RTI XCP on Ethernet Blockset

Manual

For RTI XCP on Ethernet Blockset 1.2.12

Release 2021-A – May 2021

dSPACE

How to Contact dSPACE

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

How to Contact dSPACE Support

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

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

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

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

Software Updates and Patches

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

Important Notice

Contents

About This Document

Content

This document provides access to the information you need to implement the XCP on Ethernet Real-Time Service in a real-time application for dSPACE hardware.

Symbols

dSPACE user documentation uses the following symbols:

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

Naming conventions

dSPACE user documentation uses the following naming conventions:

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

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

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.

Introduction to the XCP on Ethernet Real-Time Service

Where to go from here

Information in this section

Basics on the XCP on Ethernet Real-Time Service

ASAM MCD standard

The ASAM MCD standard defines interfaces and data formats for communication with and diagnosis of automotive electronic control units. ASAM stands for the *Association for Standardization of Automation and Measuring Systems*. MCD stands for the *Measurement, Calibration and Diagnosis* section of the ASAM.

For information on these standards, refer to http://www.asam.net.

Controller development and calibration

Principle The RTI XCP on Ethernet Blockset lets you interface dSPACE real-time hardware via measurement and calibration systems.

The measurement and calibration system reads the A2L file that was generated for your specific dSPACE real-time system and configures the XCP on Ethernet

Real-Time Service to acquire data from the memory of the dSPACE hardware. The data is then sent to the measurement and calibration system via Ethernet. By implementing the XCP on Ethernet Real-Time Service in an application and generating an A2L file, you can perform calibration tasks while the controller is still under development.

A2L file generation The A2L file that is required by the measurement and calibration system is generated during the build process. For details, refer to the Simulink Model A2L File Generation Manual 🕮.

Download to the flash memory via XCP on Ethernet

If you work with MicroAutoBox II, applications can be downloaded to the flash memory via XCP on Ethernet. Refer to Download to the Flash Memory via XCP on Ethernet on page 32.

System architecture

A system for simultaneous controller development and calibration mainly consists of the following components:

- A host PC
- dSPACE real-time hardware (refer to System Requirements and System Setup on page 8)
- A measurement and calibration system supporting XCP on Ethernet. Typically, the measurement and calibration system is installed on the host PC.

Communication between the dSPACE real-time hardware and the measurement and calibration system is performed via the Universal Measurement and Calibration Protocol (XCP).

Demo model

The software includes the demo model `rtiethxcp_demo.slx`, which is available via the RTI XCP on Ethernet Blockset library. Refer to How to Prepare the Demo Model on page 10.

Related topics

HowTos

How to Prepare the Demo Model 10

System Requirements and System Setup

Introduction

To use the XCP on Ethernet Real-Time Service, you have to check the hardware and software requirements.

Hardware requirements

The following hardware is required:

- One of the following dSPACE real-time hardware platforms:
  - DS1401 (MicroAutoBox II)
  - MicroLabBox
  - DS1007
- You must establish a connection between the dSPACE real-time hardware and the host PC via the Ethernet host interface of the dSPACE real-time hardware.

For information on how to install and configure dSPACE real-time hardware, refer to the following documentation:

- DS1007 Hardware Installation and Configuration 🕮 and PHS Bus System Hardware Reference 🕮
- MicroAutoBox II Hardware Installation and Configuration 🕮
- MicroLabBox Hardware Installation and Configuration 🕮

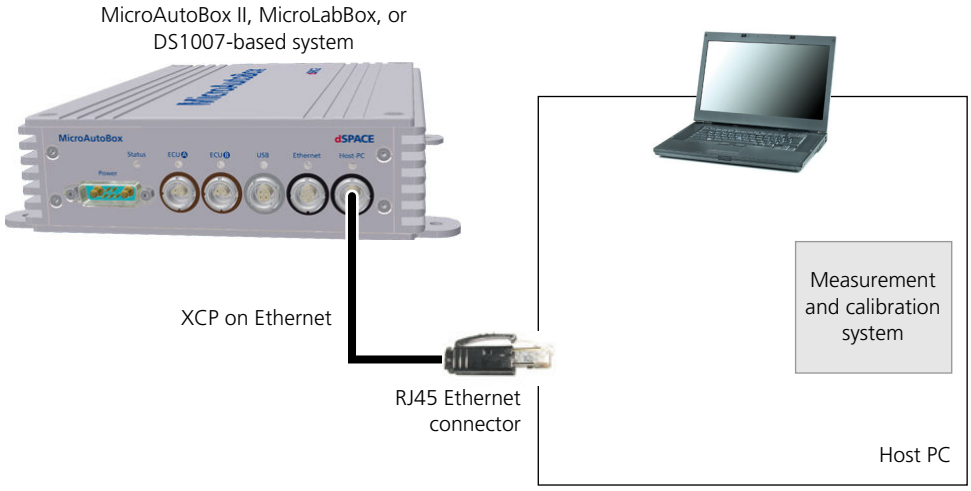
Software requirements

The measurement and calibration system used to access dSPACE real-time hardware must support XCP on Ethernet (UDP/IP or TCP/IP).

To download an application to dSPACE real-time hardware (global memory or flash memory), at least the license-free Loader Version of ControlDesk must be installed on the host PC.

System setup

The illustration below shows the system setup.



Related topics

Basics

Introduction to the XCP on Ethernet Real-Time Service..... 7

How to Prepare the Demo Model

Objective

The software includes the `rtiethxcp_demo.slx` demo model, which is available via the RTI XCP on Ethernet Blockset library. When you open the demo model, it is automatically configured for your current default board. The following boards are supported:

- DS1007
- MicroAutoBox II
- MicroLabBox

Method

To prepare the demo model

1. Open MATLAB.
2. Type `rtiethxcp` at the MATLAB command prompt.
   The RTI XCP on Ethernet Blockset opens.
3. Double-click the Demo button.
   The demo model opens.
4. Press `Ctrl` + `B` to build the real-time application.
   During the build process, an A2L file named `rtiethxcp_demo.a2l` is generated in the demo folder. In addition, the code for the XCP on Ethernet Real-Time Service is integrated into the real-time application. For further information on the build process, refer to Building and Downloading the Model (RTI and RTI-MP Implementation Guide 🕮).
5. Connect the dSPACE real-time hardware to the host PC.
6. Load the application to the flash memory of your dSPACE real-time hardware. For details, refer to Handling an Application for the Flash Memory on page 30.

Result

The demo model is prepared for use with an measurement and calibration system. In your measurement and calibration system, you can load the generated A2L file and access the real-time application for measurement and calibration purposes via XCP on Ethernet (TCP/IP or UDP/IP).

Note

If the measurement and calibration system is not installed on the host PC, you must edit the generated A2L file manually. Scroll to the two lines containing the `ADDRESS "127.0.0.1"` entry, and overwrite the IP addresses with the IP address of your host PC.

Related topics

Basics

Implementing the XCP on Ethernet Real-Time Service

Where to go from here

Information in this section

Basics on Implementing the XCP on Ethernet Real-Time Service

Introduction

The XCP on Ethernet Real-Time Service provides access to the dSPACE real-time hardware for measurement and calibration systems to make variables – parameters and signals – that are running on the dSPACE platform available to an measurement and calibration system. The XCP on Ethernet Real-Time Service,

which is based on the Universal Measurement and Calibration Protocol (XCP), is implemented in the application running on your dSPACE real-time hardware.

Features of the service

The XCP on Ethernet Real-Time Service provides the following features:

- Implementation of XCP version 1.0
- Measurement and calibration of the variables of the real-time application running on the dSPACE hardware
- Acquisition of data in polling and event-triggered mode (refer to Basics on Data Acquisition with the XCP on Ethernet Real-Time Service on page 15)
- Compliance with the ASAP1a standard defined by the *Arbeitskreis zur Standardisierung von Applikationssystemen (ASAP)*.

Accessing the XCP on Ethernet Real-Time Service

To access the DS1007, MicroAutoBox II, or MicroLabBox, its Ethernet host interface must be connected to the host PC.

Your measurement and calibration system on the host PC accesses the XCP on Ethernet Real-Time Service via the specified Ethernet port(s) of the host interface of the DS1007, MicroAutoBox II, or MicroLabBox.

Note

The DS1007 and MicroLabBox have a processor with two cores, each of which can run one application process of the real-time application.

You have to:

- Implement the XCP on Ethernet Real-Time Service in *each* application process.
- Specify *different* Ethernet ports for each application process.

Ethernet connection configuration In the RTIEthXCP Setup block:

- Select **Ethernet Host Interface** as the **XCP interface** setting. Refer to Dialog Settings (RTIEthXCP Setup Block) on page 23.
- Specify the Ethernet port(s) and the Ethernet address of the host interface of the DS1007, MicroAutoBox II, or MicroLabBox. Refer to Dialog Settings (RTIEthXCP Setup Block) on page 23.

Related topics

Basics

Basics on Data Acquisition with the XCP on Ethernet Real-Time Service 15
Implementing the XCP on Ethernet Real-Time Service 13

HowTos

How to Configure Data Acquisition 19

Basics on Data Acquisition with the XCP on Ethernet Real-Time Service

Introduction

A measurement and calibration system can acquire data from the dSPACE real-time hardware either in *polling mode* or in *data acquisition mode*.

Polling mode

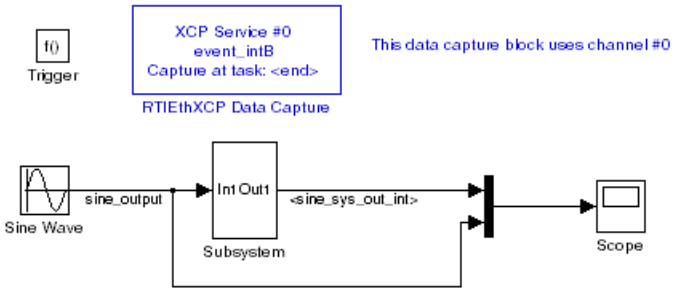
In polling mode, the measurement and calibration system sends a CTO command packet (CMD) to the dedicated Ethernet port configured in the RTIEthXCP Setup Block at a specific rate. The XCP on Ethernet Real-Time Service implemented on the dSPACE real-time hardware receives and acknowledges it, and sends the requested information back to the measurement and calibration system.

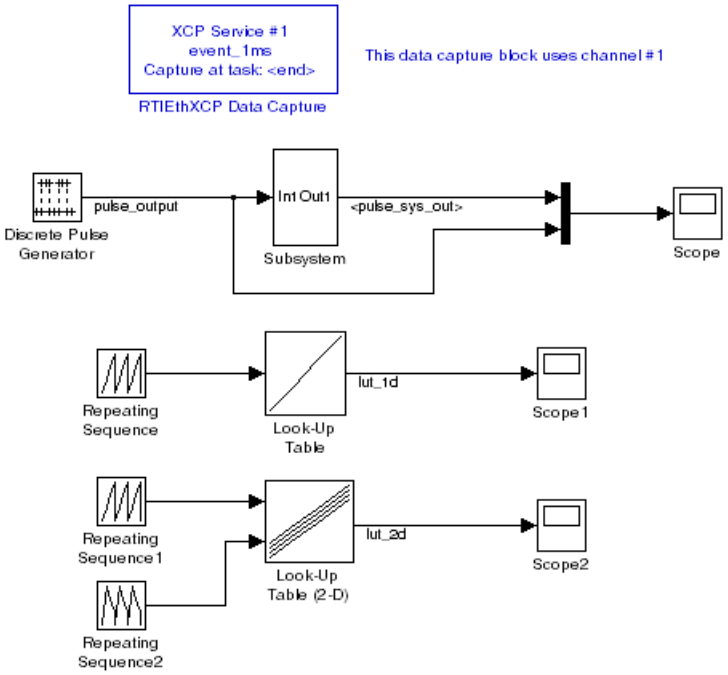
Handshake protocol A handshake protocol is used to initiate a connection between the measurement and calibration system and the dSPACE real-time hardware, configure the XCP on Ethernet Real-Time Service, and acquire data in polling mode.

Whenever a message sent to the port number configured in the RTIEthXCP Setup block is received, the command interpreter of the XCP Real-Time Service is called. The message is decoded and a command response object (RES) is sent back to the measurement and calibration system the message came from to acknowledge the command.

Event-triggered acquisition mode

In event-triggered mode, the dSPACE real-time hardware is configured to acquire and send the requested data without interaction from the measurement and calibration system. Data acquisition is triggered by one or more RTIEthXCP Data Capture blocks, which you can place in any subsystem of your Simulink model.





Event channels In your measurement and calibration system, you can assign variables to an event channel. To acquire these variables from the dSPACE real-time hardware, you have to set up the event channels in the RTIEthXCP Data Capture blocks of your Simulink model. Each RTIEthXCP Data Capture block corresponds to one event channel identified by a unique event channel number and event channel name. The variables assigned to a specific event channel are then acquired with the sample time given in the corresponding RTIEthXCP Data Capture block. The time the data is captured can be configured to the begin or the end of the subsystem execution.

Sample time You can specify a different sample time for each RTIEthXCP Data Capture block in your model – either a multiple of the model's sample time or -1. If you specify -1, the sample time of the event channel is inherited from the subsystem in which the RTIEthXCP Data Capture block resides. Data can be sampled at the beginning or at the end of the subsystem. Therefore, the results of the current sampling step can be observed.

Note

The sample time must be a multiple of the model simulation step.

Data acquisition lists Information on the variables to be acquired is stored in data acquisition (DAQ) lists. Several DAQ lists can be assigned to one event channel and sampled at the same rate.

Two types of data acquisition lists are supported:

- Static DAQ lists
- Dynamic DAQ lists

In case of static DAQ lists, the number of object descriptor tables (ODT) and object descriptor table entries (ODT entries) is preconfigured for each DAQ list.

Avoiding overruns The send queue is 140 kByte long to avoid overruns when a lot of data is sampled within a short time. If the queue is full, the whole DAQ list is dropped. The queue is checked again for the remaining DAQ lists assigned to the current event channel. This means that the remaining data is sent if the queue is no longer full. To avoid overruns, increase the sample time of the event channels or reduce the number of variables to be sampled.

Related topics

HowTos

How to Configure Data Acquisition....19
How to Implement the XCP on Ethernet Real-Time Service....17

References

RTIEthXCP Data Capture....25

How to Implement the XCP on Ethernet Real-Time Service

Objective

Implementing the XCP on Ethernet Real-Time Service in a real-time application lets measurement and calibration systems access the variables of the application.

Restrictions

- Only one instance of the XCP on Ethernet Real-Time Service can run in an application process at a time.
- Multicore (MC) and multiprocessor applications consist of several application processes. Each application process requires a separate XCP on Ethernet Real-Time Service instance for being accessed via XCP.
- In a multicore (MC) environment, each XCP on Ethernet Real-Time Service instance - one instance per application process - requires a separate Ethernet port number.

Method

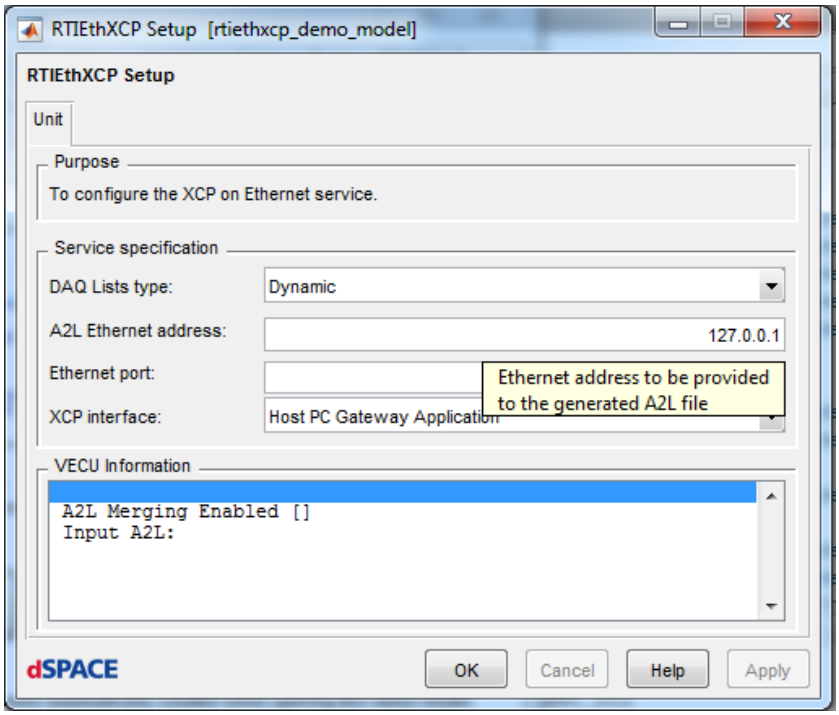
To implement the XCP on Ethernet Real-Time Service

Note

(Relevant only to DS1007 and MicroLabBox models) Perform the following steps individually for each application process.

1 In the Simulink model, insert one RTIEthXCP Setup block.

2 Double-click the RTIEthXCP Setup block.
The RTIEthXCP Setup block dialog opens.



3 Specify to use static or dynamic DAQ lists.

4 Specify the Ethernet address and port used to access the XCP on Ethernet Real-Time Service.

Note

(Relevant only to DS1007 and MicroLabBox models) Specify *different* Ethernet ports for each application process.

5 Select Ethernet Host Interface as the XCP interface.

Result

You have implemented the XCP on Ethernet Real-Time Service in your real-time application.

Related topics

HowTos

How to Configure Data Acquisition..... 19

References

Code Generation Dialog (Model Configuration Parameters Dialogs) (RTI and RTI-MP Implementation Reference 🕮)
RTIEthXCP Setup Block..... 22

How to Configure Data Acquisition

Objective

To trigger data acquisition, you have to implement RTIEthXCP Data Capture blocks in the application of the dSPACE real-time hardware.

Basics

You can place RTIEthXCP Data Capture blocks in any subsystem of your Simulink model.

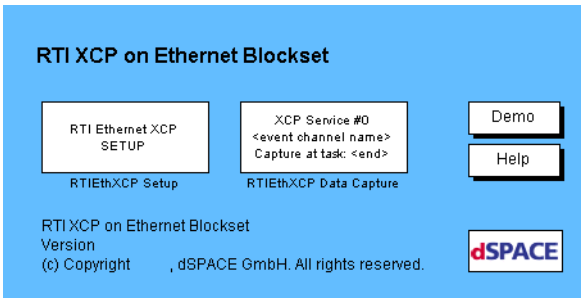
Via this block, you can specify the event channel number and name, the time when data is to be captured and the sample time for each event channel individually.

Method

To configure data acquisition

1 In the MATLAB Command Window, enter `rtiethxcp`.

The Library: rtiethxcplib opens.

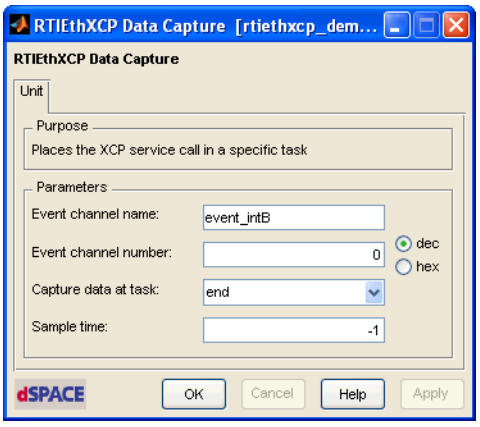


2 From the library, drag the RTIEthXCP Data Capture block into the subsystem of your model it is to be executed in.

3 Double-click the RTIEthXCP Data Capture block.

This opens the corresponding block dialog.

4 In the RTIEthXCP Data Capture block dialog, specify the event channel number and event channel name (as displayed in the A2L file), whether data is to be captured at the beginning or end of the execution of the subsystem, and the sample time.



Result

You have configured data acquisition.

Related topics

Basics

Basics on Data Acquisition with the XCP on Ethernet Real-Time Service..... 15

HowTos

How to Implement the XCP on Ethernet Real-Time Service..... 17

References

RTIEthXCP Data Capture..... 25

RTI XCP on Ethernet Blockset

Where to go from here

Information in this section

RTIEthXCP Setup Block

Purpose

To define the global settings that apply to the XCP Service access, i.e., the port number used to access the XCP on Ethernet Real-Time Service.

Where to go from here

Information in this section

Block Description (RTIEthXCP Setup)

Illustration

RTIEthXCP Setup

Access

This block is located in the *Library: rtiethxcplib*, which opens after you enter `rtiethxcp` in the MATLAB Command Window.

Purpose

To define the global settings that apply to the XCP Service access, i.e., the port number used to access the XCP on Ethernet Real-Time Service.

Description

The XCP on Ethernet Real-Time Service must be implemented in the real-time application to make the variables of the application accessible to your MC system. To ensure communication between the MC system, the dSPACE real-time hardware and the transmission of data via Ethernet, you must specify some XCP on Ethernet-specific settings.

Note

- Only one instance of the XCP on Ethernet Real-Time Service can run in an application process at a time.
- Multicore (MC) and multiprocessor applications consist of several application processes. Each application process requires a separate XCP on Ethernet Real-Time Service instance for being accessed via XCP.
- In a multicore (MC) environment, each XCP on Ethernet Real-Time Service instance - one instance per application process - requires a separate Ethernet port number.

Dialog pages

The following dialog page is available:

- Dialog Settings (RTIEthXCP Setup Block) on page 23

Related topics

References

Dialog Settings (RTIEthXCP Setup Block)

Purpose

To implement the XCP on Ethernet Real-Time Service in a real-time application.

Dialog settings

DAQ lists type Lets you specify the type of data acquisition lists. Two types are supported: DAQ lists with static size and DAQ lists with dynamic size.

A2L Ethernet address Lets you specify the Ethernet address of the host interface of the DS1007, MicroAutoBox II, or MicroLabBox.

The Ethernet address will be written to the A2L file.

Ethernet port Lets you specify the Ethernet port number used for communication. Depending on the **XCP interface** option (see below), this Ethernet port is either a port of the host PC or a port of the Ethernet host interface of the DS1007, MicroAutoBox II, or MicroLabBox.

Note

- Do not use the specified port number for purposes other than accessing the XCP on Ethernet Real-Time Service.
- On MicroAutoBox II, the port number 35350 is reserved for flashing applications via XCP which is a separate service that runs independently from the XCP on Ethernet real-time service. For details on how to write an application to the flash memory via XCP, refer to Download to the Flash Memory via XCP on Ethernet on page 32.
- Ensure that the specified Ethernet port is not blocked by a firewall application running on the host PC.

XCP interface Lets you select the Ethernet Host Interface option.

XCP communication is performed via the Ethernet host interface of the DS1007, MicroAutoBox II, or MicroLabBox.

V-ECU information Displays information on a V-ECU implementation if one was imported to the model via the V-ECU Setup block from the RTI AUTOSAR Blockset.

Related topics

Basics

Basics on Implementing the XCP on Ethernet Real-Time Service 13

HowTos

How to Implement the XCP on Ethernet Real-Time Service 17

RTIEthXCP Data Capture

Purpose

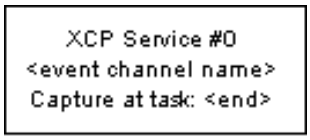
To place the service code for data acquisition via the XCP on Ethernet Real-Time Service in a specific task, and to specify the corresponding event channel number and sample time.

Where to go from here

Information in this section

Block Description (RTIEthXCP Data Capture)

Illustration



RTIEthXCP Data Capture

Access

This block is located in the *Library: rtiethxcplib*, which opens after you enter rtiethxcp in the MATLAB Command Window.

Purpose

To place the service code for data acquisition via the XCP on Ethernet Real-Time Service in a specific task, and to specify the corresponding event channel number and sample time.

Description

Data acquisition is triggered by one or more RTIEthXCP Data Capture blocks, which you can place in any subsystem of your Simulink model. Each RTIEthXCP Data Capture block defines an event channel identified by a unique event channel number. The service code needed by the XCP on Ethernet Real-Time Service is executed in the real-time application. This code samples one set of data for each sampling step at the beginning or end of the subsystem.

Dialog pages

The following dialog page is available:

- Dialog Settings (RTIEthXCP Data Capture) on page 26

Related topics

HowTos

Dialog Settings (RTIEthXCP Data Capture)

Purpose

To place the service code for data acquisition via the XCP on Ethernet Real-Time Service in a specific task, and to specify the corresponding event channel name and number, and the sample time.

Dialog settings

Event Channel Name Lets you enter a user-defined name for the event channel defined by this RTIEthXCP Data Capture block. The name is displayed in the block mask.

Event Channel Number Lets you enter a unique number for the event channel defined by this RTIEthXCP Data Capture block. You must use consecutive event channel numbers starting at 0, in either a decimal or hexadecimal format. The number is displayed in the block mask.

Note

When you import a V-ECU implementation to the model via the V-ECU Setup block from the RTI AUTOSAR Blockset, one event channel is automatically assigned to each task of the V-ECU implementation. Do not use any of the numbers of these event channels for the Data Capture blocks in your model.
The event channel numbers assigned to V-ECU tasks are displayed on the Unit page of the RTIEthXCP Setup block. Refer to Dialog Settings (RTIEthXCP Setup Block) on page 23.

Capture data at task Lets you specify whether data is to be captured before or after the subsystem is executed. It is recommended to capture data at the end of the execution of the subsystem.

Sample time Lets you enter the sample time at which the service code for the XCP on Ethernet Real-Time Service is to be executed. Choose "–1" to inherit the sample time – this is the default setting – or any multiple of the "Fixed step size" chosen for the model.

Related topics

HowTos

Appendix

Where to go from here

Information in this section

Handling an Application for the Flash Memory

Introduction

To load a real-time application automatically after power-up, it must be downloaded to the flash memory of the hardware.

Where to go from here

Information in this section

Information in other sections

Handling Applications via Command Line (ControlDesk Platform Management 🕮)
You can download and start applications on dSPACE platforms via the `cmdloader` command line tool. The tool runs in the command shell of the operating system, so you can use it in batch files or makefiles.

Down1401.exe (MicroAutoBox II RTLib Reference 🕮)
To compile, link, and download applications.

How to Clear an Application from the Flash Memory (DS100x, DS110x, MicroAutoBox II, MicroLabBox – Software Getting Started 🕮)
If an application is loaded to the flash memory, the dSPACE board starts the application automatically after reboot. If you want to avoid this, you have to clear the flash memory.

How to Download an Application to the Flash Memory and Start the Real-Time Processor (DS100x, DS110x, MicroAutoBox II, MicroLabBox – Software Getting Started 🕮)
Some dSPACE boards have a flash memory. This allows them to be used as a stand-alone system without a connection to the host PC.

How to Load an Application to the Flash Memory of dSPACE Real-Time Hardware (ControlDesk Platform Management 🕮)
Various dSPACE real-time hardware contains flash memory. Loading an application to the flash memory allows you to use dSPACE real-time hardware as a stand-alone system without a connection to the host PC.

How to Clear an Application from the Flash Memory of dSPACE Real-Time Hardware (ControlDesk Platform Management 🕮)
You can clear an application that is currently in the flash memory. This prevents the dSPACE real-time hardware from booting a flash application.

Basics of the Flash Memory

Introduction

The supported dSPACE boards are equipped with a flash memory for real-time applications. A flash memory is used to load a real-time application automatically after power-up.

Power-up

On power-up, the dSPACE board always starts executing the bootstrap loader contained in the flash memory. The loader checks for an application program currently stored in the flash memory. If it finds one, the application is started. If it does not detect an application in the flash memory, the loader enters the idle state and waits for commands from the connected host PC.

Note

- After power-up, the bootstrap loader never executes an application that was previously loaded to global memory.
- If power is turned off, an application that was previously loaded to global memory is lost.
- To prevent the dSPACE board executing an application stored in the flash memory after power-up, the flash memory must be cleared (see How to Clear an Application from the Flash Memory (DS100x, DS110x, MicroAutoBox II, MicroLabBox – Software Getting Started 🕮)).

Downloading an application to the flash memory

You can download an application to the flash memory via

- Down tool (only if you work with MicroAutoBox II, see Down1401.exe (MicroAutoBox II RTLib Reference 🕮)).
- DOS window (see Handling Applications via Command Line (ControlDesk Platform Management 🕮)).
- ControlDesk (see How to Load an Application to the Flash Memory of dSPACE Real-Time Hardware (ControlDesk Platform Management 🕮)).
- XCP on Ethernet (only if you work with MicroAutoBox II, see Download to the Flash Memory via XCP on Ethernet on page 32).

Clearing an application from the flash memory

You can clear an application from the flash memory via

- ControlDesk (see How to Clear an Application from the Flash Memory (DS100x, DS110x, MicroAutoBox II, MicroLabBox – Software Getting Started 🕮)).
- XCP on Ethernet (only if you work with MicroAutoBox II, see Download to the Flash Memory via XCP on Ethernet on page 32).

Related topics

Basics

Download to the Flash Memory via XCP on Ethernet.....32
Handling Applications via Command Line (ControlDesk Platform Management 🕮)

HowTos

How to Clear an Application from the Flash Memory (DS100x, DS110x, MicroAutoBox II, MicroLabBox – Software Getting Started 🕮)
How to Load an Application to the Flash Memory of dSPACE Real-Time Hardware (ControlDesk Platform Management 🕮)

Download to the Flash Memory via XCP on Ethernet

Introduction

MicroAutoBox II provides an XCP on Ethernet service which can be used to write real-time applications to the flash memory of MicroAutoBox II. A flash application is started automatically after the MicroAutoBox II is switched on. An application can be written to the flash memory with any host tool that implements the required XCP on Ethernet features/commands. Any dSPACE application compiled for MicroAutoBox II can be written to the flash memory.

Converting an application into XCP load file format

Before an application can be flashed via XCP on Ethernet, it must be converted into a suitable load file format. dSPACE provides the `XcpGen1401` command line tool in the `\exe` subfolder of the dSPACE RCP & HIL installation folder for this. Any application that is built with the RTI1401 Blockset or handcoded can be converted.

Using the XcpGen1401 command line tool Open a Command Prompt window and enter:

```
xcp1401 <inputFile> [outputOptions] [otherOptions]
```

The table below shows the available command parameters:

Parameter	Description
`<inputFile>`	The compiled real-time application. An application built for MicroAutoBox II usually has the file extension `.ppc`. This parameter is mandatory.

Parameter	Description
outputOptions[1)]	
`-srec <filename>`	To generate an output file in Motorola S-record file format.
`-ihex <filename>`	To generate an output file in Intel I32 hex format.
`-bin <filename>`	To generate an output file in plain binary format (debug only).
otherOptions	
`-v`	Generates verbose information. (Use -vv, -vvv, etc. to further increase verbosity.)
`-q`	Enables quiet mode (regardless of verbose level).
`-h`	Displays help text.

[1)] You can specify more than one output option. If no output options are specified, the tool automatically generates a Motorola S-record file (`<inputFile>.srec`) and an Intel file (`<inputFile>.hex`).

Example In the simplest case, an application can be converted as follows:

```
XcpGen1401 Application.ppc
```

The application named `Application.ppc` is converted to the required load file format, and the result is written in Intel file format (HEX) and Motorola S-record file format (S19). The output file names are automatically set to `Application.ppc.hex` and `Application.ppc.srec`.

Connecting to MicroAutoBox II

The XCP on Ethernet flash service on MicroAutoBox II is available only via TCP/IP, using port number 35350. It is not possible to use UDP/IP for flashing, and the default port number 35350 cannot be changed.

Unlike most ECUs, no special boot loader or flash kernel is required.

Flash programming process

The requirements for flash programming via XCP on Ethernet and the basic procedure are described below.

Memory segments MicroAutoBox II has four memory segments that are accessible via XCP on Ethernet:

Segment Number	Start Address	End Address (+1)	Length	Alignment	Type	Default
1	0x7000.0000	0x7000.0010	0x0010 (16 Byte)	0x80	DATA	0xFF
2	0x7100.0000	0x7100.0200	0x0200 (512 Byte)	0x80	DATA	0xFF
3	0x8000.0000	0x80C0.0000	0x00C0.000 (12 MB)	0x80	CODE	0xFF
4	0x8100.0000	0x8100.1000	0x1000 (4096 Byte)	0x80	DATA	0xFF

A converted real-time application allocates memory in each of these segments.

To clear and program the application, you have to perform the following steps in the given order:

1. Clear segment #1
2. Write segment #1
3. Clear segment #2
4. Write segment #2

5. Clear segment #3
6. Write segment #3
7. Clear segment #4
8. Write segment #4

The XCP on Ethernet host client can use this segment information in hardcoded form, or (preferably) request the sequence using the XCP command `GET_SECTOR_INFO`.

Processing sequence of flash programming via XCP on Ethernet To write an application to the flash memory of MicroAutoBox II, the following steps must be executed in the given order:

Step	Action	Related XCP Command(s)
1	Connect via TCP/IP to port 35350	–
2	Establish XCP connection	`CONNECT`
3	Request communication parameters	`GET_COMM_MODE_INFO`
4	Request programming parameters	`GET_PGM_PROCESSOR_INFO`
5	Request segment information	`GET_SECTOR_INFO` (multiple)
6	Start programming sequence	`PROGRAM_START`
7	Clear segment #1	`PROGRAM_CLEAR`
8	Program/write segment #1	`PROGRAM` (multiple)
9	Clear segment #2	`PROGRAM_CLEAR`
10	Program/write segment #2	`PROGRAM` (multiple)
11	Clear segment #3	`PROGRAM_CLEAR`
12	Program/write segment #3	`PROGRAM` (multiple)
13	Clear segment #4	`PROGRAM_CLEAR`
14	Program/write segment #4	`PROGRAM` (multiple)
15	(Optional) Verify segment #1	`BUILD_CHECKSUM`
16	(Optional) Verify segment #2	`BUILD_CHECKSUM`
17	(Optional) Verify segment #3	`BUILD_CHECKSUM` (multiple)
18	(Optional) Verify segment #4	`BUILD_CHECKSUM`
19	Finish programming sequence	`PROGRAM_RESET`
20	Terminate XCP connection	`DISCONNECT`
21	Disconnect TCP/IP	–

Timeouts The XCP host tool must set its command timeouts appropriately. For instance, the time required to delete application segment #3 is not fixed but depends on the current state and content of the flash memory (up to more than 10 seconds). While clearing the flash memory, the MicroAutoBox II XCP server periodically sends the XCP event "EV_CMD_PENDING" to the client to indicate that the erase sequence is still in progress. On reception of such an event, the XCP host tool must restart its timeout counter.

For further information, refer to the section on "Restarting timeout detection" in the XCP protocol specification.

Further aspects of flash programming via XCP on Ethernet

- A real-time application currently running on MicroAutoBox II is stopped before a new application is written to the flash memory. The application is stopped when the `PROGRAM_START` command is received. The MicroAutoBox II's status LED is red to indicate that the application has stopped.
- When the flash programming procedure completes, the new application is started automatically when the `PROGRAM RESET` command is received. The MicroAutoBox II's status LED is green to indicate that the application has started.
- Before the flash memory is cleared or modified, MicroAutoBox II checks if the new application will fit into the flash memory. If it does not fit, the clearing of application segment #3 is rejected with error "ERR_ACCESS_DENIED". The flash programming procedure is aborted.
- Flash programming via XCP on Ethernet is possible for applications with a file size of up to 12 MB.
- Only applications that were written via XCP can be verified via XCP. If an application which was written to the flash memory via ControlDesk is verified via XCP, the XCP host tool indicates a checksum mismatch.
- The MicroAutoBox II XCP server has an internal timeout of 10 minutes. If no XCP command is received within this time, a currently running session is terminated, and the TCP/IP connection is closed.
- While an XCP programming sequence is in progress, XCP has exclusive access to the flash memory. Any flash memory access via ControlDesk is rejected.

  The flash memory is also locked while being accessed by ControlDesk. This means that an XCP programming sequence is rejected during this time.

Interdependencies with the XCP on Ethernet CAL/DAQ service

The XCP service for writing applications to the flash memory is always active and runs independently of the XCP service for application calibration (CAL) and measurement (DAQ). The CAL/DAQ service is available only if the corresponding RTI blockset is used in the application.

Note

The TCP port 35350 is reserved for flash access and therefore cannot be used for the CAL/DAQ service.

Using the dSPACE ECU Flash Progamming Tool

The dSPACE ECU Flash Programming Tool can be used to program the flash memory of ECUs, for example, via XCP on Ethernet or XCP on CAN. A ready-to-use project configuration file for accessing MicroAutoBox II is provided by dSPACE. For further information, contact dSPACE Support.

Related topics

Basics

Basics of the Flash Memory....31

Abbreviations

Abbreviations

Abbreviation overview

The following abbreviations are used in this document:

Abbreviation	Description
ASAP1a	Software interface for connecting a measurement and calibration system to an ECU; this part describes the interface in the ECU.
A2L	File format for variable description files of ECUs. An A2L file contains information about the interface to be used and all accessible variables.
DAQ	Data acquisition
ECU	Electronic control unit
ID	Identifier
MC	Measurement and calibration
ODT	Object descriptor table
RTI	Real-Time Interface
XCP	Universal Measurement and Calibration Protocol

Limitations

Limitations

Introduction

The following limitations and workarounds apply:

Limitations of the XCP on Ethernet Real-Time Service

The XCP on Ethernet Real-Time Service has the following limitations:

- Only one instance of the XCP on Ethernet Real-Time Service can run on a dSPACE platform at the same time. If you have a multiprocessor system or if multiple RCP systems are connected to one host system, the XCP on Ethernet Real-Time Service is limited to be used with one processor board only.
- The variable description (TRC and SDF) file generated by RTI contains only the variables that are prepared for use with the XCP on Ethernet Real-Time Service. Only these variables are accessible in an experiment of ControlDesk.

Limitations for using the Signal Generator block

Using the Signal Generator block might cause problems when you download calibration parameters to the hardware. For this reason, some limitations apply to the Signal Generator block.

Workaround If you use the Signal Generator block, select **Use external signal** for the **Time (t)** value. Then connect a Clock block to the inport of the Signal Generator block. As a limitation, you cannot change the frequency of the Signal Generator block's signal during run time.

Problems while downloading an application to the flash memory without ControlDesk

If you encounter problems while downloading an application to the flash memory without ControlDesk, you can reset the device driver of your dSPACE hardware using the `dcont -r` command.

Note

- If you use the `dcont -r` command, not only the device driver of one dSPACE hardware component, for example, MicroAutoBox II, will be reset but all the device drivers of all the dSPACE hardware components connected to your PC.
- Do not use the `dcont -r` command if you have ControlDesk installed. Using the `dcont -r` command may prevent multiprocessor systems from being recognized by ControlDesk.
  If you have ControlDesk installed, use the Refresh Interface Connections (ControlDesk Platform Management 🕮) command.

Related topics

Basics

Introduction to the XCP on Ethernet Real-Time Service.....7

References

Refresh Interface Connections (ControlDesk Platform Management 🕮)