranges for calculating cycle time errors.

Description

To generate cycle time errors, two preconditions must be fulfilled:

- A cycle time must be specified for the RX message (refer to Cycle Time Defaults Page (RTICANMM MainBlock) on page 151).
- Ranges for calculating the cycle time error must be specified.

When an RX message is received, a delta time is calculated between the current message reception and the previous message reception. The delta time is compared with the cycle time specified for the RX message. If the calculated delta time differs from the cycle time and exceeds a specified range, the first RX_Error bit is set to **1** and a cycle time error occurs (refer to RX Error Ports Page (RTICANMM MainBlock) on page 184).

Since cycle time errors are calculated between the current and the previously received RX message, there are the following limitations:

- The same RX message must be received at least twice.
- Cycle time errors cannot be calculated for messages whose reception failed.

On this page, you can specify the ranges for calculating the cycle times.

Dialog settings

Always calculate from timeout tolerance [%] Lets you specify a percentage tolerance range for the defined default message cycle time (refer to Cycle Time Defaults Page (RTICANMM MainBlock) on page 151). For example, the specified cycle time of a message is 100 ms. With an error tolerance of 50%, an error will occur for cycle times lower than 50 ms or greater than 150 ms. This setting applies to all RX messages.

Note

If you change the cycle time of a message during run time, the percentage error range is also recalculated automatically.

User-defined ranges Lets you specify the tolerance range for the cycle time individually for each RX message. The list displays the user-defined ranges. The following commands are available via context menu:

- Set all items to tolerance values
- Set selected items to tolerance values
- J1939
- Configuration File

Tip

If the **User-defined values** option is selected, you can change the absolute values for the maximum and minimum time limits for the individual messages via entries in the TRC file.

Set all items to tolerance values (Available via context menu) Lets you set the tolerance ranges for the cycle times of all messages in the list in a single step. The minimum and maximum time limit values for each message are computed from the message's default cycle time (specified on the Cycle Time Defaults Page (RTICANMM MainBlock) on page 151) and the specified timeout tolerance percentage value.

Set selected items to tolerance values (Available via context menu) Lets you set the tolerance ranges for the cycle times of the selected messages in one step. For each selected message, the minimum and maximum time limit values are computed from the message's default cycle time (specified on the Cycle Time

Defaults Page (RTICANMM MainBlock) on page 151) and the specified timeout tolerance percentage value.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Min [ms] (Available only if User-defined ranges is selected) Lets you specify the minimum time limit (in milliseconds). Click Set to assign the setting to the selected message(s).

Note

If a cycle time is changed during run time, this range can be exceeded.

Max [ms] (Available only if User-defined ranges is selected) Lets you specify the maximum time limit (in milliseconds). Click Set to assign the setting to the selected message(s).

Note

If a cycle time is changed during run time, this range can be exceeded.

Set Lets you assign the specified ranges to the messages(s) selected from the User-defined ranges list.

Related topics

Basics

Lesson 7: Customizing Messages (RTI CAN MultiMessage Blockset Tutorial 🕮)

References

Message Manipulation Page (RTICANMM MainBlock)

Access

Located in Messages Page (RTICANMM MainBlock) on page 121.

Dialog pages

You can specify the settings specific to checksum calculations and custom code on the following pages:

- Manipulation Options Page (RTICANMM MainBlock) on page 190
- TX ID Page (RTICANMM MainBlock) on page 191
- J1939 TX ID Page (RTICANMM MainBlock) on page 193
- TX Message Length Page (RTICANMM MainBlock) on page 195
- Message Defaults Page (RTICANMM MainBlock) on page 197
- Message Variations Page (RTICANMM MainBlock) on page 198
- Checksum Page (RTICANMM MainBlock) on page 200
- Custom Code Page (RTICANMM MainBlock) on page 211

Related topics

References

RTICANMM MainBlock.. 68

Manipulation Options Page (RTICANMM MainBlock)

Access

Located in Message Manipulation Page (RTICANMM MainBlock) on page 190.

Purpose

To enable or disable message manipulation.

Description

You can manipulate TX messages during run time in different ways.

Dialog settings

Activate dynamic MessageLength Lets you activate dynamic message length. If dynamic message length is activated, you can change the length of a TX message for a time period of n transmissions in the Bus Navigator of ControlDesk.

Activate dynamic CRC Lets you activate a dynamic checksum algorithm. If dynamic CRC is activated, you can change the checksum algorithm of a TX message for a period of n transmissions in the Bus Navigator of ControlDesk.

Related topics

Basics

References

TX ID Page (RTICANMM MainBlock)

Access

Located in Message Manipulation Page (RTICANMM MainBlock) on page 190.

Purpose

To let you modify the IDs and message formats of non-J1939 TX messages during run time.

Description

The IDs and message formats of non-J1939 TX messages can be modified either via TRC file or via inport. Such manipulations during run time are especially important when you work with free raw messages (refer to Free Raw Messages Page (RTICANMM MainBlock) on page 131).

Note

- If you change an ID during run time, make sure that no other messages (also no free raw messages) use this ID at the same time.
- If both manipulation of the message format and manipulation of the message length are enabled for a message and if you change the message format during run time, the range of the possible message lengths is not adjusted. This means the following:
  - *Both manipulation options are enabled for a classic CAN message*: If you switch the message format during run time to a format for CAN FD messages (e.g., FD STD), the range of possible message lengths is still 0 ... 8. It is not increased although the CAN FD message could be longer.
  - *Both manipulation options are enabled for a CAN FD message*: If you switch the message format during run time to a message format for classic CAN messages (e.g., STD), the range of possible message lengths is not decreased to 0 ... 8, but stays at the range that was specified for the CAN FD message. You yourself have to make sure that your messages do not exceed the upper limit of 8 bytes.

  For more information on manipulating the lengths of TX messages, refer to TX Message Length Page (RTICANMM MainBlock) on page 195.

Dialog settings

Configured ID manipulations Displays the messages with adjustable IDs.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Manipulation option Lets you specify the manipulation option for the message(s) selected from the **Configured ID manipulations** list.

Tip

Press the **`Shift`** or **`Ctrl`** key to select several messages simultaneously.

Settings	Description
NONE	The ID cannot be modified during run time.
TRC	The ID can be modified via TRC during run time.
INPORT	The ID can be modified via an inport during run time.
BOTH (DEFAULT TRC)	By default, the ID can be modified via TRC. You can change this default setting during run time.
BOTH (DEFAULT INPORT)	By default, the ID can be modified via an inport. You can change this default setting during run time.
*	The manipulation option of the selected message(s) remains unchanged. This allows you to change other options without changing this one.

Set Lets you assign the specified manipulation option to the message(s) selected from the list.

Related topics

Basics

References

J1939 TX ID Page (RTICANMM MainBlock)

Access

Located in Message Manipulation Page (RTICANMM MainBlock) on page 190.

Purpose

To let you modify the priority and the source and destination nodes of J1939 TX messages during run time.

Description

The priority, the source node and destination node (only peer-to-peer) of J1939 TX messages can be modified either via TRC file or via inport. You can also specify to address the source and destination nodes of each individual J1939 message by their network addresses or their node indices.

Depending on the selected addressing mode, the **J1939SA** and **J1939DA** (for addressing by network address), or **J1939SNIDX** and **J1939DNIDX** (for addressing by node indices) model inputs are provided to the **TX Data** inport of the RTICANMM MainBlock.

Dialog settings

Configured J1939 ID manipulations Displays the J1939 messages with adjustable IDs.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Priority Lets you specify the manipulation option for the priority of the message(s) selected from the **Configured J19939 ID manipulations** list.

Tip

Press the **`Shift`** or **`Ctrl`** key to select several messages simultaneously.

Settings	Description
NONE	The priority cannot be modified during run time.
TRC	The priority can be modified via TRC during run time.
INPORT	The priority can be modified via an inport during run time.

Settings	Description
BOTH (DEFAULT TRC)	The priority can be modified during run time via TRC and via inport. The default is modification via TRC file and can be changed during run time.
BOTH (DEFAULT INPORT)	The priority can be modified during run time via TRC and via inport. The default is modification via inport and can be changed during run time.
*	The priority value of the selected message(s) remains unchanged. This allows you to change other options without changing this one.

Source/Destination Lets you specify the manipulation option for the source network node and the destination network node (only peer-to-peer) of the message(s) selected from the **Configured J1939 ID manipulations** list:

Settings	Description
NONE	The source/destination network node cannot be changed during run time.
TRC	The source/destination network node can be changed via TRC file during run time.
INPORT	The source/destination network node can be changed via an inport during run time.
BOTH (DEFAULT TRC)	The source/destination nodes can be changed during run time via TRC file and via inport. The default is to use the TRC file and can be changed during run time.
BOTH (DEFAULT INPORT)	The source/destination nodes can be changed during run time via TRC file and via inport. The default is to use an inport and can be changed during run time.
*	The source/destination node values of the selected message(s) remain unchanged. This allows you to change other options without changing this one.

Node Select via Lets you specify how to address the source/destination network nodes of the selected J1939 message(s):

Settings	Description
ADDRESS	The network nodes are addressed via their network addresses.
NODE INDEX	The network nodes are addressed via the source node indices and destination node indices.
*	The node addressing mode of the selected message(s) remains unchanged. This allows you to change other options without changing this one.

Set Lets you assign the specified option(s) to the J1939 message(s) selected from the list.

Related topics

Basics

Lesson 13 (Advanced): Working with a J1939-Compliant DBC File (RTI CAN MultiMessage Blockset Tutorial 📖)
Lesson 7: Customizing Messages (RTI CAN MultiMessage Blockset Tutorial 📖)

References

RTICANMM MainBlock 68

TX Message Length Page (RTICANMM MainBlock)

Access

Located in Message Manipulation Page (RTICANMM MainBlock) on page 190.

Purpose

To let you modify the lengths of TX messages during run time.

Description

The lengths of TX messages can be modified either via TRC file or via inport. Modifying the message lengths during run time is especially important when you work with free raw messages (refer to Free Raw Messages Page (RTICANMM MainBlock) on page 131).

Dialog settings

Adjustment options for message length(s) Displays the messages with adjustable message length. The following commands are available via context menu:

- Set all to database defaults
- Set selected to database defaults
- J1939
- Configuration File

Set all to database defaults (Available via context menu) Lets you set the default values for the message length of all TX messages according to the specification in the database file.

Set selected to database defaults (Available via context menu) Lets you set the default values for the message length of the message(s) selected from the list according to the specification in the database file.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Adjust option Lets you specify the adjust option for the message(s) selected from the **Adjustment options for message length(s)** list:

Setting	Description
NONE	The TX message length cannot be manipulated during run time.
TRC	The TX message length can be modified via TRC during run time.
INPORT	The TX message length can be modified via an inport during run time.
*	The default adjustment option of the selected message(s) remains unchanged. This allows you to change other options without changing this one.

Message length Lets you enter the default message length.

- For non-J1939 classic CAN messages, the maximum message length is 8 bytes.
- For CAN FD messages, the maximum message length is 64 bytes. For information on the possible data length code (DLC) values, refer to Basics on Working with CAN FD on page 25.
- For contained IPDUs, the maximum length is 60 bytes (64 bytes maximum frame length reduced by 4 bytes reserved for the short header).

  In some cases, the maximum length might even be smaller than 60 bytes, for example, if a contained IPDU or container IPDU is the payload of a secured IPDU. In such cases, some bytes are reserved for `AuthInfoTxLength` and `FreshnessValueTxLength`.
- For J1939 messages, the maximum message length is 1785 bytes.

Note

There are some points to note regarding TX message length manipulation:
- *CAN messages with a maximum length of 8 bytes*: During run time, you can manipulate the message length in the full CAN message range (0 ... 8 bytes), even if you specified a shorter length as the default.
- *CAN FD messages with a maximum length > 8 bytes*: During run time, you can manipulate the message length in the range 0 ... <default message length>. The default message length you specify here is the upper limit for the message length you can set during TX message length manipulation. For this reason, you should enter the maximum possible length for the message.

Shortening the message length to a value smaller than the specified default length during run time is always possible.

Set Lets you assign the specified adjust option to the message(s) selected from the list.

Related topics

Basics

Lesson 7: Customizing Messages (RTI CAN MultiMessage Blockset Tutorial 📖)

References

Message Defaults Page (RTICANMM MainBlock)

Access

Located in Message Manipulation Page (RTICANMM MainBlock) on page 190.

Purpose

To define the bits' default value in a message data field.

Description

You can define the default values of bits in a message data field that do not belong to a signal. You can address each bit via a unique hexadecimal value, and set the default value to 0 or 1 for each bit individually. The default value is ignored if a bit is being used by a signal.

Dialog settings

All 0x0 Lets you set the default value of all bits to 0.

All 0xFFFFFFFFFFFFFFFF Lets you set the default value of all bits to 1.

User-defined defaults Lets you define the default value of each bit via a hexadecimal value. Select this option to enable the message list, the **Default value** edit field, and **Set**. The message list shows all the available messages. Select a message to define the bits' default values. The following commands are available via context menu:

- Get all defaults from database
- Get selected defaults from database
- J1939
- Configuration File

Tip

Press the `Shift` or `Ctrl` key to select several messages simultaneously.

Get all defaults from Database (Available via context menu) Lets you set the default values of bits in a message data field according to the specification in the database file.

Get selected defaults from Database (Available via context menu) Lets you set the default values of bits in the message data field(s) selected from the list according to the specification in the database file.

J1939 - All Container (Available via context menu) Lets you select all containers of J1939 messages.

J1939 - All Instances (Available via context menu) Lets you select all instances of J1939 messages. You can define instances on the Instance to Container Page (RTICANMM MainBlock) on page 125.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Default value Lets you specify the default value of each bit in the selected message(s). The value must be entered in decimal or hexadecimal format. The edit field is enabled if User-defined defaults is selected.

Set Lets you assign the bits' default values to the selected message(s). This button is enabled if User-defined defaults is selected.

Related topics

References

Message Variations Page (RTICANMM MainBlock)

Access

Located in Message Manipulation Page (RTICANMM MainBlock) on page 190.

Purpose

To specify several variants of a message, which you can switch during run time without having to switch between MainBlock variants. This can be used if you have message variants with overlapping signals, for example.

Description

Overlapping signals are generally not allowed in the CAN standard. However, you can use message variations to generate overlapping signals into n variants of a message that do not have overlapping signals. To use message variants you must enable changing overlapping signals to message variations on the General Settings Page (refer to Change overlapping signals to message variations on page 106).

Dialog settings

Message variations Lists the messages that contain overlapping signals and lets you select the default variation and source for each message. The following commands are available via context menu:

- Filter
- Sort
- Configuration File

Filter (Available via context menu) Lets you filter and select messages in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Default variation Lets you select the default variation for the message selected from the Variations for messages list.

Source Lets you select the source by which you can switch between the message variations for the selected message in the Variations for messages list.

Settings	Description
TRC	During run time, the message variation can be changed via TRC file.
Inport	During run time, the message variation can be changed via an Inport.
Only Default	During run time, the message variation cannot be changed but the default message variation is taken.

Set Lets you assign the specified default variation and source to the message selected in the **Variations for messages** list.

Related topics

References

RTICANMM MainBlock.. 68

Checksum Page (RTICANMM MainBlock)

Access

Located in Message Manipulation Page (RTICANMM MainBlock) on page 190.

Dialog pages

You can specify the checksum definitions and messages on the following pages:

- Checksum Definition Page (RTICANMM MainBlock) on page 200
- Checksum Messages Page (RTICANMM MainBlock) on page 205

Related topics

References

RTICANMM MainBlock.. 68

Checksum Definition Page (RTICANMM MainBlock)

Access

Located in Checksum Page (RTICANMM MainBlock) on page 200.

Purpose

To specify checksum algorithms.

Description

You can implement checksum algorithms for messages via a *checksum header file*. Each algorithm has to have a C-coded switch-case directive in the header file.

Note

- Do not include files in checksum header files. Instead, you should include files via the Includes Page (RTICANMM MainBlock) on page 254.
- For J1939 messages, the CRC option is limited to the first eight bytes.

For instructions on how to create a checksum header file, refer to Step 4: How to Specify Checksums (RTI CAN MultiMessage Blockset Tutorial 🕮).

Input parameters of the header file

In the checksum header file, edit your checksum algorithms. You have access to the following input parameters of the header file:

Input Parameter	Description	
`crcoption`	▪ "0" if applied to a TX message ▪ "1" if applied to an RX message	
`RTICANMMMsgStruct* Msg`	Message structure with pointers to the following information:	
	`Msg->RAW_DATA[byteIndex 0...7]`	Raw data of the message
	`Msg_Length = Msg->len`	Length of the message
	`Id = Msg->identifier`	Identifier of the message (non-J1939 message)
`crctype`	Index of the switch-case directives for the checksum algorithms. It corresponds to the order of the specified algorithms in the Defined cases identifier list.	
`CsBitPos`	Start position of the checksum signal	
`CsLength`	Length of the checksum signal	

Checksum algorithms based on end-to-end communication protection

You can implement checksum algorithms based on end-to-end communication (E2E protection) parameters. The checksum algorithms can then contain data that is not transmitted on the bus. E2E protection checksum algorithms are implemented in the same checksum header file as the checksum algorithms without E2E protection data.

You must enable the use of E2E protection parameters. If enabled, an additional header file (named `RTICANMM_MAIN_<CAN name>_E2EP.h`) is generated and included into the S-function of the RTICANMM MainBlock. The header file contains the E2E protection description data defined in the database and the functions to call this data in checksum algorithms. You can use the E2E protection data in checksum algorithm implementations. The E2E protection header file is also saved to the destination folder.

The RTI CAN MultiMessage Blockset supports run-time access to E2E protection parameters from AUTOSAR communication matrices and DBC files. Note the following information:

- *AUTOSAR files*: In AUTOSAR, E2E communication protection is related to signal groups and their IPDUs. Since RTICANMM does not handle signal groups, the E2E protection data is imported on message level.

  The following table lists the attributes that are available in the generated header file for using the E2E protection parameters from AUTOSAR files in checksum algorithms. The corresponding AUTOSAR attributes are also displayed.

Attribute Available in Generated Header File	AUTOSAR Attribute	Data Type
ProfileName	Category	String
DataOffset	DataOffset	Integer
CounterPosition[1)]	CounterOffset	Integer
CRCPosition[2)]	CRCOffset	Integer
DataIDMode	DataIDMode	Integer
DataIDs	DataIDs	Array of Integers
DataLength	DataLength	Integer
MaxDeltaCounterInit	MaxDeltaCounterInit	Integer
MaxNoNewOrRepeatedData	MaxNoNewOrRepeatedData	Integer
SyncCounterInit	SyncCounterInit	Integer
LengthOffset[3)]	LengthOffset	Integer

[1)] The AUTOSAR attribute `CounterOffset` describes a position relative to `DataOffset` in the PDU. However, the imported `CounterPosition` attribute describes an absolute position in the message. The attribute's value is calculated as follows:
`CounterPosition = PDU StartBit + DataOffset + CounterOffset`.

[2)] The AUTOSAR attribute `CRCOffset` describes a position relative to `DataOffset` in the PDU. However, the imported `CRCPosition` attribute describes an absolute position in the message. The attribute's value is calculated as follows:
`CRCPosition = PDU StartBit + DataOffset + CRCOffset`.

[3)] Only available for profile 06.

The RTI CAN MultiMessage Blockset supports E2E protection according to the following AUTOSAR end-to-end protection profiles:

- Profile 01
- Profile 02
- Profile 05
- Profile 06
- Profile 11
- Profile 22

Profiles 05 and 06 are available only for AUTOSAR Release 4.2.1 or later. Profiles 11 and 22 are available only for AUTOSAR Release 4.3.0 or later.

The RTI CAN MultiMessage Blockset supports E2E transformers for profile 02, profile 05, and profile 06 for calling the end-to-end communication protection library (E2E library).

Tip

For an example on working with E2E protection data according to AUTOSAR E2E profiles, refer to Lesson 16 (Advanced): Working with E2E Communication Protection According to AUTOSAR E2E Profiles (RTI CAN MultiMessage Blockset Tutorial 📖).

The `E2E_checksum_advanced.h` file, which is used as the checksum header file in the tutorial lesson, contains E2E protection algorithms for all AUTOSAR E2E profiles supported by the RTI CAN MultiMessage Blockset. You can therefore use the file as a template in your project.

- *DBC files*: In DBC files, E2E information is transported via the custom attributes `GenMsgChkConstant` and `GenMsgPDUConstants`. If both attributes are defined for a message, the `GenMsgChkConstant` attribute is used.

  In the generated header file, the `ProfileName` attribute is used for these DBC attributes. A value from `GenMsgChkConstant` or `GenMsgPDUConstants` is stored in the `DataIDs` attribute.

  RTICANMM does not transport any other data related to E2E from DBC files. If DBC files are used to implement AUTOSAR-defined profiles, this must be handcoded in the checksum header file.

To use E2E protection parameters, edit your checksum header file to find associated E2E indices via search function. The search function finds all the E2E cases for a specific message defined by `MessageId` and `MessageIdFormat`. You can then insert the corresponding get functions for specific E2E attributes in the checksum algorithms where necessary. Get functions are used to find the E2E attributes that are based on the E2E indices detected via the search function.

You can get a preview of the E2E header file via the **Show E2E File** button.

Dialog settings

Header file Lets you specify a header file containing checksum algorithms. Click to browse to the file.

Use relative path Lets you set the folder relative to the model root folder or as an absolute path.

> **Tip**
>
> Using a relative path allows you to easily move the model and all the files belonging to it to another location on your file system without having to change any path settings.

Create Header File Lets you create a header file according to the switch-case directives you defined.

Edit Header File Lets you edit the selected header file in MATLAB's M-File Editor.

Check Header File Lets you check the specified function call in the header file for syntax errors. The check also returns the number of calculation algorithms (switch-case directives) contained in the header file.

Note

This option does not check the entire C code of the header file for syntax errors. The C code of the header file is checked by MATLAB during the build process of the model.

Identifier for cases Lets you define an identifier for a checksum algorithm (switch-case directives).

Defined cases identifier Lists the defined cases identifiers. For each defined cases identifier, Case <n> shows the checksum case (switch-case directive) in the checksum header file to which it refers when a header file is created via the Create Header File button. You can sort the identifiers via buttons.

Note

Sorting the identifiers via the buttons does not affect the order in checksum header files that have already been created. It only affects the order in files that will be created via the Create Header File button.

Enable E2E Protection Lets you enable or disable the use of E2E protection parameters. If the checkbox is selected, E2E protection data defined in the AUTOSAR or DBC file is imported into an additionally generated E2E protection header file, which is included automatically in the S-function of the RTICANMM MainBlock. The imported E2E protection data is available for checksum algorithm implementation. The generated header file is also saved in the destination folder. If the checkbox is cleared, no E2E protection parameters are imported from the AUTOSAR or DBC file. Checksum algorithms based on E2E protection parameters are not possible in this case.

Show E2E File Lets you show a preview of the E2E protection header file. An editor opens, displaying the generated header file containing the E2E protection data and the functions to call them.

E2E protection data is automatically updated each time the RTICANMM MainBlock's database file has changed. The preview of the E2E header file can be useful for tasks such as checking whether the database update necessitates modifying the checksum algorithms in the checksum header file.

Related topics

Basics

Aspects of Miscellaneous Supported AUTOSAR Features.. 18

Lesson 15 (Advanced): Using Checksums Based on E2E Protection Parameters (RTI CAN MultiMessage Blockset Tutorial 📖)
Lesson 16 (Advanced): Working with E2E Communication Protection According to AUTOSAR E2E Profiles (RTI CAN MultiMessage Blockset Tutorial 📖)
Lesson 6: Performing a CAN Restbus Simulation (RTI CAN MultiMessage Blockset Tutorial 📖)

References

RTICANMM MainBlock.. 68

Checksum Messages Page (RTICANMM MainBlock)

Access

Located in Checksum Page (RTICANMM MainBlock) on page 200.

Purpose

To assign checksum algorithms to messages.

Description

You can assign one checksum algorithm defined in the *checksum header file* (see Checksum Definition Page (RTICANMM MainBlock) on page 200) to each message. The way the checksum algorithm is applied is different for TX messages and RX messages:

- *Checksum calculation for TX messages*: You can assign a checksum algorithm to each TX message. In this case, a checksum is calculated according to the applied algorithm, and assigned to the TX message. Then the message is transmitted together with the calculated checksum.
- *Checksum check for RX messages*: You can assign a checksum algorithm to each RX message. A checksum is then calculated for the received message and compared to the checksum that was sent with it. If they differ, this is indicated at the corresponding error ports for RX messages if these ports are enabled. For details, refer to RX Error Ports Page (RTICANMM MainBlock) on page 184. The signals of the RX message nevertheless are read.

Note

You cannot implement the following methods directly on this page and the Checksum Definition Page (RTICANMM MainBlock) on page 200:

- You want to implement checksum algorithms that do not involve a specific checksum signal.
- You want to assign more than one checksum algorithm to a message.

To implement these methods, you have to implement custom code on the Custom Code Page (RTICANMM MainBlock) on page 211.

Dialog settings

Message Lets you select a message to which you want to assign a checksum algorithm.

Signal Lets you specify one signal contained in the selected message as the checksum signal.

Algorithm Lets you select the checksum algorithm to be applied. The displayed algorithms are contained in the header file selected on the Checksum Definition Page (RTICANMM MainBlock) on page 200.

Default There will be an entry in the generated TRC file that lets you enable or disable checksum calculation. This option lets you specify whether checksum calculation for the selected message is enabled or disabled by default.

Set Lets you apply the checksum calculation settings.

- If you did not yet specify checksum calculation settings for the selected message, a new entry will be added to the list of checksum messages.
- If you already specified checksum calculation settings for the selected message(s), the new settings will be applied.

Delete Lets you delete the selected message from the List of defined checksum signals.

Filter Lets you filter and select signals in the Filter Dialog (RTICANMM MainBlock) on page 91.

Defined checksum signals Lists the defined checksum signals. You can select message(s) to change already specified checksum calculation settings.

> **Tip**
>
> You can use **`Shift`** or **`Ctrl`** for multiselection.

Via context menu, you can create or load a configuration file or add this page settings to an existing configuration file.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Related topics

Basics

Lesson 6: Performing a CAN Restbus Simulation (RTI CAN MultiMessage Blockset Tutorial 📖)
Lesson 7: Customizing Messages (RTI CAN MultiMessage Blockset Tutorial 📖)

References

RTICANMM MainBlock 68

Secure Onboard Communication Page (RTICANMM MainBlock)

Access

Located in Message Manipulation Page (RTICANMM MainBlock) on page 190.

Purpose

To enable SecOC support and provide the algorithms for generating the authentication information via user code.

Description

The RTI CAN MultiMessage Blockset supports secure onboard communication (SecOC) according to AUTOSAR 4.3.0. Refer to Aspects of Miscellaneous Supported AUTOSAR Features on page 18.

To implement secure onboard communication in applications, you must enable SecOC support and provide the algorithms for generating the authentication information via user code.

User code for secure onboard communication

In general, user code is C code or C++ code that contains user-specific algorithms. You can use user-specific algorithms to add functionality to the RTI CAN MultiMessage Blockset, for example, for generating authentication information in SecOC scenarios. A user code implementation consists of at least one source file (C, CPP) and optional include files (H, HPP), such as header files.

Providing user code for secure onboard communication To use secure onboard communication at run time, you must provide the required OEM-specific implementation for generating authentication information via user code as follows:

1. Specify the user code, i.e., the implementation for generating authentication information according to your needs in C code (C, H files) or C++ code (CPP, HPP files).

2. Extend the user code by functions for initializing secure onboard communication and for exchanging data with the RTI CAN MultiMessage Blockset, such as accessing properties of secured IPDUs or writing generated authentication information to secured IPDUs. Use the Bus Custom Code interface for this purpose.

   For more information, refer to Bus Custom Code Interface Handling 🕮.
3. Add the code files, i.e., the user code implementation:
   - SCALEXIO: Add the user code implementation to the ConfigurationDesk application via properties of build configuration sets (**Bus Custom Code** category).

     You can add one or more user code implementations. All user code implementations that are added to one build configuration set apply to all application processes that are assigned to this build configuration set. Each application process that is assigned to a build configuration set must use all the modules that are contained in any user code implementation of the build configuration set.

     For more information on build configuration sets, refer to Specifying Options for the Build Process (ConfigurationDesk Real-Time Implementation Guide 🕮). For more information on the properties of the **Bus Custom Code** category and the bus custom code options, refer to Build Configuration Table (ConfigurationDesk User Interface Reference 🕮).
   - Non-SCALEXIO systems: Add the user code implementation to the real-time application via the Code Generation's **Custom Code** page or by adapting the user makefile (`USER_SRCS`). Refer to Adding User-Specific Files, Source Files and Libraries (RTI and RTI-MP Implementation Guide 🕮) and Using C++ Code in an RTI Application (RTI and RTI-MP Implementation Guide 🕮)
4. Reference the required user code. To do so, type the user code ID as specified in the user code implementation into the **User code identifier** edit field. In the user code implementation, the user code ID is specified via the `DS_BUS_CUSTOM_FEATURE_NAME` definition of the related source file (C, CPP).

Additional TRC file variables for SecOC

If secure onboard communication is enabled, the following additional TRC file variables are generated for you to access authentication data and verification information of sent or received secured IPDUs at run time in ControlDesk:

TRC Variable	Dir.	Description
<MessageName>_SecOC_AuthEnable	TX	Lets you specify how the authenticator value is calculated when a SecOC message is transmitted. ▪ 0: Indicates to the user code that a custom authenticator value is to be used. The custom authenticator value is manually set during run time via the SecOC message signal `Signal_<MessageName>_Auth`. ▪ 1: Indicates to the user code that the authenticator value is to be calculated as specified by AUTOSAR 4.3.0. The range of the possible values for a custom authenticator depends on the `AuthInfoTxLength` property specified in the communication matrix for each SecOC message.

TRC Variable	Dir.	Description
<MessageName>_SecOC_AuthType	TX	Lets you specify whether the authenticator value is to be invalidated when a SecOC message is transmitted. ▪ 0: The authenticator value is not to be invalidated in the user code. ▪ Others: The authenticator value is to be invalidated in the user code. After invalidation, the new value must lead to a failed verification when the SecOC message is received.
<MessageName>_SecOC_FreshnessCalcEnable	TX	Lets you specify how the freshness value is to be calculated when a SecOC message is transmitted. ▪ 0: Indicates to the user code that a custom freshness value is used. You must specify the custom freshness value manually during run time via the SecOC message signal `Signal_<MessageName>_Freshness`. The range of possible values for a custom freshness value depends on the `FreshnessValueTxLength` property specified in the communication matrix for each SecOC message. ▪ 1: Indicates to the user code that the freshness value is read from a valid time source. The freshness value is used to calculate the authenticator value and set the value for the truncated freshness signal transmitted with the SecOC message.
<MessageName>_SecOC_FreshnessCalcOffset	TX	Lets you specify an offset value that is added to the freshness value in the user code. The offset can be used to invalidate the freshness value.
<MessageName>_SecOC_FreshnessCalcResult	TX	Displays the complete freshness value calculated by the user code. The `<MessageName>_SecOC_FreshnessCalcResult` value is influenced by the `<MessageName>_SecOC_FreshnessCalcOffset` variable. As a consequence, you are recommended not to use an offset value if you want to use the actual freshness value.
<MessageName>_SecOC_RxVerificationEnable	RX	Lets you specify a `VerificationEnable` variable value during run time that is passed to the user code when a SecOC message is received. The variable is used in the user code to indicate whether the verification process is to be executed.
<MessageName>_SecOC_RxVerificationResult	RX	Displays the result of the verification process for a SecOC message. This variable is calculated in the user code and read by RTICANMM when a SecOC message is received.
<MessageName>_SecOC_FreshnessCalcResult	RX	Displays the result of the calculated complete freshness value for a SecOC message. The variable is calculated in the user code and read by RTICANMM when a SecOC message is received. The lower bytes of the calculated value are based on the received freshness value, the higher bytes are based on the expected freshness value (read from the user code).

Dialog settings

Enable Secure Onboard Communication Lets you enable or disable secure onboard communication support for a configuration:

- If the checkbox is selected, the support of secure onboard communication is enabled. With SecOC enabled, the payload of authentic IPDUs is secured by the related secured IPDUs, i.e, by authentication information that is generated according to an OEM-specific implementation provided by user code. The required user code must be referenced via a user code ID, which you specify in the **User code identifier** edit field.
- If the checkbox is cleared, the support of secure onboard communication is disabled. No user code is used, regardless of whether a user code ID is specified. No authentication information is generated and the payload of the assigned authentic IPDUs is not secured, i.e., SecOC authentication and verification are not performed.

  Authentic IPDUs, which are assigned to the configuration, are handled as follows: The payload of the authentic IPDUs is directly included in the secured IPDUs. The bits of the secured IPDUs that are reserved for the authentication information are not used. These bits are filled with the related bit pattern for unused bits (bit pattern 0). The `Freshness` and `Auth` signals are treated like normal signals, i.e., their values can be changed via inport mappings, TRC signal variables, or other mechanisms supported for signal manipulation in the RTI CAN MultiMessage Blockset.

User code identifier Lets you enter the user code ID that references the user code file containing the algorithms used to secure the payload of authentic IPDUs. The identifier must be a valid C identifier. Only letters, numbers and '_' are allowed and the string must be at least one character long. The edit field can only be edited and is only evaluated if the **Enable Secure Onboard Communication** checkbox is selected.

The user code identifier you enter here is used as a postfix for the names of SecOC-specific dSPACE bus custom code functions called during run time:

- `DsBusCustomCode_onApplicationInit_<User code identifier>`
- `DsBusCustomCode_onPduFeatureExecution_<User code identifier>`

Related topics

Basics

Aspects of Miscellaneous Supported AUTOSAR Features.. 18

References

RTICANMM MainBlock.. 68

Custom Code Page (RTICANMM MainBlock)

Access

Located in Message Manipulation Page (RTICANMM MainBlock) on page 190.

Purpose

To specify custom code functions for TX messages.

Description

Specifying custom code works by creating a header file and customizing it.

You can manipulate the raw data in TX messages with functions defined in a custom code header file. You can let these functions be executed before or after checksum calculation for the message.

Note

- Do not include files in custom code header files. Instead, you should include files via the Includes Page (RTICANMM MainBlock) on page 254.
- You cannot manipulate mode signals via custom code header files (for example, to determine which mode-dependent signals are to be transmitted). However, the mode-dependent signals that are transmitted can be manipulated via custom code header files. For details on mode and mode-dependent signals, refer to Glossary (RTI CAN MultiMessage Blockset Tutorial 📖).
- For J1939 messages, the custom code option is limited to the first eight bytes.

Input parameters of the header file

Custom code is implemented as follows: You have to create a function in the header file that passes the `RTICANMMMsgStruct` parameter. `RTICANMMMsgStruct` is a structure with the following attributes:

Attribute	Description
UInt32 RAW_DATA[64];	Raw data of the message
UInt8 len;	Length of the message
UInt8 idformat;	Identifier format of the message: ▪ 0: Classic CAN message with standard format (STD) ▪ 1: Classic CAN message with extended format (EXT) ▪ 2: CAN FD message with standard format (CAN STD FD) ▪ 3: CAN FD message with extended format (CAN EXT FD) ▪ 6: CAN FD message with standard format with baud rate switch (CAN STD FD BRS) ▪ 7: CAN FD message with extended format with baud rate switch (CAN EXT FD BRS)
UInt8 Enable;	Enables or disables message transmission: ▪ 0: Disabled ▪ 1: Enabled

Attribute	Description
UInt8 Cyclic;	Cyclic or triggered message transmission: ▪ 0: Triggered ▪ 1: Send periodically
UInt32 identifier;	CAN identifier For J1939 messages, only the PGN is shown.
UInt32 dppf;	J1939 data page format
UInt8 prio;	Priority of the J1939 message
UInt8 source;	Source of the J1939 message
double deltatime,time;	RX_DeltaTime, RX_Time (absolute time)
double DelayTime,CycleTime;	Delay time for TX message, cycle time for TX message
UInt8 status;	RX status (active for one sampling step): ▪ 0: Not received ▪ 1: Currently received

Dialog settings

Message Lets you select a TX message to which you want to apply a function defined in the custom code.

Function name Lets you enter the name of the custom code function you want to execute.

Option Lets you specify when the specified custom code function is applied.

- pre CRC executes the function before checksum calculation.
- post CRC executes the function after checksum calculation.

Default There will be an entry in the generated TRC file that lets you enable or disable the custom code function. This option lets you specify whether the custom code function for the selected message is enabled or disabled by default.

Custom code (header file) Lets you specify a header file containing custom code. Click to browse to the file.

Use relative path Lets you set the folder relative to the model root folder or as an absolute path.

Tip

Using a relative path allows you to easily move the model and all the files belonging to it to another location on your file system without having to change any path settings.

Set Lets you apply the settings for the custom code function.

- If you did not yet specify custom code function settings for the selected message, a new entry will be added to the List of defined custom code functions.
- If you already specified custom code function settings for the selected message(s), the new settings will be applied.

Delete Lets you delete the selected entry from the List of defined custom code functions.

Create Header File Lets you create a header file according to the settings you defined.

Edit Header File Lets you edit the selected header file in MATLAB's M-File Editor.

Defined custom code functions Lists the defined custom code functions. You can select message(s) to change already specified custom code function settings.

> **Tip**
>
> You can use **`Shift`** or **`Ctrl`** for multiselection.

The following commands are available via context menu for sorting the messages:

- Sort Message
- Sort File Name
- Sort Function
- Sort Option
- Sort Default

Related topics

Basics

Glossary (RTI CAN MultiMessage Blockset Tutorial 📖)

References

Signals Page (RTICANMM MainBlock)

Access

Located on the top level of the dialog tree.

Dialog pages

You can specify the settings specific to signals on the following pages:

- TX Page (RTICANMM MainBlock) on page 214
- Model Signals (RX) Page (RTICANMM MainBlock) on page 246
- Gateway Page (RTICANMM MainBlock) on page 248

Related topics

References

RTICANMM MainBlock.. 68

TX Page (RTICANMM MainBlock)

Access

Located in Signals Page (RTICANMM MainBlock) on page 213.

Dialog pages

You can specify the various aspects of signals to be transmitted on the following pages:

- Model Signals (TX) Page (RTICANMM MainBlock) on page 214
- Input Manipulation Page (RTICANMM MainBlock) on page 216
- Saturation Options Page (RTICANMM MainBlock) on page 218
- Signal Defaults Page (RTICANMM MainBlock) on page 221
- Signal Ranges Page (RTICANMM MainBlock) on page 223
- Signal Errors Page (RTICANMM MainBlock) on page 225
- Signal Mappings Page (RTICANMM MainBlock) on page 227
- Signal Manipulation Page (RTICANMM MainBlock) on page 230
- Signal Default Manipulation Page (RTICANMM MainBlock) on page 244

Related topics

References

RTICANMM MainBlock.. 68

Model Signals (TX) Page (RTICANMM MainBlock)

Access

Located in TX Page (RTICANMM MainBlock) on page 214.

Purpose

To specify TX signals as TX model signals.

Description

The messages defined as TX messages on the TX Messages Page (RTICANMM MainBlock) on page 128 contain the signals to be transmitted (TX signals). The generated TRC file gets an entry for each TX signal. This entry allows you to manipulate the TX signal from within ControlDesk.

TX model signals: If you want to manipulate TX signals from within the model, you have to specify them as TX model signals. For signals specified as TX model signals, the RTICANMM MainBlock gets a **TX Data** inport (see I/O characteristics on page 70) that allows you to manipulate the signal values from within the model. The generated TRC file gets no entries for signals specified as TX model signals.

> **Note**
>
> For optimum performance, you should specify as few TX model signals as possible.

> **Tip**
>
> The Input Manipulation Page (RTICANMM MainBlock) on page 216 lets you specify input manipulation for TX model signals. The generated TRC file also gets entries for these TX model signals. This allows you to manipulate the signal values either from within the model, or from within ControlDesk.

Dialog settings

All Lets you specify all TX signals as TX model signals.

None Lets you specify none of the available TX signals as TX model signals.

Selected Lets you specify a set of the available TX signals as TX model signals.

Available signals Displays the TX signals that are available in the TX messages. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select signals in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically **Up** or **Down**.

Selected signals Displays the selection of TX model signals that is active if you select the **Selected** option. The following commands are available via context menu:

- All
- None

All (Available via context menu) Lets you add all the available signals to the list.

None (Available via context menu) Lets you remove all signals from the list.

Related topics

Basics

Lesson 4: Manipulating Signal Values via the Model and ControlDesk (RTI CAN MultiMessage Blockset Tutorial 📖)

References

RTICANMM MainBlock.. 68

Input Manipulation Page (RTICANMM MainBlock)

Access

Located in TX Page (RTICANMM MainBlock) on page 214.

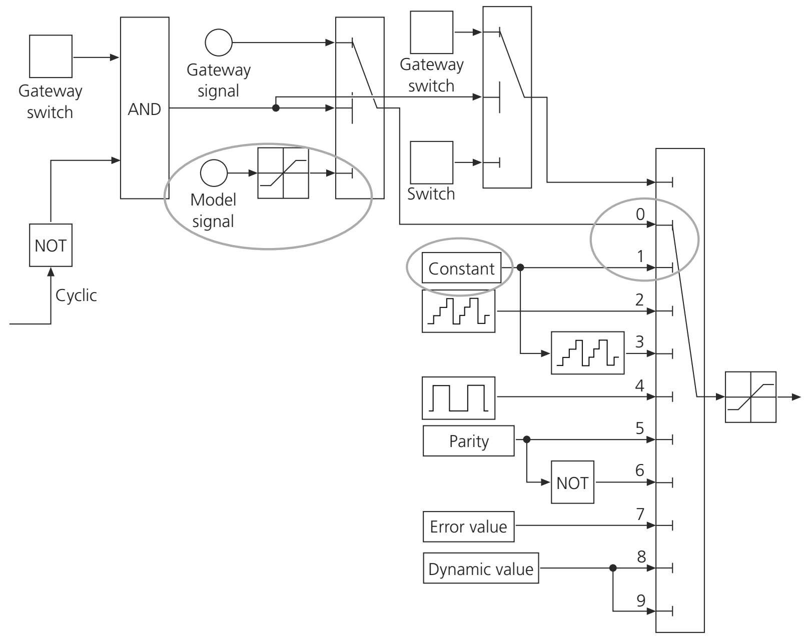
Purpose

To specify input manipulation for TX model signals.

Description

When you select standard TX signals as TX model signals on the Model Signals (TX) Page (RTICANMM MainBlock) on page 214, the RTICANMM MainBlock gets the **TX Data** inport for the TX model signals. You can disable or enable input manipulation for TX model signals.

- *Input manipulation disabled*: If you disable input manipulation for a TX model signal, its value is determined by the corresponding model input via the **TX Data** inport (see I/O characteristics on page 70). In the generated TRC file, there will be no entry for the TX model signals for which you disable input manipulation.
- *Input manipulation enabled*: If you enable input manipulation for a TX model signal, its value is determined either by the respective model input via the **TX Data** inport (see I/O characteristics on page 70), or via an entry in the generated TRC file. In the generated TRC file, there will also be an entry for each TX model signal that lets you switch between using the model input and the TRC file entry. The illustration below visualizes this switch.



The table below shows how TX model signals and standard TX signals can be manipulated.

TX Model Signal		**TX Signal**
Input Manipulation Enabled	**Input Manipulation Disabled**	**(Input Manipulation Not Available)**
▪ Via TRC file entry ▪ Via model input (switch option)	Via model input	Via TRC file entry

Dialog settings

All Lets you enable input manipulation for all TX model signals.

None Lets you disable input manipulation for all of the TX model signals.

Selected Lets you enable input manipulation for a set of TX model signals.

Available signals Displays the signals that are available in the TX messages. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select signals in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

Selected signals Displays the selection of signals that is active if you select the Selected option. The following commands are available via context menu:

- All
- None

All (Available via context menu) Lets you add all the available signals to the list.

None (Available via context menu) Lets you remove all signals from the list.

Related topics

Basics

Lesson 4: Manipulating Signal Values via the Model and ControlDesk (RTI CAN MultiMessage Blockset Tutorial 📖)

References

Saturation Options Page (RTICANMM MainBlock)

Access

Located in TX Page (RTICANMM MainBlock) on page 214.

Dialog pages

You can specify saturation and ranges on the following page:

- Saturation Page (RTICANMM MainBlock) on page 218

Related topics

References

Saturation Page (RTICANMM MainBlock)

Access

Located in Saturation Options Page (RTICANMM MainBlock) on page 218.

Purpose

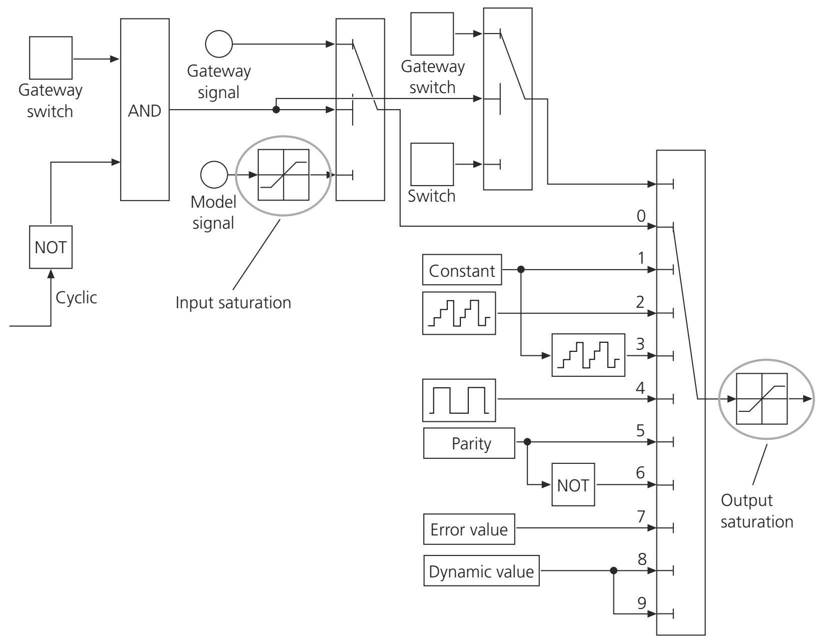
To specify signal saturation.

Description

Signal saturation means:

- TX model signals are limited according to the limits specified in the database file (input saturation) or on the Signal Ranges Page (RTICANMM MainBlock) on page 223.

- After signal manipulation and before conversion of signal values into raw data, the signal values are limited according to the limits of the raw data (output saturation).



The Signal Ranges Page (RTICANMM MainBlock) on page 223 lets you specify saturation limits other than the limits specified in the database file.

Note

For optimum performance, you should specify saturation for as few signals as possible.

Dialog settings

Option Lets you specify the saturation.

- No saturations Lets you specify to perform no signal saturation.
- All signals both saturations Lets you specify input saturation and output saturation for all signals.
- User-defined saturations Lets you specify signal saturation individually for each signal via the Saturation drop-down list.

Use bit saturations Lets you specify to use bit saturations. Bit saturation means saturating signals to limits that are based on their data types and the bit width available for the CAN bus. Bit saturation is applied to integer signals only, it is not possible in connection with signals of float or double data type. If the checkbox is selected, bit saturation is enabled for all signals. The signals are saturated to the bit ranges of the coded signals, that is, if a signal exceeds its limit, it is saturated to the highest or lowest value which can be transmitted in coded format on the CAN bus. If the checkbox is cleared, the signals also remain within their bit range limits. But after a signal reaches its upper bit range limit, it turns around to its lower bit range limit.

Example: Suppose you have an 8-bit unsigned integer signal and your CAN bus has a bit width of 8 bits. With bit saturation enabled, a signal value of 258 (0x102) is saturated to 255 (0xFF). With bit saturation disabled, the signal value is cut to 2 (0x02).

User-defined saturations Displays the messages that are available for user-defined saturations. You can create or load a configuration file or add these page settings to an existing configuration file via context menu.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Saturation Lets you specify the signal direction the saturation will be applied to:

- Both: Saturation applies to both the input and output signal (input and output saturation).
- Input: Saturation applies to the input signal (input saturation). All other signal sources (for example, error value, constant, counter) are not saturated. Input saturation uses the values specified on the Signal Ranges Page (RTICANMM MainBlock) on page 223.
- Output: Saturation applies to the output signal (output saturation). The RTI CAN MultiMessage Blockset provides several signal sources for the signal to be transmitted, such as model input, counter, or constant. The signal value that has been selected by the signal source switch is saturated, regardless of which signal source is used. Output saturation uses the values specified on the Signal Ranges Page (RTICANMM MainBlock) on page 223.
- No: No saturation is performed.

Click Set to assign the setting to the selected signal(s).

Set Lets you assign the specified saturation to the message(s) selected in the User-defined saturations list.

Related topics

References

RTICANMM MainBlock..68

Signal Defaults Page (RTICANMM MainBlock)

Access

Located in TX Page (RTICANMM MainBlock) on page 214.

Purpose

To specify default values for the signals to be transmitted.

Description

You can specify default values for the signals to be transmitted (TX signals). You can either specify default values according to your requirements, or use the default values specified in the database file.

Default settings in the database file: In some cases, database files contain default values for message-specific attributes such as the cycle time. These can be used in the RTI CAN MultiMessage Blockset. For details, refer to DBC file as the database on page 100.

> **Note**
>
> Some DBC files, however, do not provide default values for the signals to be transmitted. For details on providing default values in the DBC file, refer to the documentation of your DBC file editor.

Dialog settings

Defaults to zero Lets you set the default values of all TX signals to zero.

Defaults from database Lets you set the default values of all TX signals according to the specification in the database file. This option is only sensible if you work with a database file. Refer to General Settings Page (RTICANMM MainBlock) on page 100.

User-defined defaults Lets you set default signal values individually for each TX signal.

Value representation – Physical Lets you specify to enter and display the default values in physical units. This setting applies to all TX signals on the list.

Value representation – Coded Lets you specify to enter and display the default values in coded representation in the same way as they are transmitted on the CAN bus, i.e., according to their *original* representation on the hardware. This setting applies to all TX signals on the list.

> **Note**
>
> The value representation setting does not specify the format in which the information is defined in the database file. Instead, it determines the representation format used for data display and data input on this page.
> For DBC, FIBEX and AUTOSAR system description files, the defaults must always be stored in coded representation. For hand-written M files, the value representation can differ.

User-defined signal defaults Lists the user-defined signal defaults. The following commands are available via context menu:

- Set all to database defaults
- Set selected to database defaults
- Set all to Value/Name Pairs Start Value
- Set selected to Value/Name Pairs Start Value
- Configuration File

Set all to database defaults (Available via context menu) Lets you set the default signal value of all TX signals according to the specification in the database file.

Set selected to database defaults (Available via context menu) Lets you set the default signal value of the TX signal(s) selected from the list according to the specification in the database file.

Set all to Value/Name Pairs Start Value (Available via context menu) In some cases, a database file contains a list of names for signal values. This allows you to let these signals be represented by value/name pairs. You can use this command to set the default signal value of all TX signals to the start value of this list.

Set selected to Value/Name Pairs Start Value (Available via context menu) In some cases, a database file contains a list of names for signal values. This allows you to let these signals be represented by value/name pairs. You can use this command to set the default signal value of the selected TX signals to the start value of this list.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Default value Lets you specify the default signal value for the selected message(s). You can enter a numerical value in decimal format, or type `fromDBC` to keep the default value specified in the database file. Then click Set.

Set Lets you assign the specified default signal value to the TX signal(s) selected from the list.

Related topics

References

Signal Ranges Page (RTICANMM MainBlock)

Access

Located in TX Page (RTICANMM MainBlock) on page 214.

Purpose

To specify saturation range limits.

Description

Signal saturation requires appropriate saturation range limits. You can use the range limits specified in the database file or specify other range limits.

> **Note**
>
> The values displayed on this page are always represented in physical units.

Dialog settings

Ranges from database Lets you use the saturation range limits specified in the database file. This option is only sensible if you work with a database file. Refer to General Settings Page (RTICANMM MainBlock) on page 100.

> **Note**
>
> There is no consistency check for the saturation range limits specified in the database file. However, you can get information on the range limits specified in the database file via the database file check report (HTML file). The check is available on the General Settings Page (RTICANMM MainBlock) on page 100.

Change min=0 & max=0 to bit ranges Lets you specify saturation according to the bit range limits for the signals for which `0` is specified as the minimum value and maximum value in the database file.

User-defined ranges Lets you specify saturation range limits according to your requirements. The list displays the messages available for user-defined ranges. The following commands are available via context menu:

- Set all to min & max values from database
- Set all to absmin & absmax values from database
- Set selected to min & max values from database
- Set selected to absmin & absmax values from database
- Set items with min=0 & max=0 to absmin & absmax values from database
- Configuration File

Set all to min & max values from database (Available via context menu) Lets you assign the saturation range limits specified as the min & max values in the database file to all signals. The signals will be saturated according to the physical range limits.

Set all to absmin & absmax values from database (Available via context menu) Lets you assign the saturation range limits specified as the absmin & absmax values in the database file to all signals. Absmin & absmax represent the smallest possible and the greatest possible values that are available for the current signal using the assigned conversion formula.

Set selected to min & max values from database (Available via context menu) Lets you assign the saturation range limits specified as the min & max values in the database file to the selected signal(s). The signal(s) will be saturated according to the physical range limits.

Set selected to absmin & absmax values from database (Available via context menu) Lets you assign the saturation range limits specified as the absmin & absmax values in the database file to the selected signal(s). Absmin & absmax represent the smallest possible and the greatest possible values that are available for the current signal using the assigned conversion formula.

Set items with min=0 & max=0 to absmin & absmax values from database (Available via context menu) Lets you assign the saturation range limits specified as the absmin & absmax values in the database file to all the signals for which min = 0 and max = 0. Absmin & absmax represent the smallest possible and the greatest possible values that are available for the current signal using the assigned conversion formula.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Min Lets you specify the lower saturation limit for the selected signal(s). You can enter:

- A numerical value
- `fromDBC`: The lower limit is always adapted to the value specified in the current database.
- `*`: The value remains unchanged. This can be useful if several signals are selected to change their upper saturation limits.

Click Set to assign the setting to the selected signal(s).

Max Lets you enter the upper saturation limit for the selected signal(s). You can enter:

- A numerical value
- `fromDBC`: The upper limit is always adapted to the value specified in the current database.

- *: The value remains unchanged. This can be useful if several signals are selected to change their lower saturation limits.

Click Set to assign the setting to the selected signal(s).

Set Lets you assign the specified ranges to the message(s) selected from the User-defined ranges list.

Related topics

References

RTICANMM MainBlock.. 68

Signal Errors Page (RTICANMM MainBlock)

Access

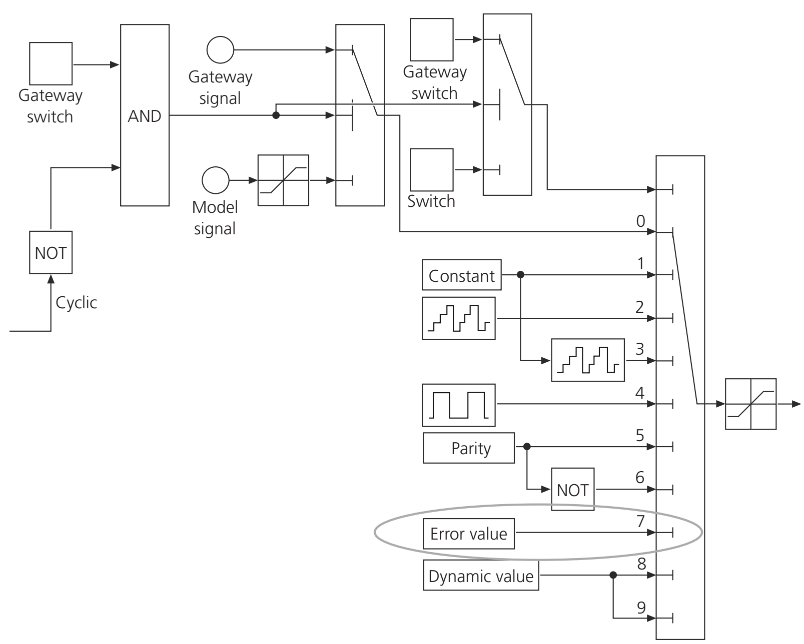
Located in TX Page (RTICANMM MainBlock) on page 214.

Purpose

To specify error values for signals to be transmitted.

Description

You can specify an error value for each signal to be transmitted. If you specify error values, there will be entries in the generated TRC file for each signal. The entries let you switch to transmitting the error value. The illustration below visualizes this switch.



Error value: You can either specify error values according to your requirements, or use error values according to the bit ranges defined in the database file. The error value of a signal is then the highest possible value in the signal's bit range if the signal's physical range is smaller. If the signal's bit range and the physical ranges are the same, there is no error value available according to the database file.

> **Note**
>
> - You cannot change the error value during run time.
> - The error values displayed on this page are always represented in physical units.

Example: Suppose an 8-bit signal has a scale factor of 2 and an offset of -10. The signal's value range is -10 … 500. Suppose that the signal's physical range is -10 … 400. The error value according to the bit ranges is then 500, which is the highest possible value within the signal's bit range and outside the physical range.

Dialog settings

No errors Lets you specify not to use error values for the signals to be transmitted. In that case, you cannot switch to transmitting an error value for any signal to be transmitted.

Errors from database Lets you set the error values of all signals to be transmitted according to the specification in the database file. This option is only sensible if you work with a database file. Refer to General Settings Page (RTICANMM MainBlock) on page 100.

User-defined errors Lets you set error values individually for each signal to be transmitted.

User-defined signal errors Lists the user-defined signal errors. The following commands are available via context menu:

- Set all to database defaults
- Set selected to database defaults
- Set all to value/name pairs end value
- Set selected to value/name pairs end value
- Configuration File

Set all to database defaults (Available via context menu) Lets you set the error values of all signals to be transmitted according to the specification in the database file. If a signal has no error value specified in the database file, 'X' is displayed as its error value.

Set selected to database defaults (Available via context menu) Lets you set the error values of the selected signal(s) to be transmitted according to the specification in the database file. If a signal has no error value specified in the database file, 'X' is displayed as the error value.

Set all to value/name pairs end value (Available via context menu) Lets you set the error value for each signal to the physical value that represents the highest coded error value of the specified value/name pairs in the database file each. If a signal has no value/name pair specified in the database file, 'X' is displayed as the error value. This setting applies to all signals to be transmitted.

Set selected to value/name pairs end value (Available via context menu) Lets you set the error value for each signal to the physical value that represents the highest coded error value of the specified value/name pairs in the database file each. If a signal has no value/name pair specified in the database file, 'X' is displayed as the error value. This setting applies to the signals that are selected to be transmitted.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Error value Lets you specify an error value. You can enter:

- A numerical value.
- `X`, if you do not want to specify an error value for the selected signal(s). In that case, you cannot switch to transmitting an error value for the selected signal(s).
- `fromDBC`, if the error values specified in the current database are to be used.

Then click Set to assign the setting to the selected signal(s).

Set Lets you assign the specified error value to the message(s) selected from the User-defined signal errors list.

Related topics

Basics

Lesson 8: Customizing Signals (RTI CAN MultiMessage Blockset Tutorial 📖)

References

Signal Mappings Page (RTICANMM MainBlock)

Access

Located in TX Page (RTICANMM MainBlock) on page 214.

Purpose

To specify the mapping structure for TX model signals.

Description

When you select standard TX signals as TX model signals on the Model Signals (TX) Page (RTICANMM MainBlock) on page 214, the RTICANMM MainBlock gets the **TX Data** inport (see I/O characteristics on page 70). The inport lets you provide the values of TX signals from within the model.

- Signal mapping of the **TX Data** inport: The RTICANMM MainBlock has an internal mapping structure for the TX model signals provided via the **TX Data** inport. To connect signals to this inport correctly, you have to connect them according to the RTICANMM MainBlock's internal mapping structure. You can let the Mapping to RTICANMM Block be created automatically for this purpose (see Create mapping block to TX on page 262). This block allows you to easily connect model signals to the **TX Data** inport according to the RTICANMM MainBlock's internal mapping structure.
- Specifying your own signal mapping: On this page, you can specify your own mapping structure between TX model signals and model signals. Specify one model signal for each TX model signal.
  - If you specify one model signal for each TX model signal (complete mapping), no Mapping to RTICANMM Block is required. You can connect model signals to the RTICANMM MainBlock.
  - If you do not specify a model signal for each TX model signal (incomplete mapping), a Mapping to RTICANMM Block is created. This block internally connects the TX model signals for which no model signal is specified to Simulink Ground blocks. Connect model signals to the Mapping to RTICANMM Block.

Note

If mapping is used in the RTICANMM MainBlock, all errors involving incorrect structure will appear inside the RTICANMM MainBlock.

Tip

If you generated the Mapping Blocks with RTICANMM 1.3, you can modify the mappings of the RTICANMM MainBlock without having to create its S-functions again.

Dialog settings

No mapping Lets you specify not to use your own mapping structure.

User-defined mapping Lets you specify your own mapping structure between model signals and TX model signals. Enter the model signal name, then click **Set** to assign the signal to the selected TX model signal.

Mapping direct to MainBlock (Available only if you specified a mapping structure for all TX model signals) If you select this checkbox, the RTICANMM MainBlock uses the specified mapping structure directly, without creating a Mapping to RTICANMM Block.

User-defined mappings Lists the user-defined mappings. The following commands are available via context menu:

- Select from Bus Selector
- Gateway (Selected)
- Gateway (All)
- Gateway ECU hierarchy (Selected)
- Gateway ECU hierarchy (All)
- Configuration File
- Renew mapping block

Select from Bus Selector (Available via context menu) Lets you specify your own mapping structure via an existing Simulink Bus Selector block. The structure of the selected Bus Selector block will be used to specify the mapping structure of the selected signal(s).

Gateway (Selected) (Available via context menu) Lets you gateway all the signals selected from the list.

Gateway (All) (Available via context menu) Lets you gateway all the signals of the list.

Gateway ECU hierarchy (Selected) (Available via context menu) Lets you specify an ECU-oriented mapping structure to trigger the transmission of the selected TX model signals.

If another RTICANMM MainBlock connected to the current RTICANMM MainBlock provides an output port bus structure with an ECU hierarchy, you can map the selected TX model signals to this bus structure (see also Peripheral Options Page (RTICANMM MainBlock) on page 262).

Gateway ECU hierarchy (All) (Available via context menu) Lets you specify an ECU-oriented mapping structure to trigger the transmission of all the TX model signals displayed in the **User-defined mapping** list.

If another RTICANMM MainBlock connected to the current RTICANMM MainBlock provides an output port bus structure with an ECU hierarchy, you can map all the TX model signals to this bus structure (see also Peripheral Options Page (RTICANMM MainBlock) on page 262).

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Renew mapping block (Available via context menu) Lets you quickly connect your application to the RTICANMM MainBlock after you modified the block's mappings. There is no need to create the whole MainBlock after modifying mappings. To create the MainBlock nevertheless, select **Force "Create"** from the Options menu.

Mapping structure Lets you specify the mapping structure for the TX model signal selected from the **User-defined mappings** list. Then click **Set**.

Set Lets you assign the specified mapping structure to the TX model signals(s) selected from the **User-defined mappings** list.

Macros Opens a dialog that informs you which macros are available for defining a mapping structure. The following macros are available:

- %CAN
- %Controller
- %ECU
- %ReceiveECU
- %MsgName
- %MsgID
- %SigName

Related topics

Basics

Lesson 5: Working with Model-Specific Bus Hierarchies (RTI CAN MultiMessage Blockset Tutorial 📖)

References

Signal Manipulation Page (RTICANMM MainBlock)

Access

Located in TX Page (RTICANMM MainBlock) on page 214.

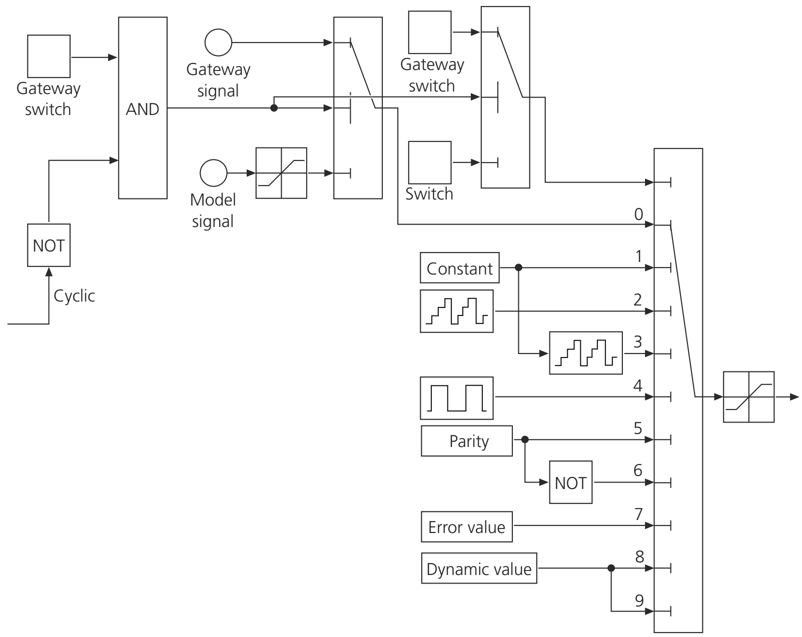
Purpose

To view the different signal manipulation options.

Description

You have several options to manipulate the values of signals to be transmitted. Depending on which options you specify, you can switch between them via corresponding entries in the generated TRC file.

The illustration below visualizes the options.



Default signal manipulation You can set a default signal manipulation option for signals and gateway signals to be transmitted. Refer to:

- Signal Default Manipulation Page (refer to Signal Default Manipulation Page (RTICANMM MainBlock) on page 244)
- Gateway Signal Default Manipulation Page (refer to Signal Default Manipulation Page (Gateway) (RTICANMM MainBlock) on page 251)

Dialog pages

To specify the different signal manipulation options, refer to the following pages:

- Custom Signal Manipulation Page (RTICANMM MainBlock) on page 231
- Dynamic Signal Page (RTICANMM MainBlock) on page 233
- Toggle Page (RTICANMM MainBlock) on page 236
- Parity Page (RTICANMM MainBlock) on page 238
- Counter Page (RTICANMM MainBlock) on page 240

Related topics

References

RTICANMM MainBlock.. 68

Custom Signal Manipulation Page (RTICANMM MainBlock)

Access

Located in Signal Manipulation Page (RTICANMM MainBlock) on page 230.

Purpose

To specify a custom signal manipulation option.

Description

You can implement a custom signal manipulation option. This allows you to manipulate the physical values of signals to be transmitted before they are converted into the raw data of the TX message. You have to specify your custom signal manipulation option in a header file.

Dialog settings

Header file Lets you specify a header file to be included. Click [h] to browse to the file.

Use relative path Lets you set the folder relative to the model root folder or as an absolute path.

Tip

Using a relative path allows you to easily move the model and all the files belonging to it to another location on your file system without having to change any path settings.

Edit Header File Lets you edit the selected header file in MATLAB's M-File Editor.

Available TX messages Lets you specify the TX messages to be manipulated according to the specified header file.

Selected TX messages Lists the selected messages. You can sort the selected messages according to your needs via buttons.

Create Header File Lets you create a header file according to the settings you defined. You can access all global variables via the created header file, which allows you to work with all the signals of the specified TX messages.

Note

The signal names used in the header file do not correspond to the signal names in the database file. Variable names are used instead, which allows you to use the header file for other RTICANMM MainBlocks referencing the same TX messages.
During code generation, the signals will automatically be renamed according to the *correct* names of the real-time variables.

Related topics

References

RTICANMM MainBlock.. 68

Dynamic Signal Page (RTICANMM MainBlock)

Dialog pages

You can specify the settings for dynamic signals on the following pages:

- Dynamic Signal Values Page (RTICANMM MainBlock) on page 233
- Dynamic Signal Defaults Page (RTICANMM MainBlock) on page 235

Related topics

References

Dynamic Signal Values Page (RTICANMM MainBlock)

Access

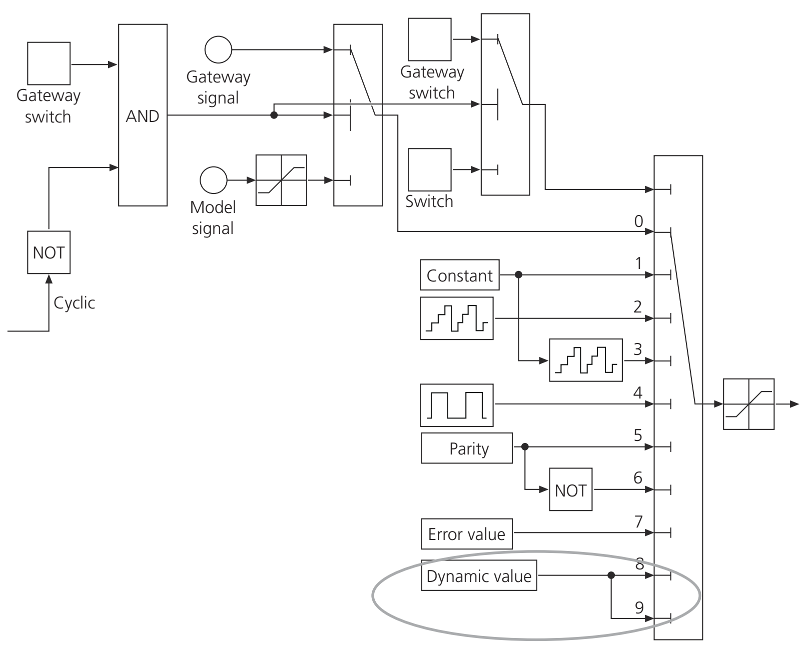
Located in Dynamic Signal Page (RTICANMM MainBlock) on page 233.

Purpose

To specify the use of dynamic values for signals to be transmitted.

Description

You can specify to use a dynamic value for each signal to be transmitted. If you specify a dynamic value for a signal, there will be an entry in the generated TRC file that lets you switch to transmitting the dynamic value. The illustration below visualizes this switch.



- *Dynamic value and countdown value*: If you switch to the dynamic value of a signal, this value is transmitted for a defined number of times. This number is the countdown value. Then signal manipulation automatically switches back to the signal manipulation option used before the dynamic value.

  For example, suppose you specify a dynamic value of 8 and a countdown value of 3. If you switch to the dynamic value of this signal, the signal value 8 is sent the next 3 times the TX message is transmitted. Then the signal manipulation option is reset.

  The generated TRC file gets corresponding entries for the dynamic value and the countdown value. You can specify default values on the Dynamic Signal Defaults Page (RTICANMM MainBlock).
- *Dynamic value of counter signals*: For a signal you specify as a counter signal (see Counter Page (RTICANMM MainBlock) on page 240), the dynamic value can represent either the counter increment value or the transmitted counter signal value, i.e., a constant value is transmitted. After the message was transmitted the defined number of times, the counter increment value is reset to its old value and the counter continues to count from the original value.

Tip

You can specify the default manipulation option (for example, dynamic or constant signal value) on the Signal Default Manipulation Page (RTICANMM MainBlock) on page 244.

Dialog settings

All Lets you enable the use of dynamic values for all signals to be transmitted.

None Lets you disable the use of dynamic values for all signals to be transmitted.

Selected Lets you enable the use of dynamic values for a set of signals to be transmitted.

Available signals Displays the signals that are available in the TX messages. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select signals in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

Selected signals Displays the selection of signals that is active if you select the Selected option. The following commands are available via context menu:

- All
- None

All (Available via context menu) Lets you add all the available signals to the list.

None (Available via context menu) Lets you remove all signals from the list.

Related topics

Basics

Lesson 8: Customizing Signals (RTI CAN MultiMessage Blockset Tutorial 📖)

References

Dynamic Signal Defaults Page (RTICANMM MainBlock)

Access

Located in Dynamic Signal Page (RTICANMM MainBlock) on page 233.

Purpose

To specify default dynamic values for signals to be transmitted.

Description

You can specify a default dynamic value and a default countdown value for each signal to be transmitted. For details on dynamic values and countdown values, refer to Dynamic Signal Values Page (RTICANMM MainBlock) on page 233.

Dialog settings

All defaults to zero Lets you specify the default dynamic values for all signals to zero.

User-defined defaults Lets you specify default dynamic values individually for each signal. The list displays the signals available for user-defined defaults. You can create or load a configuration file or add these page settings to an existing configuration file via context menu.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Value Lets you specify the default dynamic value. Click **Set** to assign the value to the selected signal(s).

Countdown Lets you specify the countdown value. Click **Set** to assign the value to the selected signal(s).

Set Lets you assign the specified value and countdown to the signal(s) selected in the **User-defined defaults** list.

Related topics

References

Toggle Page (RTICANMM MainBlock)

Access

Located in Signal Manipulation Page (RTICANMM MainBlock) on page 230.

Purpose

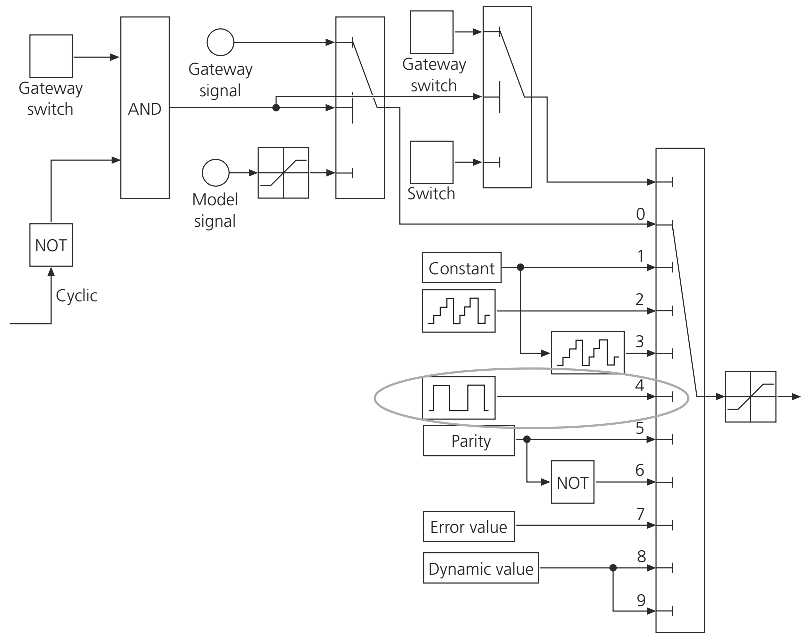
To specify toggle signals.

Description

You can specify one or more signals as toggle signals for each TX message.

- *Toggle signal*: A toggle signal is a 1-bit signal that you can use, for example, to indicate whether CAN messages are transmitted:
  - If CAN messages are transmitted, the toggle signal value alternates between 0 and 1.
  - If CAN messages are not transmitted, the toggle signal value remains constant.
- *Toggle period*: The toggle period specifies the sample period for checking CAN message transmission.

If you specify a toggle signal, there will be an entry in the generated TRC file that lets you switch to transmitting the toggle signal value. The illustration below visualizes this switch.



Tip

You can specify the default manipulation option (for example, toggle or constant signal value) on the Signal Default Manipulation Page (RTICANMM MainBlock) on page 244.

Dialog settings

Message Lets you select a TX message to specify a toggle signal for.

Signal Lets you select a signal contained in the selected TX message. Only 1-bit signals are available.

Period [s] Lets you specify the toggle period in seconds.

Note

Do not specify the cycle time of the corresponding TX message as an integer multiple of the toggle period.

Set Lets you add the selected signal to the list of toggle signals.

Edit Lets you read the settings for the signal selected from the list of toggle signals.

Delete Lets you delete the selected signal(s) from the list of toggle signals.

Filter Lets you filter and select signals in the Filter Dialog (RTICANMM MainBlock) on page 91.

User-defined toggle signals Lists the user-defined toggle signals. You can create or load a configuration file or add these page settings to an existing configuration file via context menu.

> **Tip**
>
> You can select multiple signals by pressing **Shift** or **Ctrl**.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Related topics

References

RTICANMM MainBlock.. 68

Parity Page (RTICANMM MainBlock)

Access

Located in Signal Manipulation Page (RTICANMM MainBlock) on page 230.

Purpose

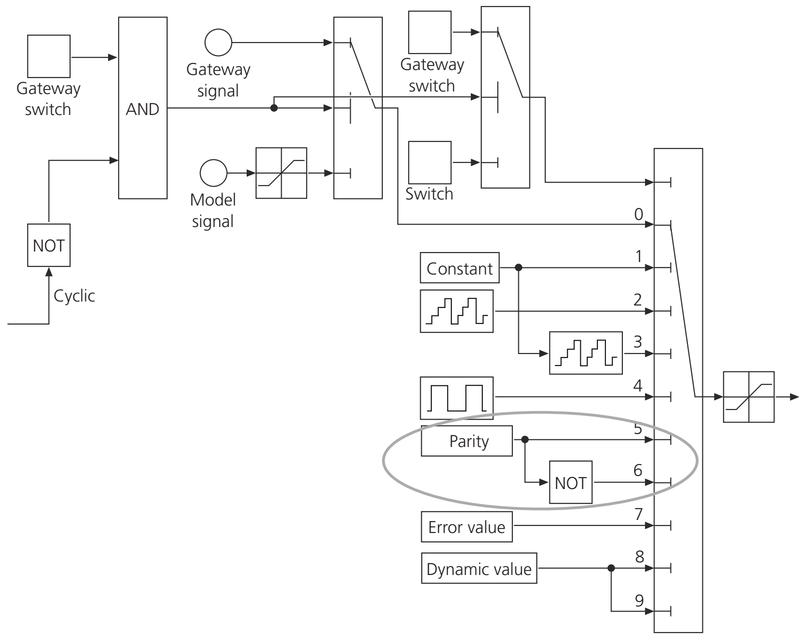
To specify parity signals.

Description

Within a TX message, you can specify signals whose parity you want to calculate. You have to specify a signal as the parity signal.

Parity signal: A parity bit is calculated for the specified signals according to whether even or odd parity is selected. This bit is appended to the parity signal. Then the TX message with the parity signal is transmitted. You can change the parity – even or odd – during run time.

If you specify a parity signal, there will be an entry in the generated TRC file that lets you switch to transmitting the parity signal value. The illustration below visualizes the switch.



Tip

You can specify the default manipulation option (for example, parity or constant signal value) on the Signal Default Manipulation Page (RTICANMM MainBlock) on page 244.

Dialog settings

Message Lets you select a TX message to specify a parity signal for.

Odd parity Lets you specify odd parity. In this case, the parity bit is set so that there is an odd number of "1" bits for the selected signal(s), including the parity bit, for example:

- {0011100,0} [signal value, parity bit]
- {0011000,1} [signal value, parity bit]

Even parity Lets you specify even parity. In this case, the parity bit is set so that there is an even number of "1" bits for the selected signal(s), including the parity bit, for example:

- {0011000,0} [signal value, parity bit]
- {0011100,1} [signal value, parity bit]

Parity signal Lets you select a signal contained in the selected TX message and specify it as a parity signal.

Signals Lets you select one or more signals of the TX message for which you want to calculate the parity. You can select all the signals of the message except for the parity signal itself.

Add Lets you add the selected signal to the list of parity signals, or – if the parity signal already is on the list – apply a change.

Edit Lets you read the settings for the parity signal selected from the list of parity signals.

Delete Lets you delete the selected parity signal from the list of parity signals.

User-defined parity signals Lists the user-defined parity signals. You can create or load a configuration file or add these page settings to an existing configuration file via context menu.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Related topics

References

Counter Page (RTICANMM MainBlock)

Access

Located in Signal Manipulation Page (RTICANMM MainBlock) on page 230.

Purpose

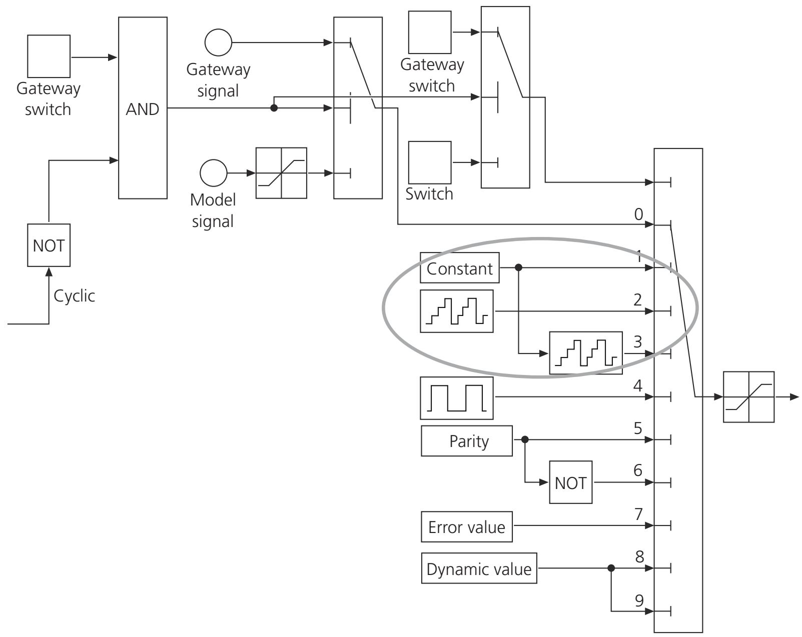
To specify counter signals.

Description

TX messages are counted when they are transmitted. RX messages are counted when they are received. Using the counter in RX mode lets you compare the received counter value to the value of the counter. If the values differ, a counter error occurs (refer to RX Error Ports Page (RTICANMM MainBlock) on page 184).

Counter signal: For each message, you can specify one or more counter signals that provide the number of message transmissions/receptions. You can specify the counter start value, the increment, and the counter stop value. Each time the counter reaches the stop value, it turns around to the counter start value.

If you specify a counter signal, there will be an entry in the generated TRC file that lets you switch to transmitting the counter signal value. You can also set the increment to the constant value during run time via this entry. The illustration below visualizes the switch.



Tip

You can specify the default manipulation option (for example, counter or constant signal value) on the Signal Default Manipulation Page (RTICANMM MainBlock) on page 244.

Dialog settings

Message Lets you specify a TX or RX message to be counted.

Mode signals Lets you filter the mode signals of all TX messages.

Signal Lets you specify the counter signal.

- If you select **Message** (see above), all the signals of the selected TX or RX message are available in the drop-down list.
- If you select **Mode signals** (see above), the mode signals of all TX messages are available in the drop-down list.

Start Lets you specify the counter start value.

Step Lets you specify the increment of the counter.

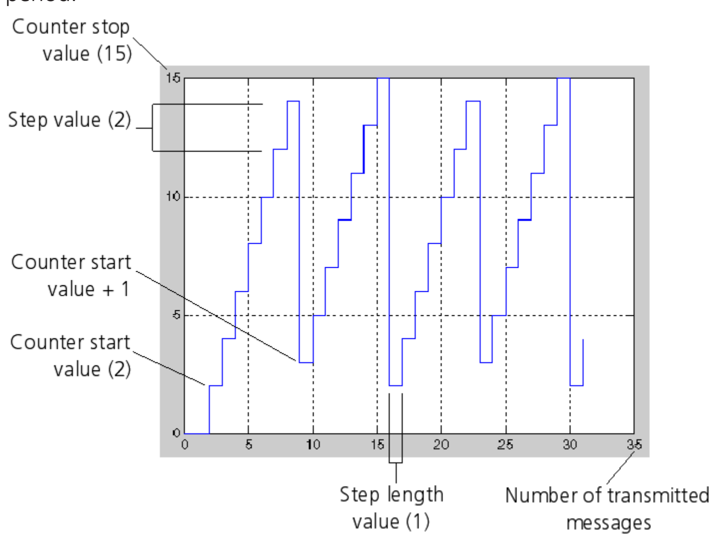
Divisor Lets you specify the step length of the counter. If you specify a divisor value of 3, for example, the counter value remains constant for 3 message transmissions/receptions. After the next message transmission/reception, the counter value is incremented by the **Step** value. Additionally, the **Start**, **Step**, **Step Length**, and **Stop** parameters are divided by the divisor's value.

> **Note**
>
> The Divisor parameter is provided only for compatibility reasons. Specify a divisor value of 1 and work with the Step length instead.

Step length Lets you specify the step length of the counter. If you specify a step length of 3, for example, the counter value remains constant for 3 message transmissions/receptions. After the next message transmission/reception, the counter value is incremented by the Step value.

> **Note**
>
> If you work with the Step length, specify a Divisor value of 1.

Stop Lets you specify the counter stop value. The counter value exceeds the stop value periodically: Suppose you specify Stop = 15 and Step = 2. The counter turns around to (counter start value + 1) when it reaches 14 (first period). For the second period, the counter reaches the counter stop value and turns around to the counter start value. The cycle starts again with the first period.



Behavior in Const. Mode Lets you specify the counter behavior if the signal is not transmitted/received with the counter but with another signal type such as a constant or toggle value.

Setting	Description
Continue Counting	The counter continues counting whether or not it is transmitted/received. When a different signal type is transmitted/received, the counter is incremented internally each time the other signal type is transmitted/received. The next time the counter is transmitted/received, it starts counting from the internally incremented counter value.

Setting	Description
Stop Counting	When a signal type other than the counter is transmitted/received, the counter stops counting. The next time the counter is transmitted/received, it starts counting from the previous counter value.
Set to Constant	When a signal type other than the counter is transmitted/received, the counter stops counting. The next time the counter is transmitted/received, it starts counting from the previously received signal value, regardless of the signal type.

Note

If you switch from a gateway signal value to the counter value, the counter always continues counting in "Set to Constant" mode.

Tip

Additionally, you can specify the counter behavior during a timeout of a TX message, refer to TX Timeout Enable Page (RTICANMM MainBlock) on page 161.

Show Lets you plot the specified counter signal graphically.

Set Lets you add the selected signal to the list of counter signals, or – if the selected signal already is on the list – apply a change.

Delete Lets you delete the signal selected from the list of counter signals.

Filter Lets you filter and select signals in the Filter Dialog (RTICANMM MainBlock) on page 91.

User-defined counter signals Lists the user-defined counter signals. You can create or load a configuration file or add these page settings to an existing configuration file via context menu.

Tip

You can select multiple signals by pressing `Shift` or `Ctrl`.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Related topics

Basics

Lesson 6: Performing a CAN Restbus Simulation (RTI CAN MultiMessage Blockset Tutorial 📖)
Lesson 8: Customizing Signals (RTI CAN MultiMessage Blockset Tutorial 📖)

References

RTICANMM MainBlock.. 68

Signal Default Manipulation Page (RTICANMM MainBlock)

Access

Located in TX Page (RTICANMM MainBlock) on page 214.

Purpose

To set the default signal manipulation option of signals to be transmitted (TX signals).

Description

You can define a default signal manipulation option for each signal to be transmitted.

Dialog settings

All to default Lets you set the default manipulation option of all signals to be transmitted. The first applicable default from the following sequence is used:

- Input
- Counter
- Toggle
- Parity
- Constant
- Error
- Dynamic Value

User-defined defaults Lets you set the default signal manipulation option individually for each signal to be transmitted. The list displays the user-defined defaults. The following commands are available via context menu:

- Set All to Default
- Set All to Constant
- Set All Inputs to Input
- Set All Counters to Counter
- Set All Parity to Parity
- Set All Toggle to Toggle

- Set All Errors to Error
- Configuration File

Set All to Default (Available via context menu) Lets you set the default manipulation option of all signals to be transmitted.The first applicable default from the following sequence is used:

- Input
- Counter
- Toggle
- Parity
- Constant
- Error
- Dynamic Value

Set All to Constant (Available via context menu) Lets you set the default signal manipulation option of all signals to be transmitted to Constant.

For input signals for which you did not enable input manipulation on the Input Manipulation Page (RTICANMM MainBlock) on page 216, the default signal manipulation option is set to Input.

Set All Inputs to Input (Available via context menu) Lets you set the default manipulation option to Input for all the signals that are configured as input signals on the Model Signals (TX) Page (RTICANMM MainBlock) on page 214.

Set All Counters to Counter (Available via context menu) Lets you set the default manipulation option to Counter for all the signals you enabled as counter signals on the Counter Page (RTICANMM MainBlock) on page 240.

Set All Parities to Parity (Available via context menu) Lets you set the default manipulation option to Parity for all the signals you enabled as parity signals on the Parity Page (RTICANMM MainBlock) on page 238.

Set All Toggles to Toggle (Available via context menu) Lets you set the default manipulation option to Toggle for all the signals you enabled as toggle signals on the Toggle Page (RTICANMM MainBlock) on page 236.

Set All Errors to Error (Available via context menu) Lets you set the default manipulation option to Error for all the signals for which you specified an error value on the Signal Errors Page (RTICANMM MainBlock) on page 225.

Configuration File - Create (Available via context menu) Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.

Configuration File - Add (Available via context menu) Lets you add the settings on the current page to an existing configuration file.

Configuration File - Load (Available via context menu) Lets you load the settings of a configuration file to the current page.

Default Manipulation Lets you specify the default manipulation to the signal(s) selected from the list of User-defined defaults. The following signal manipulation options are available depending on the signal manipulation option you enabled:

Signal Manipulation Option	Option is Available if ...
Constant	Always available except for the following case: The signal is specified as a TX model signal on the Model Signals (TX) Page (RTICANMM MainBlock) on page 214, and input manipulation is disabled for the signal on the Input Manipulation Page (RTICANMM MainBlock) on page 216.
Constant/Gateway	The signal is not specified as a TX model signal on the Model Signals (TX) Page (RTICANMM MainBlock) on page 214, and a gateway signal is specified for the signal on the Gateway Signals Page (RTICANMM MainBlock) on page 248.
Input	The signal is specified as a TX model signal on the Model Signals (TX) Page (RTICANMM MainBlock) on page 214.
Input/Gateway	The signal is specified as a TX model signal on the Model Signals (TX) Page (RTICANMM MainBlock) on page 214, and a gateway signal is specified for the signal on the Gateway Signals Page (RTICANMM MainBlock) on page 248.
Error	An error value is specified for the signal on the Signal Errors Page (RTICANMM MainBlock) on page 225.
Toggle	The signal is specified as a toggle signal on the Toggle Page (RTICANMM MainBlock) on page 236.
Parity	The signal is specified as a parity signal on the Parity Page (RTICANMM MainBlock) on page 238.
Counter / IncrementCounter	The signal is specified as a counter signal on the Counter Page (RTICANMM MainBlock) on page 240.
DynValue	The use of dynamic values is specified for the signal on the Dynamic Signal Values Page (RTICANMM MainBlock) on page 233.
Global time synchronization	The signal data is read from the relevant time base manager instance.

Set Lets you assign the specified default manipulation to the signal(s) selected from the list of **User-defined defaults**.

Related topics

Basics

Lesson 8: Customizing Signals (RTI CAN MultiMessage Blockset Tutorial 📖)

References

RTICANMM MainBlock.. 68

Model Signals (RX) Page (RTICANMM MainBlock)

Access

Located in Signals Page (RTICANMM MainBlock) on page 213.

Purpose

To specify RX signals as RX outport signals (RX model signals).

Description

You can analyze received signals (RX signals). The generated TRC file gets an entry for each signal in an RX message for this purpose. This allows you to analyze the RX signal with the **Bus Navigator** of ControlDesk.

RX model signals: You can also analyze RX signals in the model. You have to specify them as RX model signals for this purpose.

- The RTICANMM MainBlock gets a corresponding **RX Data** outport (see I/O characteristics on page 70) that allows you to analyze RX model signals in the model. Connecting a Simulink Bus Selector block to this outport gives you easy access to all RX model signals.
- The generated TRC file also gets entries for signals specified as RX model signals.

Note

For optimum performance, you should specify as few RX model signals as possible.

Dialog settings

All Lets you specify all RX signals as RX model signals.

None Lets you select none of the available RX signals as RX model signals.

Selected Lets you specify a set of the available RX signals as RX model signals.

Available signals Displays the signals that are available in the RX messages. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select signals in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically **Up** or **Down**.

Selected signals Displays the selection of RX model signals that is active if you select the **Selected** option. The following commands are available via context menu:

- All
- None

All (Available via context menu) Lets you add all the available signals to the list.

None (Available via context menu) Lets you remove all signals from the list.

Related topics

References

Gateway Page (RTICANMM MainBlock)

Access

Located in Signals Page (RTICANMM MainBlock) on page 213.

Dialog pages

You can specify the aspects of gatewaying on the following pages:

- Gateway Signals Page (RTICANMM MainBlock) on page 248
- Gateway Defaults Page (RTICANMM MainBlock) on page 250
- Signal Default Manipulation Page (Gateway) (RTICANMM MainBlock) on page 251

Related topics

References

Gateway Signals Page (RTICANMM MainBlock)

Access

Located in Gateway Page (RTICANMM MainBlock) on page 248.

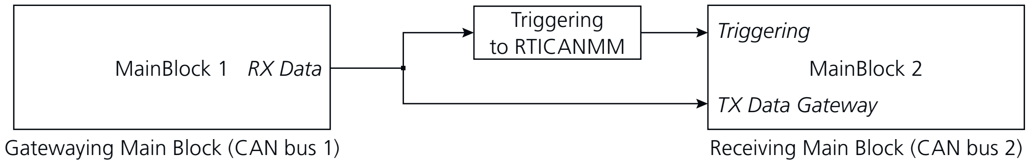
Purpose

To specify gateway signals for signals to be transmitted.

Description

Gateway signals can be manipulated before they are exchanged between two CAN buses. If you specify one or more gateway signals, the RTICANMM MainBlock gets the TX Data Gateway inport to connect gateway signals to the block (see I/O characteristics on page 70).

Example



MainBlock1 gateways messages and their signals to MainBlock2. Specifying gateway signals for MainBlock2 adds a **TX Data Gateway** inport to this block. The specified gateway signals will be transmitted via CAN bus 2 with the signal values received from MainBlock1. The transmission of the gateway signals is triggered by the messages received from MainBlock1. You therefore have to specify triggered message transmission for the messages of the gateway signals on the Message Cyclic Page (RTICANMM MainBlock) on page 144.

Manipulation of gateway signals Gateway signals can be manipulated like normal signals. You can specify counter, toggle, parity, or dynamic signal values, for example. If a gateway signal is not manipulated but transmitted with the value received from the gatewaying MainBlock, it is always a constant signal. If you manipulate gateway signals, you can specify the default manipulation option on the Signal Default Manipulation Page (Gateway) (RTICANMM MainBlock) on page 251.

Note

Implementing gateway signals at least doubles the number of block inports and therefore reduces performance. If you want to exchange messages between CAN buses and do not want to perform signal manipulation, you should use the RTICANMM Gateway on page 280 block instead.

Dialog settings

Option Lets you specify the gateway signals.

- **All TX signals from model** Lets you specify gateway signals for all signals of all TX messages.
- **No gateway signals from model** Lets you specify gateway signals for none of the signals of TX messages.
- **Gateway signal same as input signals** Lets you specify gateway signals for the signals you specified as TX signals on the Model Signals (TX) Page (RTICANMM MainBlock) on page 214.
- **Selected gateway signals from model** Lets you specify gateway signals for a set of the signals of TX messages.

Available signals Displays all the signals that are available in the database file. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select signals in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

Selected signals Displays the selection of signals that is active if you select the Selected option. The following commands are available via context menu:
- All TX Signals from Model
- No Gateway Signals from Model
- Gateway Signals Same as Input Signals
- Gateway Signals Inverse to Input Signals

Related topics

Basics

Lesson 12 (Advanced): Gatewaying Messages (RTI CAN MultiMessage Blockset Tutorial 📖)

References

Gateway Defaults Page (RTICANMM MainBlock)

Access

Located in Gateway Page (RTICANMM MainBlock) on page 248.

Purpose

To specify the default gateway switch setting.

Description

The generated TRC file has a gateway signal switch entry for each signal for which you specify a gateway signal. You can specify the default setting of the entry (active or inactive).

Dialog settings

All active Lets you set the default gateway switch setting of all gateway signals to active.

All inactive Lets you set the default gateway switch setting of all gateway signals to inactive.

Selected active Lets you set the default gateway switch setting of a set of gateway signals to active.

Available signals Displays all the signals that are available in the database file. The following commands are available via context menu:

- Filter
- Sort

Filter (Available via context menu) Lets you filter and select signals in the Filter Dialog (RTICANMM MainBlock) on page 91.

Sort (Available via context menu) Lets you sort the list alphabetically Up or Down.

Selected signals Displays the selection of signals that is active if you select the Selected option.

Related topics

References

Signal Default Manipulation Page (Gateway) (RTICANMM MainBlock)

Access

Located on Gateway Page (RTICANMM MainBlock) on page 248.

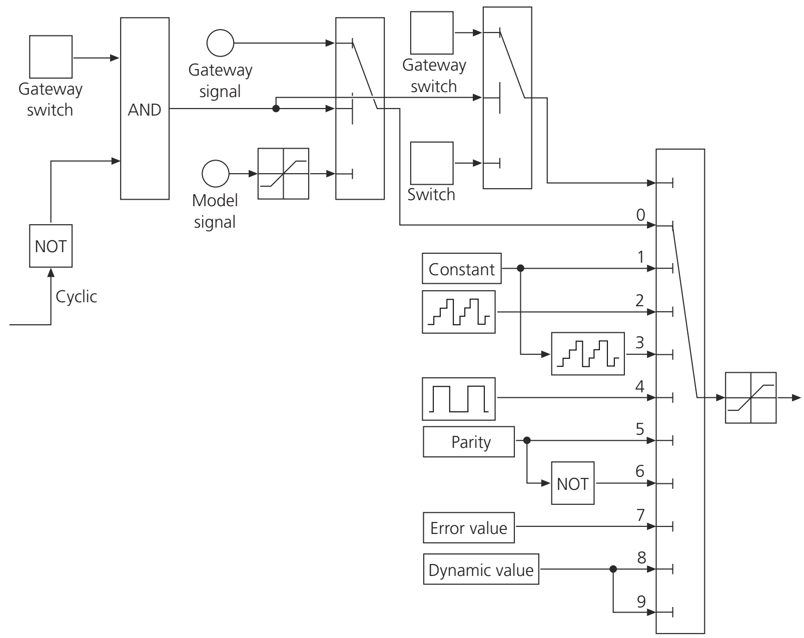
Purpose

To set the default signal manipulation option for gateway signals to be transmitted.

Description

You can define a default signal manipulation option for each gateway signal to be transmitted. This manipulation will be active during the gateway of the signal (Message: triggered and gateway option: auto). If the signal is not gatewayed (Message: cyclic or gateway: off), the standard manipulation option will be used.

The illustration below visualizes the options.



Dialog settings

All to default Lets you set the default manipulation option of all gateway signals to be transmitted. The first applicable default from the following sequence is used:

- Input
- Counter
- Toggle
- Parity
- Constant
- Error
- Dynamic Value

Enable Manipulation Lets you enable default manipulation.

User-defined defaults Lets you set the default signal manipulation option individually for each gateway signal to be transmitted. The list displays the user-defined defaults. The following commands are available via context menu:

- Set All to Default Lets you set the default manipulation option of all gateway signals to be transmitted. The first applicable default from the following sequence is used:
  - Input
  - Counter
  - Toggle
  - Parity
  - Constant
  - Error
  - Dynamic Value

- **Set All to Constant** Lets you set the default signal manipulation option for all gateway signals to be transmitted to **Constant**.

  If you did not enable input manipulation for specific signals on the Input Manipulation Page (RTICANMM MainBlock) on page 216, their default signal manipulation option is set to **Input**.
- **Set All Inputs to Input** Lets you set the default manipulation option to **Input** for all the gateway signals that are configured as input signals on the Model Signals (TX) Page (RTICANMM MainBlock) on page 214
- **Set All Counters to Counter** Lets you set the default manipulation option to **Counter** for all the gateway signals you enabled as counter signals on the Counter Page (RTICANMM MainBlock) on page 240.
- **Set All Parities to Parity** Lets you set the default manipulation option to **Parity** for all the gateway signals you enabled as parity signals on the Parity Page (RTICANMM MainBlock) on page 238.
- **Set All Toggles to Toggle** Lets you set the default manipulation option to **Toggle** for all the gateway signals you enabled as toggle signals on the Toggle Page (RTICANMM MainBlock) on page 236.
- **Set All Errors to Error** Lets you set the default manipulation option to **Error** for all the gateway signals for which you specified an error value on the Signal Errors Page (RTICANMM MainBlock) on page 225.
- **Configuration File – Create** Lets you create a new configuration file from the settings on the current page. For details, refer to Configuration File on page 80.
- **Configuration File – Add** Lets you add the settings on the current page to an existing configuration file.
- **Configuration File – Load** Lets you load the settings of a configuration file to the current page.

Default Manipulation Lets you specify the default manipulation to the gateway signal(s) selected from the list of **User-defined defaults**. The following signal manipulation options are available depending on the signal manipulation option you enabled:

Signal Manipulation Option	Option is Available if ...
Constant	Always available except for the following case: The signal is specified as a TX model signal on the Model Signals (TX) Page (RTICANMM MainBlock) on page 214, and input manipulation is disabled for the signal on the Input Manipulation Page (RTICANMM MainBlock) on page 216.
Constant/Gateway	The signal is not specified as a TX model signal on the Model Signals (TX) Page (RTICANMM MainBlock) on page 214, and a gateway signal is specified for the signal on the Gateway Signals Page (RTICANMM MainBlock) on page 248.
Input	The signal is specified as a TX model Signal on the Model Signals (TX) Page (RTICANMM MainBlock) on page 214.
Input/Gateway	The signal is specified as a TX model signal on the Model Signals (TX) Page (RTICANMM MainBlock) on page 214, and a gateway signal is specified for the signal on the Gateway Signals Page (RTICANMM MainBlock) on page 248.

Signal Manipulation Option	Option is Available if ...
Error	An error value is specified for the signal on the Signal Errors Page (RTICANMM MainBlock) on page 225.
Toggle	The signal is specified as a toggle signal on the Toggle Page (RTICANMM MainBlock) on page 236.
Parity	The signal is specified as a parity signal on the Parity Page (RTICANMM MainBlock) on page 238.
Counter / IncrementCounter	The signal is specified as a counter signal on the Counter Page (RTICANMM MainBlock) on page 240.
DynValue	The use of dynamic values is specified for the signal on the Dynamic Signal Values Page (RTICANMM MainBlock) on page 233.

Set Lets you assign the specified default manipulation to the signal(s) selected from the list of **User-defined defaults**.

Related topics

References

Includes Page (RTICANMM MainBlock)

Access

Located on the top level of the dialog tree.

Purpose

To specify header files to be included in the build process.

Description

This page allows you to specify header files you want to include in the build process. For example, if you want to include your own checksum algorithm, you should specify the corresponding header file on this page. The header files you specify on this page will be included at the beginning of the generated code.

Dialog settings

Define files to be included Lets you specify a header file to be included. Click [browse button] to browse to the file, then click **Set** to add the file to the list of files to be included.

Use relative path Lets you set the folder relative to the model root folder or as an absolute path.

Tip

Using a relative path allows you to easily move the model and all the files belonging to it to another location on your file system without having to change any path settings.

Set Lets you add the selected file to the list of files to be included.

Delete Lets you delete the selected file(s) from the list of files to be included.

Include files Lists the specified include files.

Related topics

References

RTICANMM MainBlock.. 68

General Options Page (RTICANMM MainBlock)

Access

Located on the top level of the dialog tree.

Dialog pages

You can specify the general settings on the following pages:

- Experimental Software Page (RTICANMM MainBlock) on page 255
- Code Options Page (RTICANMM MainBlock) on page 257
- TRC Page (RTICANMM MainBlock) on page 259
- Peripheral Options Page (RTICANMM MainBlock) on page 262

Related topics

References

RTICANMM MainBlock.. 68

Experimental Software Page (RTICANMM MainBlock)

Access

Located in General Options Page (RTICANMM MainBlock) on page 255.

Purpose

To enable CAN message handling in real-time testing and to specify the number of additional experimental messages.

Description

This page allows you to enable CAN message handling in real-time testing by specifying the number of additional experimental messages.

Experimental messages Experimental messages can be used for CAN replay and real-time testing. Experimental messages are independent of the database file and not included in the model's TRC file.

> **Tip**
>
> - For more information on CAN replay, refer to Monitoring and Logging Bus Communication (ControlDesk Bus Navigator 📖).
> - For more information on CAN message handling in real-time testing, refer to Basics on the rttlib.canlib Module (Real-Time Testing Guide 📖).

Transmitting experimental messages and activating CAN message handling in real-time testing To transmit experimental messages, you have to specify the number of experimental messages to be transmitted. This has to be done on this page. When you specify messages on this page, CAN message handling in real-time testing is activated automatically. You can specify up to 200 experimental messages to be transmitted for CAN replay/real-time testing.

> **Note**
>
> If you work with RTI-MP, you must additionally enable real-time testing in the Multiprocessor Setup block. Refer to Main Page (Multiprocessor Setup Dialog) (RTI and RTI-MP Implementation Reference 📖).

Dialog settings

CAN Replay and real-time testing CANLib Lets you specify the maximum number of experimental messages that can be transmitted during run time by CAN replay and real-time testing.

Related topics

References

RTICANMM MainBlock.. 68

Code Options Page (RTICANMM MainBlock)

Access

Located in General Options Page (RTICANMM MainBlock) on page 255.

Purpose

To specify options for S-function code for the RTICANMM MainBlock.

Dialog settings

No RX status for loopback messages Lets you specify the RX status for the reception of loopback messages. The RX status is provided via the RTICANMM MainBlock's RX Data outport if this outport is enabled (see RX Status and Time Ports Page (RTICANMM MainBlock) on page 173).

- If selected, the RX status remains `0` whenever loopback messages are received.
- If not selected, the RX status is set to `1` whenever loopback messages are received.

Use specific data types Lets you specify the data types of the RTICANMM MainBlock's inports and outports that are struct arrays.

- If the checkbox is selected, the Simulink data types of the RTICANMM MainBlock's inport and outport struct arrays are set according to the data types calculated from the database's signal ranges (refer to Signal Ranges Page (RTICANMM MainBlock) on page 223).
- If the checkbox is not selected, the Simulink data type of the RTICANMM MainBlock's inport and outport struct arrays is set to double.

For information on the RTICANMM MainBlock's inports and outports, refer to I/O characteristics on page 70.

Enable log view Lets you specify whether all the received messages are written to the log file with their raw data.

> **Note**
>
> Too many messages, received at too great a speed, can make the log view unreadable.

Enable "Reset" - TRC Lets you reset all the RTICANMM MainBlock settings to their default values during run time. If selected, RTICANMM generates the TRC file entry `Reset CAN`. When you change its value from `0` to `1`, all the settings of the RTICANMM MainBlock are reset.

Enable "Reset" - Inport Lets you reset all the RTICANMM MainBlock settings to their default values during run time. If selected, an Reset inport is generated for the RTICANMM MainBlock. You can connect the inport to a Simulink Constant block, etc.

Reset cycle time by kickout Lets you specify whether the cycle time of a TX message is reset by kickout.

- If the checkbox is selected, the cycle time is reset and the cyclic message transmission starts again with a new cycle after a message kickout.

- If the checkbox is not selected, the message transmission continues with the current cycle after a message kickout.

Kickout is active if kickout greater than 0 Lets you specify the kickout behavior depending on the value.

- If the checkbox is selected, the kickout is active if the value is greater than 0.
- If the checkbox is not selected, the kickout is active if the value is greater than the value before.

No feedthrough for TX input ports Lets you disable direct feed-through for the TX input ports of the RTICANMM MainBlock in your model in order to avoid algebraic loops. Algebraic loops occur when an input port with direct feed-through is driven by the output of the same block (directly or through other blocks with direct feed-through enabled). For example, suppose you work with two RTICANMM MainBlocks. MainBlock 1 receives data and transmits it to MainBlock 2, which in turn sends the received data to MainBlock 1.

- If the checkbox is selected, direct feed-through is disabled for the TX input ports. As a consequence, there are no algebraic loop occurrences between RTICANMM MainBlocks that are connected in a loop.

  Note

  You should select this option only if your model really contains algebraic loops between RTICANMM MainBlocks, because disabling direct feed-through affects the execution order of the blocks in your model. If the option is selected, the messages are transmitted by the `mdlUpdate` function instead of by the `mdlOutputs` function of the RTICANMM MainBlock S-function. For information on the execution order of S-functions, refer to the Simulink user documentation.

- If the checkbox is cleared, direct feed-through is enabled for the TX input ports.

Suppress Extended Data Page (EDP) bit for J1939 Lets you specify how to handle the extended data page bit of the CAN message identifier for J1939.

- If the checkbox is selected, the extended data page bit is suppressed. The RTI CAN MultiMessage Blockset can receive J1939 messages with the extended data page bit set to 0 or 1.
- If the checkbox is cleared, the extended data page bit is not suppressed. As a consequence, the RTI CAN MultiMessage Blockset only receives messages whose extended data page bit is set to 0. Messages whose extended data page bit is set to 1 are ignored.

By default, the checkbox is cleared. This setting corresponds to the earlier behavior of the RTI CAN MultiMessage Blockset, where the former CAN message identifier definition with the Reserved bit instead of the extended data page bit was used.

Related topics

References

RTICANMM MainBlock 68

TRC Page (RTICANMM MainBlock)

Access

Located in General Options Page (RTICANMM MainBlock) on page 255.

Dialog pages

You can specify the TRC options and extras on the following pages:

- TRC Options Page (RTICANMM MainBlock) on page 259
- TRC Extras Page (RTICANMM MainBlock) on page 261

Related topics

References

RTICANMM MainBlock 68

TRC Options Page (RTICANMM MainBlock)

Access

Located in TRC Page (RTICANMM MainBlock) on page 259.

Purpose

To specify general settings for the TRC file.

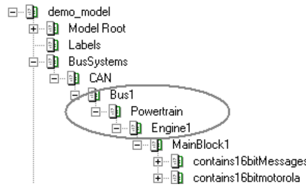
Dialog settings

Hierarchy of TRC tree to this block (use / to separate levels) Lets you specify the hierarchy of nodes in the TRC file. You can specify the hierarchy starting from the BusSystems/CAN node down to the level of the current RTICANMM MainBlock. Enter / to separate the hierarchy levels.

Example:

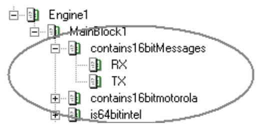
Suppose you specify the following hierarchy:

The corresponding hierarchy of nodes in the TRC file will be like this:

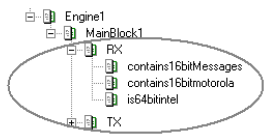


Macros Lets you open a list of the macros that you can use for the Hierarchy of TRC tree to this block (use / to separate levels).

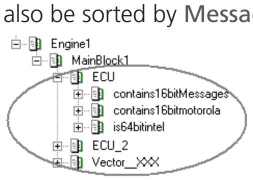
Message name as top level Lets you use message names as the topmost TRC file nodes for the current RTICANMM MainBlock.



TX/RX as top level Lets you use "TX" and "RX" as the topmost TRC file nodes for the current RTICANMM MainBlock. This allows you to separate TX and RX messages in the generated TRC file.



Use Network Node hierarchies Lets you use network node names as the topmost TRC file nodes for the current RTICANMM MainBlock. This allows you to identify all the messages of a network node in the generated TRC file. If no transmitting or receiving network node is specified, messages are placed under the *Vector__XXX* node. Message nodes that are under network node nodes can also be sorted by Message name as top level and TX/RX as top level.



Related topics

References

TRC Extras Page (RTICANMM MainBlock)

Access

Located in TRC Page (RTICANMM MainBlock) on page 259.

Purpose

To add additional information to the TRC file.

Description

RTI CAN MultiMessage Blockset lets you include database file attributes and user-defined variables to the TRC file and monitor them in ControlDesk. This allows you to display, for example, database file version information in ControlDesk. If you do not use a database, you can add only user-defined variables (see General Settings Page (RTICANMM MainBlock) on page 100).

In ControlDesk the variables and their values are read-only.

Dialog settings

Extra TRC variables Lists the additional variables and their values. You can select the variables to edit their names and values or delete them.

Name Lets you enter any character to define the name of a variable.

> **Note**
>
> You cannot add variables with identical names to the variable list. If you add a variable whose name already exists, the value of the existing variable is overwritten by the new value. The old value cannot be restored.

Value Lets you select an attribute of the database file or specify a user-defined value. If you select **userdefined=>**, the edit field is enabled and you can enter any numerical value. If you do not use a database, you can only add a user-defined value (see General Settings Page (RTICANMM MainBlock) on page 100).

Delete Lets you delete the selected variable from the variable list.

Add Lets you add a variable to the variable list.

Related topics

References

Peripheral Options Page (RTICANMM MainBlock)

Access

Located in General Options Page (RTICANMM MainBlock) on page 255.

Purpose

To specify general settings for the RTICANMM MainBlock.

Dialog settings

Create enable output port Lets you add an Enable outport to the current RTICANMM MainBlock. For details on this outport, refer to I/O characteristics on page 70.

Create traffic output port Lets you create a Traffic outport. If the RTICANMM MainBlock receives a message from another MainBlock or via the bus, the Traffic outport value is `1`. If the RTICANMM MainBlock receives no messages, the Traffic outport value is `0`, i.e., there is no traffic on the bus configured by this RTICANMM MainBlock. This can be useful for network management purposes.

Create triggering block Lets you add a Triggering to RTICANMM Block to the model. The block is connected to the RTICANMM MainBlock's Triggering inport. The Triggering to RTICANMM Block on page 293 is automatically added the next time you create an S-function for the RTICANMM MainBlock. For details, refer to Triggering to RTICANMM Block on page 293.

Create option for triggering block Lets you specify how to proceed with an existing Triggering to RTICANMM Block. The following options are available:

Option	Description
Ask if exists	Lets you decide how to proceed with an existing Triggering to RTICANMM Block each time the S-function for the RTICANMM MainBlock is created.
Overwrite if exists	Lets you overwrite the old Triggering to RTICANMM Block by a new one.
Add new block	Lets you add a new Triggering to RTICANMM Block without overwriting the old one.
Do nothing if exists	Lets you leave the existing Triggering to RTICANMM Block unchanged.

Name of triggering block (Available only if you select Create triggering block) Lets you enter a name for the triggering block to be created. If you leave this field empty, a default name will be used.

Create mapping block to TX Lets you add a Mapping to RTICANMM Block on page 295 to the model. The block is connected to the RTICANMM MainBlock's TX Data inport. The Mapping to RTICANMM Block on page 295 is automatically added the next time you create an S-function for the RTICANMM MainBlock. For details, refer to Mapping to RTICANMM Block on page 295.

Create option for mapping block Lets you specify how to proceed with an existing Mapping to RTICANMM Block. The following options are available:

Option	Description
Ask if exists	Lets you decide how to proceed with an existing Mapping to RTICANMM Block each time the S-function for the RTICANMM MainBlock is created.
Overwrite if exists	Lets you overwrite the old Mapping to RTICANMM Block by a new one.
Add new block	Lets you add a new Mapping to RTICANMM Block without overwriting the old one.
Do nothing if exists	Lets you leave the existing Mapping to RTICANMM Block unchanged.

Signals source in mapping (Available only if you select Create mapping block to TX) Lets you create a Simulink Ground block or Constant block for each TX model signal.

Note

Simulink Ground blocks do not have a data type. If you replace them by model signals to be transmitted as TX model signals, you have to take the data type of the model signals into account.

Name of mapping block (Available only if you select Create mapping block to TX) Lets you enter a name for the mapping block to be created. If you leave this field empty, a default name will be used.

ECU hierarchy in RX output port Lets you create an output port bus structure with an ECU hierarchy. If this checkbox is cleared, the output port bus structure is message-oriented.

Related topics

References

RTICANMM MainBlock 68

Pages of RTICANMM MainBlock (Message View)

Access

You can activate the message view by clicking **Settings - Open Settings Message View** in the RTICANMM MainBlock dialog. You can return to the dialog tree by clicking **OK**.

Where to go from here

Information in this section

ECU Page (RTICANMM MainBlock Message View)

Purpose

Displays the messages sent by the selected ECU.

Description

The message view displays all ECUs that send messages. ECUs that only receive messages are not displayed. All the messages of the selected ECU are displayed on the ECU page and can be sorted via context menu.

Dialog settings

ECU: <ECU_name> Displays the messages sent by the ECU. You can sort the list via context menu.

You can sort the list by:

- Name
- ID
- ID (hex)
- Length
- Signals
- Comment
- Cycle time
- Delay time
- Base time

- Update time
- Send type
- Start value
- J1939 priority
- J1939 destination
- J1939 source
- J1939 mapping
- Sending ECU(s)
- Receiving ECU(s)

Related topics

References

Message View (RTICANMM MainBlock)... 79

Messages Container Page (RTICANMM MainBlock Message View)

Purpose

Displays the settings of all messages of the selected ECU.

Dialog settings

ECU: <ECU_name> Displays the settings of all messages of the selected ECU. You can modify the settings of the messages on the corresponding Message Page (RTICANMM MainBlock Message View) on page 266. You can sort the list via context menu.

You can sort the list by:

- Name
- ID
- ID (hex)
- Length
- Signals
- Comment
- Cycle time
- Delay time
- Base time
- Update time
- Send type
- Start value
- J1939 priority
- J1939 destination
- J1939 source
- J1939 mapping

- Sending ECU(s)
- Receiving ECU(s)

Related topics

References

Message Page (RTICANMM MainBlock Message View)

Purpose

Lets you specify all the settings belonging to the specific message.

Description

This page lets you specify all message-specific settings located on several pages in the dialog tree.

The message view and the dialog tree depend on each other. As a consequence, all the pages of the dialog tree must be displayed if you want to specify all the settings of a message (refer to Tree Views Page (RTICANMM MainBlock) on page 98).

Note

If you specify settings in the message view while the appropriate pages in the dialog tree are hidden, all the settings that relate to the hidden pages are reset to their defaults.
If you want to reduce the dialog tree after you specified settings in the message view, select **User Specific Configuration Tree Settings** on the Tree Views Page (RTICANMM MainBlock) on page 98 and choose **Delete all default pages** from the context menu of the dialog tree (refer to Commands of the Dialog Tree (RTICANMM MainBlock) on page 88). If you reduce the dialog tree by its predefined selection of dialog pages (**Filter Configuration View by Feature** on the Tree Views Page (RTICANMM MainBlock) on page 98), you are warned if one or more pages have user-defined settings.

Dialog settings

TX Lets you specify the message as a TX message.

RX Lets you specify the message as a RX message.

Enable (Enabled only if the checkbox **Use these options for all** on the Message Enable Page (RTICANMM MainBlock) on page 140 in the dialog tree is not selected). Lets you specify message-specific transmission conditions. You can specify **Source**, **Logic**, **Def. Source**, and **Default** (see below).

Cyclic (Enabled only if the checkbox **Use these options for all** on the Message Cyclic Page (RTICANMM MainBlock) on page 144 in the dialog tree is not selected). Lets you specify cyclic message transmission. You can specify **Source**, **Logic**, **Def.Source**, and **Default** (see below).

Kickout (Enabled only if the checkbox **Use these options for all** on the Message Kickout Page (RTICANMM MainBlock) on page 148 in the dialog tree is not selected). Lets you transmit the message via kickout. You can specify the **Source** (see below).

Source Lets you specify the condition source. You can let an entry be included in the TRC file and/or generate a **Triggering** inport for the block as the condition source.

Setting	Description
None	Neither a TRC file entry nor a **Triggering** inport is generated for the selected message.
TRC	One entry is included in the TRC file for the selected TX message. The entry lets you enable or disable message transmission. No **Triggering** inport is generated. If an inport already exists, its structure remains unchanged.
Inport	A **Triggering** inport is generated if the inport does not yet exist. The entry lets you enable or disable message transmission. If you select cyclic triggering, this is added to the inport structure. See I/O characteristics on page 70. No TRC file entry is generated.
Both	For the selected TX message, one entry is included in the TRC file and/or a **Triggering** inport is generated if the inport does not yet exist. If you select cyclic triggering, this is added to the inport structure. See I/O characteristics on page 70. You can evaluate this option by a logical **AND** or **OR**.

Logic Lets you select how the source values are combined logically if **Both** is selected.

Settings	Description
AND	The source is evaluated by an **AND** operation.
OR	The source is evaluated by an **OR** operation.

Def. Source Lets you select the condition source to specify a **Default** triggering condition for. The available options are determined by the **Source** settings.

Setting	Description
TRC	The **Default** value is specified only for TRC, the **Triggering** inport remains unchanged.
Inport	The **Default** value is specified only for the **Triggering** inport, TRC remains unchanged.

Setting	Description
Both	The Default value is specified for both TRC and the Triggering inport.
*	The default source value of the selected message remains unchanged. This allows you to change other options without changing this one.

Default Lets you specify the default value for the triggering condition for the selected Default source.

Settings	Description
Enable	(Available if Enable is selected) Message transmission for the specified Default source is enabled by default.
Disable	(Available if Enable is selected) Message transmission for the specified Default source is disabled by default.
Cyclic	(Available if Cyclic is selected) Message transmission for the specified Default source is cyclic by default.
Triggered	(Available if Cyclic is selected) Message transmission for the specified Default source is triggered by default.

CycleTime Lets you specify the cycle time (in milliseconds) for the selected message.

DelayTime Lets you specify the delay time (in milliseconds) for the selected message.

Base/Update Time Lets you enable the base time, the update time, and the update number. Select the checkbox, then click Apply.

Base Time (Enabled if Base/Update Time is selected and Apply was clicked) Lets you specify the base time (in milliseconds) for the selected message.

Update Time (Enabled if Base/Update Time is selected and Apply was clicked) Lets you specify the update time (in milliseconds) for the selected message.

Update Number (Enabled if Base/Update Time is selected and Apply was clicked) Lets you specify the number of transmissions.

Timeout - Countdown (Enabled if Enable Timeout on the Triggering Options Page (RTICANMM MainBlock) on page 136 in the dialog tree and TX are selected) Lets you enter the default value for the number of possible transmissions of the selected message. The valid values are in the range 0 … 100.

Timeout - Counter (Enabled if Enable Timeout on the Triggering Options Page (RTICANMM MainBlock) on page 136 in the dialog tree and TX are selected) Lets you choose the behavior of the counter during timeout. You can specify whether the transmissions of a TX message are counted during timeout.

TX Status Port (Enabled if TX is selected) Lets you enable status ports for a TX message.

RX Status Port (Enabled if RX is selected) Lets you enable status ports for a RX message.

RX ID Port (Enabled if RX is selected) Lets you create the RX Data outport as a bus for the RTICANMM MainBlock (if it does not already exist). The message ID is added to the outport.

RX Message Length Port Lets you create the RX Data outport as a bus for the RTICANMM MainBlock (if it does not already exist). The message length is added to the outport. This is especially important when free raw messages are processed.

RX Message Counter Lets you enable the RX message counter.

TX Raw Data (Enabled if TX is selected) Lets you manipulate TX messages with raw data.

Settings	Description
Activate	Lets you activate the raw data option for the selected message.
Deactivate	Lets you deactivate the raw data option for the selected message.
TRC	Lets you select a TRC file as the source of the raw data.
Inport	Lets you select an inport as the source of the raw data.
Raw Data	Lets you select raw data as the default of the selected message.
Signals	Lets you select signals as the default of the selected message.

TX Raw Data Display (Enabled if TX is selected) Lets you include entries for TX raw data in the generated TRC file. You can then display TX raw data in ControlDesk.

RX Raw Data Display (Enabled if RX is selected) Lets you include entries for RX raw data in the generated TRC file. You can then display RX raw data in ControlDesk.

RX Raw Data Port (Enabled if RX is selected) Lets you enable raw data ports for the selected message.

RX Error Port (Enabled if RX is selected) Lets you enable error ports for the selected message.

RX Error Display (Enabled if RX is selected) Lets you include entries for RX errors in the generated TRC file. You can then display RX errors in ControlDesk.

Cycle Time Error Lets you specify the ranges for calculating cycle time errors.

Three edit fields are available:

- The value of the first edit field is the range (in %) for the defined message cycle time. This edit field is view-only. You have to specify the range in the edit field Always calculate from timeout tolerance [%] on the Cycle Time Error Page (RTICANMM MainBlock) on page 187 in the dialog tree.
- The second edit field (enabled if User-defined ranges on the Cycle Time Error Page (RTICANMM MainBlock) on page 187 in the dialog tree and RX are selected) lets you specify the minimum time limit (in milliseconds).
- The third edit field (enabled if User-defined ranges on the Cycle Time Error Page (RTICANMM MainBlock) on page 187 in the dialog tree and RX are selected) lets you specify the maximum time limit (in milliseconds).

TX ID Lets you modify the IDs of a TX message during run time. Modifying the IDs during run time is especially important when you work with free raw messages.

Settings	Description
NONE	The ID cannot be modified during run time.
TRC	The ID can be modified via TRC during run time.
INPORT	The ID can be modified via an inport during run time.
BOTH (DEFAULT TRC)	By default, the ID can be modified via TRC. You can change this default setting during run time.
BOTH (DEFAULT INPORT)	By default, the ID can be modified via an inport. You can change this default setting during run time.
*	The manipulation option of the selected message(s) remains unchanged. This allows you to change other options without changing this one.

TX Length Lets you modify the length of the TX message during run time. For non-J1939 messages, you can specify a message length in the range 1 ... 8. For J1939 messages, the maximum message length is 1785 bytes.

Settings	Description
None	The length cannot be modified during run time.
TRC	The length can be modified via TRC during run time.
Inport	The length can be modified via an inport during run time.
Default TRC	By default, the length can be modified via TRC. You can change this default setting during run time.
Default Inport	By default, the length can be modified via an inport. You can change this default setting during run time.

Message Variation (Enabled if **Change overlapping signals to message variations** on the General Settings Page (RTICANMM MainBlock) on page 100 in the dialog tree is selected and one or more messages with overlapping signals exist) Lets you switch between several variants of a message during run time. You can specify the default message variation and select the source by which you can switch between the message variations during run time.

Settings	Description
TRC	The message variation can be changed via TRC during run time.
Inport	The message variation can be changed via an Inport during run time.
Only Default	The message variation cannot be changed during run time, the default message variation is taken.

Checksum (Enabled if a header file is selected and one or more identifiers for checksum algorithms are specified on the Checksum Definition Page (RTICANMM MainBlock) on page 200 in the dialog tree) Lets you assign checksum algorithms to the message. You can specify one signal of the message as the checksum signal and select the checksum algorithm to be applied. You can enable or disable checksum calculation.

CustomCode (View-only) Displays the settings specified for Function name and Default on the Custom Code Page (RTICANMM MainBlock) on page 211 in the dialog tree. The first vertical drop-down list row displays the settings for preCRC, the second vertical drop-down list row displays the settings for postCRC.

Related topics

References

Message Info Page (RTICANMM MainBlock Message View)

Purpose

Displays all the database information of the selected message.

Dialog settings

Message: <message_name> Displays all the database information of the selected message in a list. This list is view-only.

Related topics

References

Signals Container Page (RTICANMM MainBlock Message View)

Purpose

Displays the settings of all the signals of the chosen message.

Dialog settings

Message: <message_name> Displays the settings of all the signals of the chosen message. You can modify the settings of the signals on the corresponding Signal Page (RTICANMM MainBlock Message View) on page 272. You can sort the list via context menu.

You can sort the list by:

- Name
- Startbit
- Length
- Byte layout
- Factor
- Offset
- Unit
- InitialValue
- Error value
- Min
- Max
- ReceiveECUs
- ModeValue
- AbsMin
- AbsMax
- Comment

Related topics

References

Signal Page (RTICANMM MainBlock Message View)

Purpose

Lets you specify all the settings belonging to one signal.

Description

This page lets you specify all signal-specific settings located on several pages in the dialog tree.

The message view and the dialog tree depend on each other. As a consequence, all the pages of the dialog tree must be displayed if you want to specify all the settings of a signal (refer to Tree Views Page (RTICANMM MainBlock) on page 98).

Note

If you specify settings in the message view while the appropriate pages in the dialog tree are hidden, all the settings that relate to the hidden pages are reset to their defaults.

If you want to reduce the dialog tree after you specified settings in the message view, select User Specific Configuration Tree Settings on the Tree Views Page (RTICANMM MainBlock) on page 98 and choose Delete all default pages from the context menu of the dialog tree (refer to Commands of the Dialog Tree (RTICANMM MainBlock) on page 88). If you reduce the dialog tree by its predefined selection of dialog pages (Filter Configuration View by Feature on the Tree Views Page (RTICANMM MainBlock) on page 98), you are warned if one or more pages have user-defined settings.

Dialog settings

RX Output (Enabled if RX on the Message Page (RTICANMM MainBlock Message View) on page 266 in the dialog tree is selected) Lets you specify the selected signal as an RX model signal.

TX Input (Enabled if TX on the Message Page (RTICANMM MainBlock Message View) on page 266 in the dialog tree is selected) Lets you specify the selected signal as a TX model signal.

Constant (Enabled if TX on the Message Page (RTICANMM MainBlock Message View) on page 266 in the dialog tree is selected) Lets you enable input manipulation for TX model signals. If input manipulation is enabled, the signal can be manipulated either by the respective model input via the TX Data inport (see I/O characteristics on page 70), or via a TRC file.

Saturation (Enabled if User-defined saturation on the Saturation Page (RTICANMM MainBlock) on page 218 in the dialog tree and TX are selected) Lets you specify the direction to which the saturation will be applied.

Settings	Description
None	The saturation is disabled.
Both	The saturation applies to both the input and the output signal.
Input	The saturation applies to the input signal (input saturation).
Output	The saturation applies to the output signal (output saturation).

Ranges - Min (Enabled if User-defined ranges on the Signal Ranges Page (RTICANMM MainBlock) on page 223 in the dialog tree is selected) Lets you enter the lower saturation limit for the selected signal. You can enter

- A numerical value
- `fromDBC`: The lower limit is always adapted to the value specified in the current database.
- *: The value remains unchanged. This can be useful if several signals are selected to change their upper saturation limits.

Ranges - Max (Enabled if **User-defined ranges** on the Signal Ranges Page (RTICANMM MainBlock) on page 223 in the dialog tree is selected) Lets you enter the upper saturation limit for the selected signal(s). You can enter

- A numerical value
- `fromDBC`: The upper limit is always adapted to the value specified in the current database.
- `*`: The value remains unchanged. This can be useful if several signals are selected to change their lower saturation limits.

Default (Enabled if **User-defined defaults** on the Signal Defaults Page (RTICANMM MainBlock) on page 221 in the dialog tree is selected) Lets you enter the default value for the transmission of the selected signal. You can enter a numerical value in decimal format, or type `fromDBC` to keep the default value specified in the database file.

SignalError (Enabled if **User-defined errors** on the Signal Errors Page (RTICANMM MainBlock) on page 225 in the dialog tree is selected) Lets you enter an error value. You can enter

- A numerical value.
- `X` if you do not want to specify an error value for the selected signal. In that case, you cannot switch to transmitting an error value for the selected signal.
- `fromDBC` if the error value specified in the current database is to be used.

Dynamic Signal Lets you activate dynamic values for the transmission of the selected signal. Select the checkbox and click **Apply** to enable **Value** and **Countdown**.

Value (Enabled if **Dynamic Signal** is selected) Lets you specify the default dynamic value.

Countdown (Enabled if **Dynamic Signal** is selected) Lets you specify the countdown value.

Toggle Lets you specify the toggle period (in seconds) for the selected signal.

Parity Lets you specify the selected signal as a parity signal. You can select another signal for which you want to calculate the parity.

Odd Lets you specify odd parity. The parity bit is set so that there is an odd number of `"1"` bits for the selected signal, including the parity bit.

Even Lets you specify even parity. The parity bit is set so that there is an even number of `"1"` bits for the selected signal, including the parity bit.

Counter Lets you specify the selected signal as a counter signal.

Start Lets you specify the counter start value.

Step Lets you specify the increment of the counter.

Divisor Lets you specify the step length of the counter. If you specify a divisor value of 3, for example, the counter value remains constant for 3 message transmissions. After the next message transmission, the counter value is incremented by the **Step** value. Additionally, the **Start**, **Step**, and **Step length** parameters are divided by the divisor's value.

Note

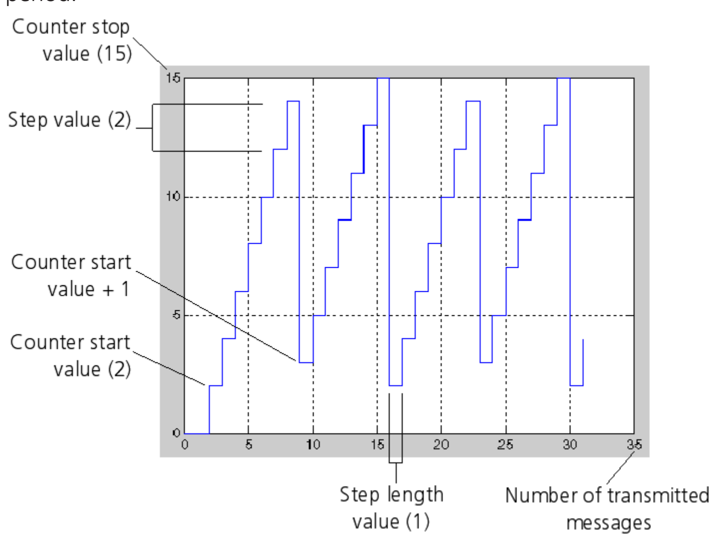
The Divisor parameter is provided only for compatibility reasons. Specify a divisor value of 1 and work with the Step length instead.

Step length Lets you specify the step length of the counter. If you specify a step length of 3, for example, the counter value remains constant for 3 message transmissions. After the next message transmission, the counter value is incremented by the Step value.

Note

If you work with the Step length, specify a Divisor value of 1.

Stop Lets you specify the counter stop value. The counter value exceeds the stop value periodically: Suppose you specify Stop = 15 and Step = 2. The counter turns around to (counter start value + 1) when it reaches 14 (first period). For the second period, the counter reaches the counter stop value and turns around to the counter start value. The cycle starts again with the first period.



Counter behavior Lets you specify the counter behavior if the signal is not transmitted with the counter but with another signal type such as a constant or toggle value.

Setting	Description
Continue Counting	The counter continues counting whether or not it is transmitted/received. When a different signal type is transmitted/received, the counter is incremented internally each time the other signal type is transmitted/received. The next time the counter is transmitted/received, it starts counting from the internally incremented counter value.

Setting	Description
Stop Counting	When a signal type other than the counter is transmitted/received, the counter stops counting. The next time the counter is transmitted/received, it starts counting from the previous counter value.
Set to Constant	When a signal type other than the counter is transmitted/received, the counter stops counting. The next time the counter is transmitted/received, it starts counting from the previously received signal value, regardless of the signal type.

Show Counter Lets you plot the specified counter signal graphically.

Default Manipulation (Enabled if **Enable Manipulation** and **User-defined defaults** on the Signal Default Manipulation Page (RTICANMM MainBlock) on page 244 in the dialog tree are selected) Lets you set the default signal manipulation option individually for each signal to be transmitted. The following signal manipulation options are available depending on the signal manipulation option you enabled:

Signal Manipulation Option	Option is Available if ...
Constant	Always available except for the following case: The signal is specified as a TX model signal, and input manipulation is disabled for the signal.
Constant/Gateway	The signal is not specified as a TX model signal, and a gateway signal is specified for the signal.
Input	The signal is specified as a TX model signal.
Input/Gateway	The signal is specified as a TX model signal, and a gateway signal is specified for the signal.
Error	An error value is specified for the signal.
Toggle	The signal is specified as a toggle signal.
Parity	The signal is specified as a parity signal.
Counter / IncrementCounter	The signal is specified as a counter signal.
DynValue	The use of dynamic values is specified for the signal.

Gateway Signal (Enabled if gateway signals on the Gateway Signals Page (RTICANMM MainBlock) on page 248 are defined, and **Enable Manipulation** and **User-defined defaults** on the Signal Default Manipulation Page (Gateway) (RTICANMM MainBlock) on page 251 in the dialog tree are selected) Lets you enable signal default manipulation for gateway signals to be transmitted. The following signal manipulation options are available depending on the signal manipulation option you enabled:

Signal Manipulation Option	Option is Available if ...
Constant	Always available except for the following case: The signal is specified as a TX model signal, and input manipulation is disabled for the signal.
Constant/Gateway	The signal is not specified as a TX model signal, and a gateway signal is specified for the signal.
Input	The signal is specified as a TX model signal.

Signal Manipulation Option	Option is Available if ...
Input/Gateway	The signal is specified as a TX model signal, and a gateway signal is specified for the signal.
Error	An error value is specified for the signal.
Toggle	The signal is specified as a toggle signal.
Parity	The signal is specified as a parity signal.
Counter / IncrementCounter	The signal is specified as a counter signal.
DynValue	The use of dynamic values is specified for the signal.

Related topics

References

Message View (RTICANMM MainBlock).......79
Signal Info Page (RTICANMM MainBlock Message View).......277
Signals Page (RTICANMM MainBlock).......213

Signal Info Page (RTICANMM MainBlock Message View)

Purpose

Displays all the database information of the selected signal.

Dialog settings

Message: <message_name> - Signal: <signal_name> Displays all the database information of the selected signal in a list. This list is view-only.

Related topics

References

Message View (RTICANMM MainBlock).......79
Signal Page (RTICANMM MainBlock Message View).......272

CAN Bus Gateway Block

Introduction

The dSPACE_Blocksets/rticanmmlib library provides a CAN bus gateway block that lets you exchange CAN messages between two CAN buses.

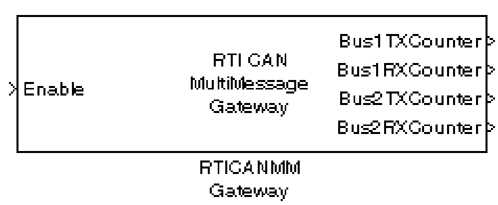
RTICANMM Gateway

Where to go from here

Information in this section

Block Description (RTICANMM Gateway)

Block



Purpose

To exchange messages between two CAN buses.

Description

Gatewaying means exchanging CAN messages between two CAN buses. Gatewaying also applies to messages that are not specified in the database file.

Messages specified in the database file can be excluded from being exchanged.

Note

- You cannot gateway between two RTICANMM MainBlocks if the corresponding CAN controllers belong to different processors of a multiprocessor system.
- You cannot gateway between a CAN controller for which you use the RTI CAN MultiMessage Blockset and a CAN controller for which you use the RTI CAN Blockset.
- The CAN bus you want to use, and general settings for using the RTI CAN MultiMessage Blockset, must already have been specified using RTICANMM ControllerSetup and RTICANMM GeneralSetup blocks.
- The loopback mechanism is disabled for CAN messages that are gatewayed via the RTICANMM Gateway block, i.e., messages that are transmitted by a CAN controller via the Gateway block are not received as loopback messages by the same CAN controller. This prevents endless transfer of the same messages between the buses.

I/O characteristics

The table below describes the available block inport:

Simulink Inport	Range	Simulink Data Type	Description
Enable	0 … 1	Boolean	Enables or disables the exchange of messages between two CAN buses. If CAN Bus 1 <=> CAN Bus 2 is selected as the gateway direction, the Enable inport is used for globally enabling gatewaying in both directions. In addition, the Enable Dir x inports are available for enabling/disabling the individual gateway directions.
Enable Dir 1	0 … 1	Boolean	(Available only if CAN Bus 1 <=> CAN Bus 2 is selected as the gateway direction) Enables or disables the exchange of messages from CAN Bus 1 to CAN Bus 2.
Enable Dir 2	0 … 1	Boolean	Available only if CAN Bus 1 <=> CAN Bus 2 is selected as the gateway direction) Enables or disables the exchange of messages from CAN Bus 2 to CAN Bus 1.

The table below describes the available block outports:

Simulink Outport	Range	Simulink Data Type	Description
Bus1TXCounter	0 … $(2^{32}-1)$	Uint32	Outputs the number of messages transmitted by this block to CAN bus 1
Bus1RXCounter	0 … $(2^{32}-1)$	Uint32	Outputs the number of messages received by this block and sent by CAN bus 1
Bus2TXCounter	0 … $(2^{32}-1)$	Uint32	Outputs the number of messages transmitted by this block to CAN bus 2
Bus2RXCounter	0 … $(2^{32}-1)$	Uint32	Outputs the number of messages received by this block and sent by CAN bus 2

Dialog pages

The dialog settings can be specified on the following pages:

- Setup Page (RTICANMM Gateway) on page 283
- Databases Page (RTICANMM Gateway) on page 284
- Exclude Message Lists Page (RTICANMM Gateway) on page 285
- General Options Page (RTICANMM Gateway) on page 290

Related topics

Basics

Lesson 12 (Advanced): Gatewaying Messages (RTI CAN MultiMessage Blockset Tutorial 📖)
Lesson 13 (Advanced): Working with a J1939-Compliant DBC File (RTI CAN MultiMessage Blockset Tutorial 📖)

References

Settings Menu (RTICANMM Gateway)

Purpose

To save or load the settings of the RTICANMM Gateway block.

Description

For easy reuse you can save the settings of the current RTICANMM Gateway block in a MAT file. To reuse the saved settings, you can load the MAT file to the current or any other RTICANMM Gateway block.

Menu commands

Load Settings Lets you load the settings for the current RTICANMM Gateway block. Specify the MAT file containing the desired settings.

Save Settings Lets you save the settings of the RTICANMM Gateway block to a MAT file.

Related topics

Basics

Setup Page (RTICANMM Gateway)

Purpose

To specify the gateway of CAN messages.

Description

You can gateway CAN messages in two ways:

- *Controller gateway*: This is a gateway between two CAN controllers. You gateway between two RTICANMM ControllerSetup blocks. It is independent of the active CAN controller variation.
- *MainBlocks gateway*: This is a gateway between different variations of two CAN controllers. You gateway between two RTICANMM MainBlocks. The MainBlocks gateway is active only if the variations of both CAN controllers are active at the same time.

You can also specify the gateway direction of messages. Messages can be exchanged between CAN bus 1 and CAN bus 2 and/or vice versa.

Dialog settings

MainBlocks gateway Lets you specify a gateway between different variations of two CAN controllers.

Controller gateway Lets you specify a gateway between two CAN controllers, regardless of the active CAN controller variation.

Select MainSetup Block (Available only if you select MainBlocks gateway) Lets you select the RTICANMM MainBlock for gatewaying.

Select ControllerSetup Block (Available only if you select Controller gateway) Lets you select the RTICANMM ControllerSetup blocks for gatewaying.

Gateway direction Lets you specify the gateway direction of messages or deactivate gatewaying from CAN bus 1 to CAN bus 2 and vice versa. However, messages specified on the Exclude Message Lists Page (RTICANMM Gateway) on page 285 are never exchanged. You can choose:

- CAN Bus 1 <=> CAN Bus 2 Lets you exchange messages from CAN bus 1 to CAN bus 2 and vice versa.
- CAN Bus 1 => CAN Bus 2 Lets you exchange messages from CAN bus 1 to CAN bus 2.
- CAN Bus 1 <= CAN Bus 2 Lets you exchange messages from CAN bus 2 to CAN bus 1.
- Gateway Disabled Lets you deactivate gatewaying. No messages will be exchanged.

The selected gateway direction(s) are displayed in the list. You can specify manipulation conditions for their enable behavior.

Source Lets you specify the source by which you can manipulate the gateway directions during run time. You can let an entry be included in the TRC file and/or generate an Enable inport for the block as the condition source.

Setting	Description
NONE	The gateway direction(s) cannot be manipulated during run time.
TRC	The selected gateway direction(s) can be enabled/disabled via TRC file.
INPORT	The selected gateway direction(s) can be enabled/disabled via an inport. One **Enable** inport is generated for each gateway direction.
BOTH	One entry is included in the TRC file and one **Enable** inport is generated for each selected gateway direction. You can evaluate this option by a logical **AND** or **OR**.

Logic Lets you select how to combine the source values logically if **BOTH** is selected.

Settings	Description
AND	The source is evaluated by an **AND** operation.
OR	The source is evaluated by an **OR** operation. This is the default value.

Default TRC (Available only if **TRC** or **BOTH** is selected as the source) Lets you specify the default value for the TRC file that is used at model start.

Set Lets you assign the specified conditions to the selected gateway direction(s).

Related topics

References

Databases Page (RTICANMM Gateway)

Purpose

To specify the databases for the RTICANMM Gateway.

Description

You can specify a separate database file for each direction specifying the exclude lists for messages of this direction.

Dialog settings

Database for direction CAN Bus 1 to CAN Bus 2 Lets you specify a database file to exclude messages from being gatewayed from CAN Bus 1 to CAN Bus 2. By default, the checkbox is cleared and the database file of the MainBlock is specified if **MainBlocks gateway** on the Setup Page (RTICANMM

Gateway) on page 283 is selected. Selecting the checkbox lets you enable the edit field and specify another database file.

Use relative path (Available only if you select Database for direction CAN Bus 1 to CAN Bus 2) Lets you set the folder relative to the model root folder or as an absolute path.

> **Tip**
>
> Using a relative path allows you to easily move the model and all the files belonging to it to another location on your file system without having to change any path settings.

Cluster (Enabled if you specify a database file other than a MAT file) Lets you select the bus whose messages you want to exclude from gatewaying.

Database for direction CAN Bus 2 to CAN Bus 1 Lets you specify a database file to exclude messages from being gatewayed from CAN Bus 2 to CAN Bus 1. By default, the checkbox is cleared and the database file of the MainBlock is specified if MainBlocks gateway on the Setup Page (RTICANMM Gateway) on page 283 is selected. Selecting the checkbox lets you enable the edit field and specify another database file.

Use relative path (Available only if you select Database for direction CAN Bus 2 to CAN Bus 1) Lets you set the folder relative to the model root folder or as an absolute path.

> **Tip**
>
> Using a relative path allows you to easily move the model and all the files belonging to it to another location on your file system without having to change any path settings.

Cluster (Enabled if you specify a database file other than a MAT file) Lets you select the bus whose messages you want to exclude from gatewaying.

Related topics

References

Exclude Message Lists Page (RTICANMM Gateway)

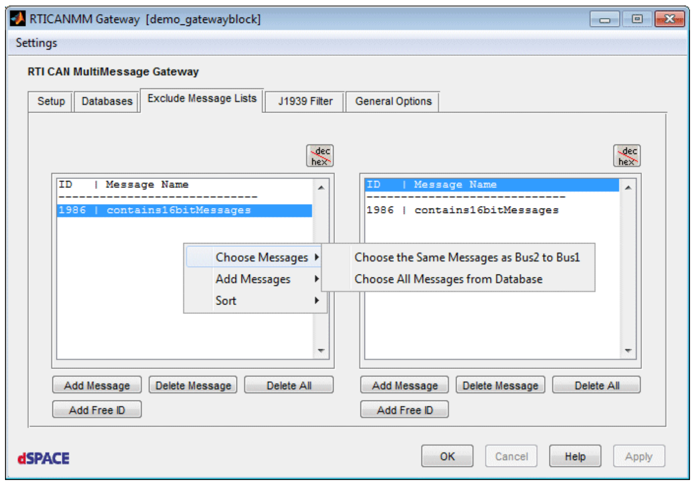
Purpose

To exclude messages from being exchanged between two CAN buses.

Description

The RTICANMM Gateway block works in both directions: from CAN bus 1 to CAN bus 2 and vice versa.

- *Excluding messages from being transferred*: You can exclude single messages from being transferred. You can exclude messages individually for each direction (CAN bus 1 to CAN bus 2, and CAN bus 2 to CAN bus 1).
- *Context menu*: The lists of excluded messages provide a context menu that allows you to select specific messages. The menu commands depend on whether you specify a Controller gateway or a MainBlocks gateway.



Dialog settings

dec/hex Lets you display the message identifiers in decimal or hexadecimal numeric format.

Messages list Lists the IDs of messages to be excluded.

Add Message (Available only if a DBC file, MAT file or AUTOSAR system description file is selected as the database on the Databases Page (RTICANMM Gateway) on page 284) Opens the **RTICANMM Gateway (Select Messages)** dialog, where you can select messages you want to add to the Exclude message list.

Delete Message Lets you remove the selected message from the Exclude message list. The message is then transferred if the RTICANMM Gateway block receives it.

Delete All Removes all messages from the Exclude message list. All configured messages are then transferred.

Add Free ID Lets you add IDs to the exclude list that are not contained in the database file.

Related topics

References

J1939 Filter Page (RTICANMM Gateway)

Purpose

Excludes messages based on a J1939 database from being exchanged between two CAN buses.

Description

Extended identifier format for J1939 To work with messages of a J1939-compliant DBC file, you have to specify an extended identifier format on the Setup Page (RTICANMM ControllerSetup) of the ControllerSetup block.

Parameter group number (PGN) A 16-bit number in the 29-bit message identifier of a CAN message defined in a J1939-compliant DBC file. Each PGN references a *parameter group* that groups parameters and assigns them to the 8-byte data field of the message. A parameter group can be the engine temperature including the engine coolant temperature, the fuel temperature, etc. PGNs and parameter groups are defined by the SAE (see SAE J1939/71 Vehicle Application Layer).

The first 8 bits of the PGN represent the *PDU_F* (Protocol Data Unit format). The PDU_F value specifies the communication mode of the message (peer-to-peer or broadcast). The interpretation of the *PDU_S* value (PDU-specific) depends on the PDU_F value. For messages with a PDU_F < 240 (peer-to-peer communication, also called PDU1 messages), PDU_S is not relevant for the PGN, but contains the destination address of the network node that receives the message. For messages with a PDU_F ≥ 240 (broadcast messages, also called PDU2 messages), PDU_S specifies the second 8 bits of the PGN and represents the group extension. A group extension is used to increase the number of messages that can be broadcast in the network.

PDU_F (first 8 bits)	PDU_S (second 8 bits)	Communication Mode
< 240	Destination address	Peer-to-peer (message is transmitted to one destination network node)

PDU_F (first 8 bits)	PDU_S (second 8 bits)	Communication Mode
≥ 240	Group extension	Broadcast (message is transmitted to any network node connected to the network)

Broadcast messages J1939 messages with a PDU_F ≥ 240 are broadcast messages. In a CAN-J1939 network, a broadcast message is transmitted to any network node connected to the network.

Peer-to-peer messages J1939 messages with a PDU_F < 240 are peer-to-peer messages. In a CAN-J1939 network, a peer-to-peer message is transmitted to only one destination network node (receiving network node).

Excluding J1939 from gatewaying You can specify messages based on a J1939 database to be excluded from gatewaying. You can specify messages only if

- MainBlock gateway on the Setup Page (RTICANMM Gateway) is selected and the DBC file of the related MainBlocks supports the J1939 standard *or*
- DBC files that support the J1939 standard are specified on the Databases Page (RTICANMM Gateway).

Filter list The filter list displays the messages to be excluded from gatewaying. Empty cells are interpreted like wildcards: For example, if you do not specify the source for a message, the corresponding cell in the filter list is empty. As a consequence, all messages that match the other specified settings of this message are excluded from gatewaying, regardless of their specified source value or message name. As the message name is not evaluated, it is only an aid to identifying the message in the message list. Via context menu, you can:

- Choose messages Lets you choose messages via the Choose from MainBlock dialog. This dialog opens only if you specified an EXT message identifier format on the Setup Page (RTICANMM ControllerSetup) and both related MainBlocks were generated. You can select all messages of one MainBlock according to their format (TX or RX) and filter direction. The selected messages are added to the filter list, all previous specified messages in the list are deleted. Adding the messages to the filter list can take some seconds depending on the number of selected messages.
- Add messages Lets you add messages to the filter list via the Add from MainBlock dialog. This dialog opens only if you specified an EXT message identifier format on the Setup Page (RTICANMM ControllerSetup) and both related MainBlocks were generated. You can select all messages of one MainBlock depending on their format (TX or RX) and filter direction. The selected messages are added to the previously defined messages in the filter list. Adding the messages to the filter list can take some seconds depending on the number of selected messages.
- Sort Lets you sort the filter list by the filter direction, priority, PDU format, PDU specific, source or message name.

Dialog settings

dec/hex Lets you display the message identifiers in decimal or hexadecimal numeric format.

Add Opens the RTICANMM Gateway (Select J1939 Messages) dialog, where you can specify messages you want to add to the filter list. In the dialog you can specify the following settings:

- Filter direction - any Lets you specify a filter direction for both directions, from CAN bus 1 to CAN bus 2 and vice versa. If you select this setting, you have to specify a Select database.
- Filter direction - CAN bus 1 => CAN bus 2 Lets you specify a filter direction from CAN bus 1 to CAN bus 2.
- Filter direction - CAN bus 1 <= CAN bus 2 Lets you specify a filter direction from CAN bus 2 to CAN bus 1.
- Select database (enabled if any is selected as Filter direction) Lets you select a database from CAN bus 1 to CAN bus 2 or vice versa.
- Messages Lets you select a message specified in the selected DBC file.
- Src Lets you specify a source ECU (transmitting ECU) for the selected message.
- Dest (enabled only if you select a peer-to-peer message) Lets you specify a destination ECU (receiving ECU) for the selected message. The specified ECU applies to the `PDU_S` table row in the message list.
- Priority Lets you specify a priority for the selected message. You can select a priority from 1 (high) ... 7 (low).
- Group Ext. (dec) (enabled only if you select a broadcast message) Lets you change the group extension. The specified group extension is the PDU_S value for the broadcast message. You can change this value to specify a message you cannot specify in the DBC file or via the Bus Navigator of ControlDesk. As you change the PDU_S value of the PGN, you specify a new message.

Note

If you change the group extension, the message excluded from gatewaying differs from the message name specified in the filter list. As a consequence, you cannot identify the excluded message by its message name.

Click OK to add the specified message to the filter list. You can add several messages to the filter list without closing the dialog. The dialog remains open until you click Close.

Add J1939 container Opens the RTICANMM Gateway (Select J1939 Container) dialog, where you can select container messages you want to add to the filter list. In the dialog you can specify the following settings:

- Filter direction - any Lets you specify a filter direction for both directions, from CAN bus 1 to CAN bus 2 and vice versa. If you select this setting, you have to specify a Select database.
- Filter direction - CAN bus 1 => CAN bus 2 Lets you specify a filter direction from CAN bus 1 to CAN bus 2.
- Filter direction - CAN bus 1 <= CAN bus 2 Lets you specify a filter direction from CAN bus 2 to CAN bus 1.
- Select database (enabled if any is selected as Filter direction) Lets you select a database from CAN bus 1 to CAN bus 2 or vice versa.

- **Select J1939 container for exclusion** Lets you select the J1939 container messages to be excluded. Multiple selection is possible with the **`Shift`** or **`Ctrl`** key. Click **OK** to add the container messages to the filter list and close the dialog. For peer-to-peer container messages, only the PDU_F value is specified in the message list, for broadcast container messages, the PDU_F and PDU_S values are specified in the message list.

Delete Lets you delete the selected message from the filter list.

Delete All Lets you delete all messages from the filter list.

Related topics

Basics

Lesson 13 (Advanced): Working with a J1939-Compliant DBC File (RTI CAN MultiMessage Blockset Tutorial 📖)

General Options Page (RTICANMM Gateway)

Purpose

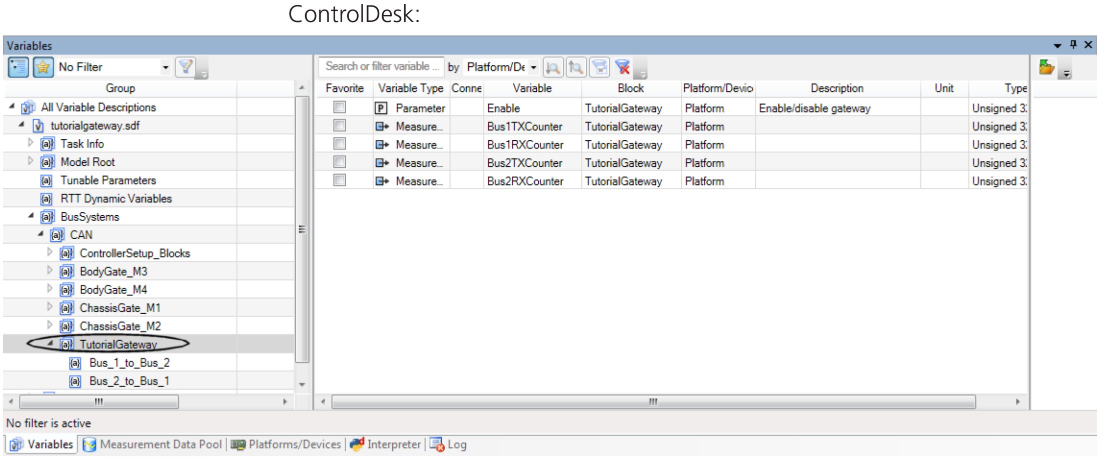
To make the RTICANMM Gateway block available in the TRC file.

Description

Whenever you create an S-function for an RTICANMM Gateway block, a TRC file is also automatically created. The TRC file contains entries for the gateway, for example, for all the CAN signals of the messages to be gatewayed. These entries let you enable or disable the gateway and modify the exclude message lists during run time.

- *TRC file and ControlDesk*: When you open the `<model>.sdf` file in ControlDesk, you will find the entries of the RTICANMM Gateway block in the **Variables** controlbar of ControlDesk below "BusSystems/CAN" by default.

  The illustration below shows an SDF file in the **Variables** controlbar of ControlDesk:



- *Bus Navigator*: The **Bus Navigator** of ControlDesk displays gateways as nodes on the same level as the CAN controllers in the Bus Navigator tree. You can use the gateway node to create a gateway layout. The gateway layout displays information on the gateway and lets you change the filter lists during run time.

  For more information on the **Bus Navigator** of ControlDesk, refer to ControlDesk Bus Navigator 📖.

You can configure the TRC-related settings of the block on this page.

Dialog settings

Gateway name Lets you specify the name for the gateway, which is used in the TRC file. This name must be unique within the model even if it is an RTI-MP model, in a similar way to the CAN name of the MainBlock. For SCALEXIO systems, the gateway name must be unique within a whole multicore application.

Hierarchy of TRC tree to this block (use / to separate levels) Lets you specify the hierarchy of nodes in the TRC file. You can specify the hierarchy starting from the **BusSystems/CAN** node down to the level of the current RTICANMM Gateway block. Enter **/** to separate the hierarchy levels.

Disable filter list - CAN bus 1 to CAN bus 2 Lets you disable the filter list for the direction from CAN bus 1 to CAN bus 2.

Disable filter list - CAN bus 2 to CAN bus 1 Lets you disable the filter list for the direction from CAN bus 2 to CAN bus 1.

Change online - CAN bus 1 to CAN bus 2 Lets you add messages to or remove them from the filter list for the direction from CAN bus 1 to CAN bus 2 during run time.

Change online - CAN bus 2 to CAN bus 1 Lets you add messages to or remove them from the filter list for the direction from CAN bus 2 to CAN bus 1 during run time.

Sample time Lets you enter the sample time (the intervals for reading data).

Sample time	Meaning
- 1	Lets you use the sample time inherited from the model.
> 0	Lets you specify the sample time in milliseconds. Use any multiple of the "Fixed step size" chosen for the model.

Related topics

References

RTICANMM Gateway..280

Peripheral Blocks

Introduction

The RTICANMM MainBlock on page 68 can automatically create peripheral blocks. You can use these blocks to provide the connected RTICANMM MainBlock on page 68 with message triggering options and signals from the model to be transmitted via the corresponding CAN controller.

Where to go from here

Information in this section

Triggering to RTICANMM Block

Block

Purpose

To specify triggering options for TX message transmission.

Description

This block is automatically created if the following conditions are met:

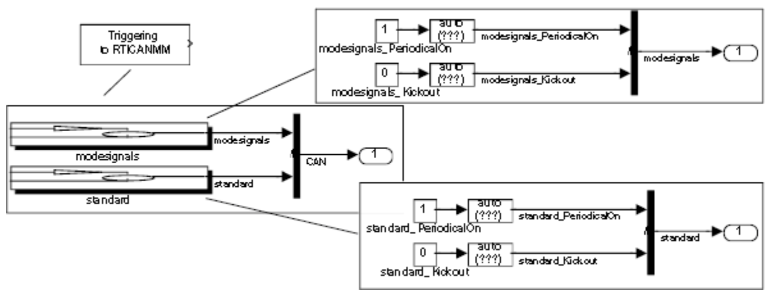
- You selected **Create triggering block** on the Peripheral Options Page (RTICANMM MainBlock) on page 262
- You selected **Enable Options**, **Cyclic Options**, and/or **Kickout Options** as a triggering option for TX message transmission (see Triggering Options Page (RTICANMM MainBlock) on page 136).

Or:

You specify the cycle time or delay time of at least one TX message from within the model: see TX Cycle Time Page (RTICANMM MainBlock) on page 154 and TX Delay Time Page (RTICANMM MainBlock) on page 156.

The corresponding RTICANMM MainBlock on page 68 gets the **Triggering** inport (see I/O characteristics on page 70). This inport is automatically connected to the Triggering to RTICANMM block.

- *Triggering options for TX messages*: The Triggering to RTICANMM block provides the connected RTICANMM MainBlock on page 68 with a struct containing the triggering options for each TX message.
- *Internal structure of the Triggering to RTICANMM block*: The illustration below shows an example Triggering to RTICANMM block and its internal structure. The **Cyclic Options** and **Kickout Options** triggering options are selected for the **modesignals** and **standard** TX messages. These options are represented as Simulink Constant blocks within the Triggering to RTICANMM block.



You can replace the Simulink Constant blocks, for example, by block inputs.

Note

Ensure that the labels will not get lost.

Related topics

Basics

Lesson 3: Triggering Message Transmission via the Model and ControlDesk (RTI CAN MultiMessage Blockset Tutorial 📖)
Lesson 5: Working with Model-Specific Bus Hierarchies (RTI CAN MultiMessage Blockset Tutorial 📖)

References

Mapping to RTICANMM Block

Block

Purpose

To specify the signals from the model to be transmitted.

Description

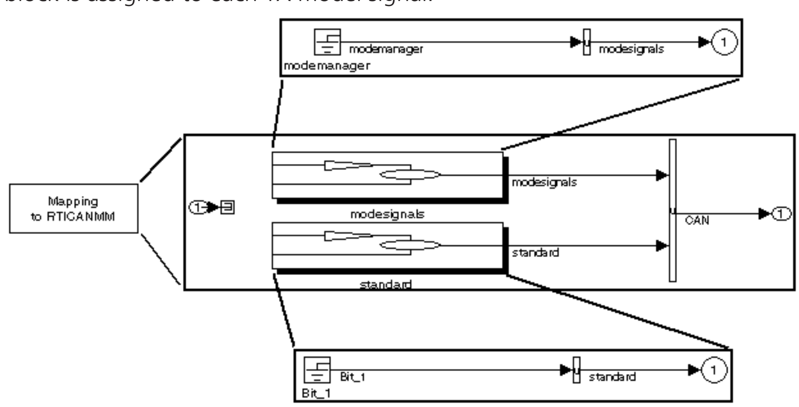
This block is automatically created if the following conditions are met:

- You selected **Create mapping block to TX** on the Peripheral Options Page (RTICANMM MainBlock) on page 262
- You specified one or more signals as TX model signals on the Model Signals (TX) Page (RTICANMM MainBlock) on page 214

The corresponding RTICANMM MainBlock on page 68 gets the **TX Data** inport (see I/O characteristics on page 70). This inport is automatically connected to the Mapping to RTICANMM block.

- *Signals from the model to be transmitted*: The Mapping to RTICANMM block provides the connected RTICANMM MainBlock on page 68 with a struct containing the signals from the model to be transmitted.
- *Internal structure of the Mapping to RTICANMM block*: The illustration below shows a Mapping to RTICANMM block and its internal structure. The signals **modemanager** and **Bit_1** are specified as TX model signals. They belong to the **modesignals** and **standard** TX messages, respectively.

  The Mapping to RTICANMM block contains subsystems for the **modesignals** and **standard** TX messages. Within the subsystems, one Simulink Ground block is assigned to each TX model signal.



- *Specifying TX model signals from within the model*: To specify a TX model signal from within the model, replace the Simulink Ground block that corresponds to the TX model signal by an empty Simulink subsystem. Name the subsystem according to the TX model signals. Connect the model signal to the subsystem. This ensures that the TX model signals label will not get lost when you specify the model signal.

Related topics

Basics

Lesson 4: Manipulating Signal Values via the Model and ControlDesk (RTI CAN MultiMessage Blockset Tutorial 📖)
Lesson 5: Working with Model-Specific Bus Hierarchies (RTI CAN MultiMessage Blockset Tutorial 📖)

References

Model Signals (TX) Page (RTICANMM MainBlock)........ 214
Peripheral Blocks........ 293
Peripheral Options Page (RTICANMM MainBlock)........ 262
RTICANMM MainBlock........ 68

Appendix

Where to go from here

Information in this section

Mapping Imported DBC Files and AUTOSAR System Description Files

Mapping Implicitly Generated Signals

Introduction

Unlike DBC files, AUTOSAR system description files contain signals that are not explicitly defined. Instead, they are implicitly generated from other AUTOSAR elements to make them available in RTICANMM.

Examples of such signals are listed below. The associated descriptions in the AUTOSAR system description file and in the DBC file are displayed for each example, and you can see how the database entries are mapped to the signals in RTICANMM.

MuxSwitch signals

MuxSwitch signals are generated for multiplexed signals. They are used to activate the corresponding mode-dependent signals.

Description in AUTOSAR system description file

```
<MULTIPLEXED-I-PDU>
   <SHORT-NAME>multiplexed_i_pdu_1</SHORT-NAME>
   ...
   <SELECTOR-FIELD-BYTE-ORDER>MOST-SIGNIFICANT-BYTE-FIRST
   </SELECTOR-FIELD-BYTE-ORDER>
   <SELECTOR-FIELD-LENGTH>3</SELECTOR-FIELD-LENGTH>
   <SELECTOR-FIELD-START-POSITION>15</SELECTOR-FIELD-START-POSITION>
</MULTIPLEXED-I-PDU>
```

Description in DBC file

```
SG_ <MESSAGE>_MODE M  : 15|3@1+ (1,0) [0|7] "" ECU1,ECU2
```

The following illustration shows what the signal description consists of:

```
SG_  <MESSAGE>_MODE M  : 15|3@1+ (1,0) [0|7] "" ECU1,ECU2
 |        |         |     | | ||  | |  | |  |  |____|
 |        |         |     | | ||  | |  | |  |     |________ Network node name list (receivers)
 |        |         |     | | ||  | |  | |  |______________ Unit (32 characters)
 |        |         |     | | ||  | |  | |_________________ Maximum
 |        |         |     | | ||  | |  |___________________ Minimum
 |        |         |     | | ||  | |______________________ Offset
 |        |         |     | | ||  |________________________ Factor
 |        |         |     | | ||___________________________ Sign: + = Unsigned, - = Signed
 |        |         |     | | |____________________________ Byte type: 0 = MOTOROLA, 1 = INTEL
 |        |         |     | |______________________________ Bit count
 |        |         |     |________________________________ Start bit
 |        |         |______________________________________ Signal type: empty = standard signal,
 |        |                                                              M = mode signal
 |        |________________________________________________ Signal name
 |_________________________________________________________ Signal keyword
```

Signal presentation in RTICANMM Based on the descriptions above, RTICANMM generates the following signal:

Signal Property	Value
Name	multiplexed_i_pdu_1_Mode
Length	3
MinValue	0
MaxValue	7
Offset	0
Scale	1
StartBit	15

Update bit for PDUs

The update bit of a PDU indicates whether data of the PDU has been changed.

Description in AUTOSAR system description file

```
<FRAME>
   <SHORT-NAME>frame_18</SHORT-NAME>
   <FRAME-LENGTH>8</FRAME-LENGTH>
   <PDU-TO-FRAME-MAPPINGS>
      <PDU-TO-FRAME-MAPPING>
         <SHORT-NAME>frame_18_pdu</SHORT-NAME>
         <PACKING-BYTE-ORDER>MOST-SIGNIFICANT-BYTE-LAST</PACKING-BYTE-ORDER>
         <PDU-REF DEST="SIGNAL-I-PDU">signal_i_pdu_10</PDU-REF>
         <START-POSITION>0</START-POSITION>
         <UPDATE-INDICATION-BIT-POSITION>63</UPDATE-INDICATION-BIT-POSITION>
      </PDU-TO-FRAME-MAPPING>
   </PDU-TO-FRAME-MAPPINGS>
</FRAME>
```

This AUTOSAR description results in an implicit signal with the name 'frame_18_pdu_UB'.

The general rule for determining the name of a PDU update bit is `<PDU short name>_UB`.

Description in DBC file

```
SG_ frame_18_pdu_UB  : 63|1@1+ (1,0) [0|1] "" ECU1,ECU2
```

Signal presentation in RTICANMM Based on the descriptions above, RTICANMM generates the following signal:

Signal Property	Value
Name	signal_i_pdu_10_UB
Length	1
MinValue	0
MaxValue	1
Offset	0
Scale	1
StartBit	63

Update bit for signal groups

The update bit of a signal group indicates whether data of at least one signal in the signal group has changed.

Description in AUTOSAR system description file The description in the AUTOSAR system description files varies according to the AUTOSAR System Template version used.

- *AUTOSAR 3.x system description file*

```
<SYSTEM-SIGNAL-GROUP>
   <SHORT-NAME>system_signal_group_18</SHORT-NAME>
   <SYSTEM-SIGNAL-REFS>
      <SYSTEM-SIGNAL-REF DEST="SYSTEM-SIGNAL">system_signal_140
      </SYSTEM-SIGNAL-REF>
      <SYSTEM-SIGNAL-REF DEST="SYSTEM-SIGNAL">system_signal_141
      </SYSTEM-SIGNAL-REF>
      <SYSTEM-SIGNAL-REF DEST="SYSTEM-SIGNAL">system_signal_142
      </SYSTEM-SIGNAL-REF>
   </SYSTEM-SIGNAL-REFS>
</SYSTEM-SIGNAL-GROUP>
<I-SIGNAL-TO-I-PDU-MAPPING>
   <SHORT-NAME>i_signal_to_i_pdu_mapping_120</SHORT-NAME>
   <SIGNAL-REF DEST="I-SIGNAL">i_signal_120</SIGNAL-REF>
   <UPDATE-INDICATION-BIT-POSITION>62</UPDATE-INDICATION-BIT-POSITION>
</I-SIGNAL-TO-I-PDU-MAPPING>
<I-SIGNAL>
   <SHORT-NAME>i_signal_120</SHORT-NAME>
   <SYSTEM-SIGNAL-REF DEST="SYSTEM-SIGNAL-GROUP">system_signal_group_18
   </SYSTEM-SIGNAL-REF>
</I-SIGNAL>
<I-SIGNAL>
   <SHORT-NAME>i_signal_116</SHORT-NAME>
   <SYSTEM-SIGNAL-REF DEST="SYSTEM-SIGNAL">system_signal_140
   </SYSTEM-SIGNAL-REF>
</I-SIGNAL>
<I-SIGNAL>
   <SHORT-NAME>i_signal_117</SHORT-NAME>
   <SYSTEM-SIGNAL-REF DEST="SYSTEM-SIGNAL">system_signal_141
   </SYSTEM-SIGNAL-REF>
</I-SIGNAL>
<I-SIGNAL>
   <SHORT-NAME>i_signal_118</SHORT-NAME>
   <SYSTEM-SIGNAL-REF DEST="SYSTEM-SIGNAL">system_signal_142
   </SYSTEM-SIGNAL-REF>
</I-SIGNAL>
```

- *AUTOSAR 4.x system description file*

```
<SYSTEM-SIGNAL-GROUP>
   <SHORT-NAME>system_signal_group_18</SHORT-NAME>
   <SYSTEM-SIGNAL-REFS>
      <SYSTEM-SIGNAL-REF DEST="SYSTEM-SIGNAL">system_signal_140
      </SYSTEM-SIGNAL-REF>
      <SYSTEM-SIGNAL-REF DEST="SYSTEM-SIGNAL">system_signal_141
      </SYSTEM-SIGNAL-REF>
      <SYSTEM-SIGNAL-REF DEST="SYSTEM-SIGNAL">system_signal_142
      </SYSTEM-SIGNAL-REF>
   </SYSTEM-SIGNAL-REFS>
</SYSTEM-SIGNAL-GROUP>
```

```
<I-SIGNAL-TO-I-PDU-MAPPING>
   <SHORT-NAME>i_signal_to_i_pdu_mapping_120</SHORT-NAME>
   <SIGNAL-REF DEST="I-SIGNAL-GROUP">i_signal_120</SIGNAL-REF>
   <UPDATE-INDICATION-BIT-POSITION>62</UPDATE-INDICATION-BIT-POSITION>
</I-SIGNAL-TO-I-PDU-MAPPING>
<I-SIGNAL-GROUP>
   <SHORT-NAME>i_signal_120</SHORT-NAME>
   <SYSTEM-SIGNAL-REF DEST="SYSTEM-SIGNAL-GROUP">system_signal_group_18
   </SYSTEM-SIGNAL-REF>
</I-SIGNAL-GROUP>
<I-SIGNAL>
   <SHORT-NAME>i_signal_116</SHORT-NAME>
   <SYSTEM-SIGNAL-REF DEST="SYSTEM-SIGNAL">system_signal_140
   </SYSTEM-SIGNAL-REF>
</I-SIGNAL>
<I-SIGNAL>
   <SHORT-NAME>i_signal_117</SHORT-NAME>
   <SYSTEM-SIGNAL-REF DEST="SYSTEM-SIGNAL">system_signal_141
   </SYSTEM-SIGNAL-REF>
</I-SIGNAL>
<I-SIGNAL>
   <SHORT-NAME>i_signal_118</SHORT-NAME>
   <SYSTEM-SIGNAL-REF DEST="SYSTEM-SIGNAL">system_signal_142
   </SYSTEM-SIGNAL-REF>
</I-SIGNAL>
```

Description in DBC file

```
SG_ system_signal_group_18_UB : 62|1@1+ (1,0) [0|1] "" ECU1,ECU2
```

Signal presentation in RTICANMM Based on the descriptions above, RTICANMM generates the following signal:

Signal Property	Value
Name	system_signal_group_18_UB
Length	1
MinValue	0
MaxValue	1
Offset	0
Scale	1
StartBit	62

Update bit for signals

The update bit of a signal indicates whether data of the signal the update bit signal belongs to is updated.

Description in AUTOSAR system description file

```
<I-SIGNAL>
   <SHORT-NAME>i_signal_113</SHORT-NAME>
   <SYSTEM-SIGNAL-REF DEST="SYSTEM-SIGNAL">system_signal_17</SYSTEM-SIGNAL-REF>
</I-SIGNAL>
<I-SIGNAL-TO-I-PDU-MAPPING>
   <SHORT-NAME>i_signal_to_i_pdu_mapping_113</SHORT-NAME>
   <PACKING-BYTE-ORDER>MOST-SIGNIFICANT-BYTE-LAST</PACKING-BYTE-ORDER>
   <SIGNAL-REF DEST="I-SIGNAL">system_signal_17</SIGNAL-REF>
   <START-POSITION>48</START-POSITION>
   <UPDATE-INDICATION-BIT-POSITION>59</UPDATE-INDICATION-BIT-POSITION>
</I-SIGNAL-TO-I-PDU-MAPPING>
```

This AUTOSAR description results in an implicit signal with the name 'system_signal_17_UB'.

The general rule for determining the name of a signal update bit is `<ISignal short name>_UB`.

Description in DBC file

```
SG_ system_signal_17_UB : 59|1@1+ (1,0) [0|1] "" ECU1,ECU2
```

Signal presentation in RTICANMM Based on the descriptions above, RTICANMM generates the following signal:

Signal Property	Value
Name	system_signal_17_UB
Length	59
MinValue	0
MaxValue	1
Offset	0
Scale	1
StartBit	1

Related topics

References

General Settings Page (RTICANMM MainBlock)... 100

Working with Generated TRC File Entries

Introduction

The RTI CAN MultiMessage Blockset lets you generate TRC variables for many functions. This section gives you an overview of these functions and the resulting TRC file entries.

Functions with Activated TRC Variable Generation

Introduction

The RTI CAN MultiMessage Blockset lets you activate the generation of TRC variables for many functions. This topic gives an overview of the functions for which you can enable the generation of TRC file entries and provides helpful information for working with them.

Description

The table below serves as a guide to generated TRC file entries. It provides the following information on functions for which the generation of TRC file entries can be generated:

- Default name of generated TRC variable
- Default description
- TRC path (path which is displayed in the variable tree of ControlDesk's Variables controlbar)

> **Tip**
>
> For example, influencing the naming of dynamic attributes in the generated TRC file via the Naming Page (RTICANMM MainBlock) on page 108 can make it difficult to find generated TRC file entries. The TRC path helps you to locate TRC variables in ControlDesk.

- Page of the RTICANMM MainBlock that deals with the function
- Configuration setting on the dialog page that results in generating the TRC variable

Function	Default Name	Default Description	TRC Path[1)] `BusSystems/CAN/<CAN controller>/...`	Page of RTICANMM MainBlock	Activation of TRC Variable Generation
CANName	Chassis_M2	Name of this CAN bus	`BusSystems/CAN/<CAN controller>`	General Settings Page (RTICANMM MainBlock) on page 100	Activated by default
Global Enable	GlobalEnable	Enable %CAN	`GlobalEnable`	Triggering Options Page (RTICANMM MainBlock) on page 136	GlobalEnable_TRC = '1'

Function	Default Name	Default Description	TRC Path[1] BusSystems/CAN/<CAN controller>/...	Page of RTICANMM MainBlock	Activation of TRC Variable Generation
Global TX Enable	GlobalEnableTX	Enable TX of %CAN	`GlobalEnableTX`	Triggering Options Page (RTICANMM MainBlock) on page 136	GlobalEnable TX_TRC = '1'
ECU Enable	%ECU_Enable	Enable %ECU	`ECU_Enable`	Network Node Enable Page (RTICANMM MainBlock) on page 117	Source = 'TRC' or 'BOTH'
Message(TX) ID	%MsgName_ID	Message ID	`<Message>/TX/<Message>_ID`	TX ID Page (RTICANMM MainBlock) on page 191	Manipulation Option = 'TRC' or 'BOTH'
Message(TX) ID format	%MsgName_ IDFormat	Message ID Format	`<Message>/TX/<Message>_IDFormat`	TX ID Page (RTICANMM MainBlock) on page 191	Manipulation Option = 'TRC' or 'BOTH'
Message(TX) Length	%MsgName_Length	Message Length	`<Message>/TX/<Message>_Length`	TX Message Length Page (RTICANMM MainBlock) on page 195	Adjust Option = 'TRC'
Message(TX) Dynamic Length Value	%MsgName_ DynamicLengthValue	Message Dynamic Length Value	`<Message>/TX/<Message>_DynamicLengthValue`	Manipulation Options Page (RTICANMM MainBlock) on page 190	Select Activate dynamic MessageLength.
Message(TX) Dynamic Length Countdown	%MsgName_DynamicLengthCountdown	Message Dynamic Length Countdown	`<Message>/TX/<Message>_DynamicLengthCountdown`	Manipulation Options Page (RTICANMM MainBlock) on page 190	Select Activate dynamic MessageLength.
Message(TX) CheckEnable	%MsgName_ CheckEnable	Check enabled %MsgName	`<Message>/TX/<Message>_CheckEnable`	TX Messages Page (RTICANMM MainBlock) on page 128	Select a message.
Message(TX) Enable	%MsgName_Enable	Enable %MsgName	`<Message>/TX/<Message>_Enable`	Message Enable Page (RTICANMM MainBlock) on page 140	Source = 'TRC' or 'BOTH'
Message(TX) Timeout	%MsgName_Timeout	Timeout for %MsgName	`<Message>/TX/<Message>_Timeout`	TX Timeout Enable Page (RTICANMM MainBlock) on page 161	Select a message. The timeout mechanism is available only if Enable Timeout = '1' on the Triggering Options Page (RTICANMM

Function	Default Name	Default Description	TRC Path[1)] `BusSystems/CAN/<CAN controller>/...`	Page of RTICANMM MainBlock	Activation of TRC Variable Generation
					MainBlock) on page 136.
Message(TX) Cyclic	%MsgName_ PeriodicalOn	Activate cyclic transmission for %MsgName	`<Message>/TX/<Message>_ PeriodicalOn`	Message Cyclic Page (RTICANMM MainBlock) on page 144	Source = 'TRC' or 'BOTH'
Message(TX) Kickout	%MsgName_Kickout	Kickout %MsgName	`<Message>/TX/<Message>_ Kickout`	Message Kickout Page (RTICANMM MainBlock) on page 148	Source = 'TRC' or 'BOTH'
Message(TX) CycleTime	%MsgName_ CycleTime	CycleTime %MsgName	`<Message>/TX/<Message>_ CycleTime`	TX Cycle Time Page (RTICANMM MainBlock) on page 154	Select a message.
Message(TX) DelayTime	%MsgName_ DelayTime	DelayTime %MsgName	`<Message>/TX/<Message>_ DelayTime`	TX Delay Time Page (RTICANMM MainBlock) on page 156	Select a message.
Message(TX) BaseTime	%MsgName_ BaseTime	BaseTime %MsgName	`<Message>/TX/<Message>_ BaseTime`	Base/Update Messages Page (RTICANMM MainBlock) on page 158	Select a message.
Message(TX) UpdateTime	%MsgName_ UpdateTime	UpdateTime %MsgName	`<Message>/TX/<Message>_ UpdateTime`	Base/Update Messages Page (RTICANMM MainBlock) on page 158	Select a message.
Message(TX) UpdateNumber	%MsgName_ UpdateNum	UpdateNum %MsgName	`<Message>/TX/<Message>_ UpdateNum`	Base/Update Messages Page (RTICANMM MainBlock) on page 158	Select a message.
Message(TX) VariationIndex	%MsgName_ VariationIndex	VariationIndex %MsgName	`<Message>/TX/<Message>_ VariationIndex`	Message Variations Page (RTICANMM MainBlock) on page 198	Source = 'TRC'
Message(TX) CRCEnable	%MsgName_crc	Enable CRC %MsgName	`<Message>/TX/<Message>_ crc`	Checksum Messages Page (RTICANMM MainBlock) on page 205	Select a message with assigned checksum algorithm.
Message(TX) CRCAlgorithm	%MsgName_crc_ algorithm	Algorithm Index of CRC %MsgName	`<Message>/TX/<Message>_ crc_algorithm`	Checksum Messages Page (RTICANMM MainBlock) on page 205	Select a message with assigned checksum algorithm.
Message(TX) CRC dynamic Algorithm	%MsgName_dyn_ algorithm	Dynamic CRC Algorithm %MsgName	`<Message>/TX/<Message>_ dyn_algorithm`	Manipulation Options Page (RTICANMM MainBlock) on page 190	Select Activate dynamic CRC.

Function	Default Name	Default Description	TRC Path[1)] `BusSystems/CAN/<CAN controller>/...`	Page of RTICANMM MainBlock	Activation of TRC Variable Generation
Message(TX) CRC dynamic Countdown	%MsgName_dyn_ countdown	Dynamic countdown of CRC %MsgName	`<Message>/TX/<Message>_dyn_countdown`	Manipulation Options Page (RTICANMM MainBlock) on page 190	Select Activate dynamic CRC.
Message(TX) CCEnable(1)	%MsgName_CC_ preCRC	Enable CC %MsgName pre CRC	`<Message>/TX/<Message>_CC_preCRC`	Custom Code Page (RTICANMM MainBlock) on page 211	Option = 'pre CRC'
Message(TX) CCEnable(2)	%MsgName_CC_ postCRC	Enable CC %MsgName post CRC	`<Message>/TX/<Message>_CC_postCRC`	Custom Code Page (RTICANMM MainBlock) on page 211	Option = 'post CRC'
Message(TX) Status	%MsgName_status	%MsgName Status	`<Message>/TX/<Message>_status`	TX Messages Page (RTICANMM MainBlock) on page 128	Select a message.
Message(TX) SigManipulation	%MsgName_ sigmanip	Enable %MsgName userdefined SignalManipulation	`<Message>/TX/<Message>_sigmanip`	Custom Signal Manipulation Page (RTICANMM MainBlock) on page 231	Select a message.
Message(TX) Raw Switch	%MsgName_RAW_ switch	%MsgName RAW data switch	`<Message>/TX/<Message>_RAW_switch`	TX Raw Data Page (RTICANMM MainBlock) on page 178	Activate raw data option = 'Active' and Source of raw data = 'TRC'
Message(TX) Raw Byte	%MsgName_RAW_ byte	%MsgName Byte	`<Message>/TX/<Message>_RAW_byte_[1…8]`	TX Raw Data Display Page (RTICANMM MainBlock) on page 180	Select a message.
Signal(TX) Switch	%SigName_Switch	Switch %SigName	`<Message>/TX/<Signal>_Switch`	TX Page (RTICANMM MainBlock) on page 214	Select two or more signal manipulation options (e.g., TRC and Inport).
Signal(TX) Switch in GatewayMode	%SigName_ GwSwitch	Switch %SigName in GatewayModus	`<Message>/TX/<Signal>_GWSwitch`	Signal Default Manipulation Page (RTICANMM MainBlock) on page 244	Select Enable Manipulation.
Signal(TX) Constant	%SigName	%SigDesc	`<Message>/TX/<Signal>`	TX Page (RTICANMM MainBlock) on page 214	Select a signal.
Signal(TX) GatewaySwitch	%SigName_GW	Switch GatewayModus for %SigName	`<Message>/TX/<Signal>_GW`	Gateway Signals Page (RTICANMM MainBlock) on page 248	Select a signal.

Function	Default Name	Default Description	TRC Path[1)] `BusSystems/CAN/<CAN controller>/...`	Page of RTICANMM MainBlock	Activation of TRC Variable Generation
Signal(TX) DynValue	%SigName_dynvalue	Dynamic Value for %SigName	`<Message>/TX/<Signal>_dynvalue`	Dynamic Signal Values Page (RTICANMM MainBlock) on page 233	Select a signal.
Signal(TX) DynCountdown	%SigName_dyncount	Countdown for Dynamic Value	`<Message>/TX/<Signal>_dyncount`	Dynamic Signal Values Page (RTICANMM MainBlock) on page 233	Select a signal.
Message(RX) ID	%MsgName_ID	Message ID	`<Message>/RX/<Message>_ID`	TX Messages Page (RTICANMM MainBlock) on page 128	Select a message.
Message(RX) ID format	%MsgName_IDFormat	Message ID format	`<Message>/RX/<Message>_IDFormat`	TX Messages Page (RTICANMM MainBlock) on page 128	Select a message.
Message(RX) Length	%MsgName_Length	Message Length	`<Message>/RX/<Message>_Length`	TX Messages Page (RTICANMM MainBlock) on page 128	Select a message.
Message(RX) Status	%MsgName_status	%MsgName Status	`<Message>/RX/<Message>_status`	TX Messages Page (RTICANMM MainBlock) on page 128	Select a message.
Message(RX) Time	%MsgName_time	%MsgName Time	`<Message>/RX/<Message>_time`	TX Messages Page (RTICANMM MainBlock) on page 128	Select a message.
Message(RX) Deltatime	%MsgName_dt	%MsgName DeltaTime	`<Message>/RX/<Message>_dt`	TX Messages Page (RTICANMM MainBlock) on page 128	Select a message.
Message(RX) Error	%MsgName_error	%MsgName Error	`<Message>/RX/<Message>_error`	RX Error Display Page (RTICANMM MainBlock) on page 185	Select a message.
Message(RX) Counter	%MsgName_counter	%MsgName Counter	`<Message>/TX/<Message>_counter_[attribute]`	Counter Page (RTICANMM MainBlock) on page 240	Select a signal.
Message(RX) Raw Byte	%MsgName_RAW_byte	%MsgName Byte	`<Message>/RX/<Message>_RAW_byte_[1…8]`	RX Raw Data Display Page (RTICANMM MainBlock) on page 182	Select a message.
Signal(RX) Name	%SigName	%SigDesc	`<Message>/RX/RX_<Message>_<Signal>`	RX Messages Page (RTICANMM	Select a message.

Function	Default Name	Default Description	TRC Path[1)]`BusSystems/CAN/<CAN controller>/...`	Page of RTICANMM MainBlock	Activation of TRC Variable Generation
				MainBlock) on page 130	

1) This is the path which is displayed in the variable tree of ControlDesk's Variables controlbar.

Related topics

References

Naming Page (RTICANMM MainBlock)... 108

Limitations

Where to go from here

Information in this section

Limitations with RTICANMM

RTI CAN MultiMessage Blockset

The following limitations apply to the RTI CAN MultiMessage Blockset:

- The configuration file supports only messages whose name does not begin with an underscore.
- Do not use the RTI CAN MultiMessage Blockset and the RTI CAN Blockset for the same CAN controller.
- Do not use the RTI CAN MultiMessage Blockset in enabled subsystems, triggered subsystems, configurable subsystems, or function-call subsystems. As an alternative, you can disable the entire RTI CAN MultiMessage Blockset by switching the CAN controller variant, or by setting the **GlobalEnable** triggering option. This option is available on the Triggering Options Page (RTICANMM MainBlock) on page 136.
- Do not run the RTI CAN MultiMessage Blockset in a separate task.
- Do not copy blocks of the RTI CAN MultiMessage Blockset. To add further blocks of the RTI CAN MultiMessage Blockset to a model, always take them directly from the **rticanmmlib** library. To transfer settings between two MainBlocks or between two Gateway blocks, invoke the **Save Settings** and **Load Settings** commands from the **Settings** menu (refer to RTICANMM MainBlock or RTICANMM Gateway on page 280).
- The RTI CAN MultiMessage Blockset is not included in the RTI update mechanism and is not updated when you open a model with an older version. To update the RTI CAN MultiMessage Blockset, invoke **Create S-Function for All RTICANMM Blocks** from the **Options** menu of the RTICANMM GeneralSetup on page 38.

As an alternative, you can create new S-functions for all RTICANMM blocks manually (use the following order):

1. RTICANMM GeneralSetup on page 38
2. RTICANMM ControllerSetup on page 43
3. RTICANMM MainBlock on page 68
4. RTICANMM Gateway on page 280

- Model path names with multi-byte character encodings are not supported.
- Mode signals with opaque byte order format that are longer than 8 bits are not supported.
- The RTI CAN MultiMessage Blockset generates data structures on the basis of the relevant element names specified in the database file. The length of an element name is limited to 56 characters. If an element name exceeds this limit, the RTI CAN MultiMessage Blockset shortens the name to 56 characters, using a checksum to ensure name uniqueness, and makes an entry in the log file.

  The following list shows the element types whose maximum name length must not exceed 56 characters:
  - Messages
  - Signals
  - UpdateBit signals
  - Mode signals
  - Nodes
- Simulink can store design data that your model uses in a data dictionary as a persistent repository. Data dictionaries are not supported by the RTI CAN MultiMessage Blockset.

FIBEX 3.1, FIBEX 4.1, FIBEX 4.1.1, or FIBEX 4.1.2 file as the database

The RTI CAN MultiMessage Blockset does not support multiple computation methods for signals. If several CompuMethods are defined for a signal in the FIBEX file, the RTI CAN MultiMessage Blockset uses the first linear computation method it finds for the signal.

MAT file as the database

In the RTI CAN MultiMessage Blockset, the length of signal names is restricted to 32 characters. However, MATLAB allows longer signal names. When MATLAB entries are mapped to the signals in RTICANMM, the signal names are truncated at the end and supplemented by a consecutive number, if necessary. To ensure that unchanged signal names are used in the RTI CAN MultiMessage Blockset, the signal names in the Simulink model must not exceed 32 characters.

AUTOSAR system description file as the database

- The RTI CAN MultiMessage Blockset does not support the following features that can be defined in an AUTOSAR 3.2.2, 4.0.3, 4.1.1, 4.1.2, 4.2.1, 4.2.2, 4.3.0, 4.3.1, 4.4.0, or AUTOSAR Classic R19-11 or R20-11 system description file:
  - Partial networking (There are some exceptions: Partial networking is supported for the MicroAutoBox II equipped with a DS1513 I/O Board, the MicroLabBox, and dSPACE hardware that is equipped with DS4342 CAN FD Interface Modules.)
  - Unit groups
  - Segment positions for MultiplexedIPdus
  - End-to-end protection for ISignalGroups
- The RTI CAN MultiMessage Blockset does not support the new features of AUTOSAR Release 4.4.0 and AUTOSAR Classic Platform Release R19-11 and R20-11.
- When you work with an AUTOSAR ECU Extract as the database, the RTI CAN MultiMessage Blockset does not support frames with multiplexed IPDUs whose PDUs are only partially included (e.g., the imported ECU Extract contains only their dynamic parts while their static parts are contained in another ECU Extract).

Limitations for container IPDUs

- The RTI CAN MultiMessage Blockset does not support nested container IPDUs.
- For contained IPDUs that are included in container IPDUs with a dynamic container layout, the RTI CAN MultiMessage Blockset does not support the long header type. For the `ContainerIpduHeaderType` AUTOSAR attribute, the RTI CAN MultiMessage Blockset supports only the `SHORT_HEADER` value.
- For the `ContainedIpduCollectionSemantics` AUTOSAR attribute, the RTI CAN MultiMessage Blockset supports the `QUEUED` and `LAST_IS_BEST` values. However, when a container IPDU with a queued semantics is received that contains multiple instances of a contained IPDU, only the last received instance is displayed.
- For the `RxAcceptContainedIpdu` AUTOSAR attribute, the RTI CAN MultiMessage Blockset does not support the `ACCEPT_CONFIGURED` value for container IPDUs, which allows only a certain set of contained IPDUs in a container IPDU.
- The RTI CAN MultiMessage Blockset supports TX message length manipulation (static and dynamic length manipulation) only for contained IPDUs that are included in container IPDUs with a dynamic container layout, not for contained IPDUs included in container IPDUs with a static container layout and also not for dynamic and static container IPDUs.
- The RTI CAN MultiMessage Blockset lets you manipulate the length of a contained IPDU that is included in container IPDUs with a dynamic container layout as long as the IPDU has not yet been written to a container IPDU. Once a contained IPDU is written to its container IPDU, the length manipulation options no longer have any effect on the instance of the contained IPDU that is currently triggered and written to the container IPDU. But the length manipulation options take effect again when the contained IPDU is triggered the next time. Length manipulation is not supported for contained IPDUs that are included in container IPDUs with a static container layout.

- The RTI CAN MultiMessage Blockset supports TX message ID manipulation only for contained IPDUs that are included in container IPDUs with a dynamic container layout, not for contained IPDUs that are included in container IPDUs with a static container layout and also not for dynamic and static container IPDUs. By activating the TX message ID manipulation option for contained IPDUs in dynamic container IPDUs, you actually manipulate the `SHORT_HEADER` of the contained IPDUs.
- The RTI CAN MultiMessage Blockset supports neither TX signal manipulation nor gateway signal manipulation for container IPDU signals.
- When you gateway messages using the RTICANMM Gateway block, you cannot exclude contained IPDUs from being gatewayed. Excluding container IPDUs is possible.

Limitations for secure onboard communication

- The RTI CAN MultiMessage Blockset does not support counters as freshness values. Only time stamp values can be used as freshness values.
- Cryptographic IPDUs are not displayed on the dialog pages of the RTICANMM MainBlock.
- The RTI CAN MultiMessage Blockset supports secured PDU headers only for container IPDUs with a dynamic container layout. For all other IPDU types, secured PDU headers are not supported.

Limitations for global time synchronization

- The RTI CAN MultiMessage Blockset does not support the simulation of a global time master.
- The RTI CAN MultiMessage Blockset does not support offset GTS messages (offset synchronization messages (OFS messages) and offset adjustment messages (OFNS messages)).
- GTS messages are not displayed on the Checksum Messages Page (RTICANMM MainBlock). In the case of secured GTS messages, a predefined checksum algorithm is used if the **GTS** manipulation option is selected on the Signal Default Manipulation Page (RTICANMM MainBlock) for the SyncSecuredCRC and FupSecuredCRC signals.
- The RTI CAN MultiMessage Blockset does not support switching between the secured and the unsecured GTS message types at run time, i.e., you cannot switch from a CRC-secured SYNC and FUP message pair to an unsecured message pair, or vice versa.
- If multiple time slaves are defined for a GTS message, only the highest `FupTimeout` value is imported and can be used during run time.
- Only valid pairs of SYNC and FUP messages can update the time in a time base manager instance. SYNC and FUP messages form a valid pair if they meet the following conditions:
  - Both messages use the same CAN identifier and the same ID format.
  - Both messages use the same time domain identifier.
  - Both messages must be CRC-secured or both must be unsecured.
- For signals of GTS messages, the RTI CAN MultiMessage Blockset only supports Global time synchronization and Constant as TX signal manipulation options, where Global time synchronization is set as default option. Other TX signal manipulation options are not supported for signals of GTS messages.

- The RTI CAN MultiMessage Blockset does not support gateway signal manipulation for signals of GTS messages.
- For the `crcValidated` AUTOSAR attribute, the RTI CAN MultiMessage Blockset does not support the following values:
  - `crcIgnored`
  - `crcOptional`
- Clearing the **Use specific data types** checkbox on the Code Options Page (RTICANMM MainBlock) of the RTICANMM MainBlock has no effect on GTS messages. GTS messages always use specific data types.

Visualization with the Bus Navigator

The current version of the RTI CAN MultiMessage Blockset supports visualization with the **Bus Navigator** in ControlDesk 4.2.1 or later. You cannot work with earlier versions of ControlDesk in connection with applications created with the current version of the RTI CAN MultiMessage Blockset.

Limitations with CAN FD

Limitations

The following limitations apply to the CAN FD support of the RTI CAN MultiMessage Blockset:

- To use CAN FD, you must provide a suitable DBC or AUTOSAR system description file containing descriptions in CAN FD format.
- For a MicroAutoBox II and DS4342, CAN FD messages cannot cause a transceiver wake-up.
- If message format manipulation and message length manipulation are enabled for a message, and if you switch between classic CAN and CAN FD formats during run time, the range of possible message lengths is not adjusted, neither upwards, nor downwards. The upper limit for the message length remains unchanged.

Limitations with J1939-Support

Limitations

The following limitations apply to the J1939 support of the RTI CAN MultiMessage Blockset:

- The J1939 support for the RTI CAN MultiMessage Blockset requires a separate license.
- To use J1939, you must provide a J1939-compliant DBC file.
- Though most messages are already defined in the J1939 standard, you must specify the required messages in your DBC file.

- When you gateway messages, J1939 network management (address claiming) is not supported. This limitation applies to gatewaying via RTICANMM MainBlocks and via RTICANMM Gateway block.
- When you gateway J1939 messages via an RTICANMM Gateway block, multipacket messages cannot be added to the filter list. This means that J1939 messages longer than 8 bytes cannot be excluded from being gatewayed.
- For J1939 messages, the CRC option is limited to the first eight bytes.
- For J1939 messages, the custom code option is limited to the first eight bytes.
- Peer-to-peer communication for J1939 messages longer than 8 bytes via RTS/CTS is supported only for receiving network nodes whose simulation type is set to 'simulated' or 'external'.
- CAN messages with extended identifier format and also J1939 messages use a 29-bit message identifier. Because the RTI CAN MultiMessage Blockset cannot differentiate between the two message types on the CAN bus, working with extended CAN messages and J1939 messages on the same bus is not supported.
- For J1939 messages, the manipulation of the PGN is not supported.

Migration

Migrating from an Earlier Version of the RTI CAN MultiMessage Blockset

Working with models from earlier RTI CAN MultiMessage Blockset versions

To reuse a model created with an earlier RTI CAN MultiMessage Blockset version, you must update the S-functions for all the RTICANMM blocks and save the model before modifying the CAN configuration.

To create new S-functions for all the RTICANMM blocks in a model in one step, you can perform one of the following actions after opening the model:

- In the MATLAB Command Window, enter `rtimmsu_update('System', bdroot)`.

  For more information on the command and its options, enter `help rtimmsu_update` in the MATLAB Command Window.
- Select the Create S-Function for all CAN Blocks command from the Options menu of the RTICANMM GeneralSetup block.

For more information, refer to Limitations with RTICANMM on page 309.

Discontinuation of FIBEX 2.0 support

As of RTI CAN MultiMessage Blockset 5.6, FIBEX 2.0 files can no longer be specified as the database for an RTICANMM MainBlock.

Changed evaluation of CanFrameBehavior attributes

In AUTOSAR and FIBEX files, the `CanFrameTxBehavior` and/or `CanFrameRxBehavior` attributes of a message can be defined to specify whether the message is to be treated as a CAN FD message or classic CAN 2.0 message. With the RTI CAN MultiMessage Blockset 5.4, the evaluation and application of these attribute values have changed:

- Up to and including the RTI CAN MultiMessage Blockset 5.3, only the `CanFrameTxBehavior` value was considered. If the `CanFrameTxBehavior` attribute was not set for a message in the database, it was assumed for both directions that it was not a CAN FD message.
- As of RTI CAN MultiMessage Blockset 5.4, the `CanFrameRxBehavior` attribute is also evaluated if required:
  - If the `CanFrameTxBehavior` attribute is defined for a message in the database file, RTICANMM uses this setting for the message on the CAN bus for both directions, i.e., for sending and receiving the message.
  - If the `CanFrameTxBehavior` attribute is not defined in the database for a message, RTICANMM uses the message's `CanFrameRxBehavior` setting for sending or receiving the message.

CAN FD: Using checksum algorithms originally developed for application containing only classic CAN messages

Checksum algorithms that were originally developed for an application containing CAN messages cannot be reused for applications that contain CAN FD messages, because CAN FD includes new message types and longer data fields. Existing checksum algorithms can still be used for applications that contain only classic CAN messages. For CAN FD applications, you must adapt the checksum algorithms.

Compiler messages when using code generated by an RTI CAN MultiMessage Blockset version < 4.0

If you use code that was generated by an RTI CAN MultiMessage Blockset version < 4.0, several compiler warning messages that contain the phrase `<<argument of type "can_tp1_canChannel *" is incompatible with parameter of type "DsTCanCh">>` will be displayed during the build process of a simulation model. This is due to a modified data type. These warnings can be ignored and disappear after you use the current blockset version to generate the RTICANMM code again.