CODESYS STORE

CANopen Example

This sample project deals with the programming interface of the CANopen master and CANopen slave communication stack. The project contains examples for SDO, NMT, diagnostics, and the dynamic configuration of communication parameters (baud rate, network ID, node ID), and many more.

Product description

This sample project contains different examples for using the programming interface of the CANopen master and CANopen slave stack. Topics that are handled include the following:

- SDO (expedited, segmented, block), object directory access
- Network Management (NMT)
- Diagnostics (CANopen state, EMCY)
- Event handler for object directory and CANopen manager state machine
- Reconfigure: Dynamic changing of baud rate, network ID, and node ID
- Access to CAN configuration by means of the device diagnosis library

More information

The project includes a CANopen master and a CANopen slave stack that communicate with each other via CANbus. Two interconnected CAN interfaces are needed for commissioning the project. It is easiest to implement this with CODESYS Control Win V3 and a PEAK PCAN-USB Pro or two PEAK PCAN USB adapters. For this to work, the runtime system component “CmpPCANBasicDrv” must be entered in the configuration file. For more information, refer to the CODESYS online help.

The project can also be operated on any controller with two CAN interfaces by means of an update device of the PLC.

The project is divided into three areas:

CANbus examples

- `ChangeBaudrate`: Dynamic change of baud rate in runtime mode by means of reconfigure
- `EnableDisableCANbus`: Dynamic activation and deactivation of a CANbus in runtime mode by means of reconfigure
- `GetCANbus`: Generic search of an engineered CANbus instance by means of the device diagnosis library

CANopen master examples

Diagnostics

- `ReceiveEMCYFromAllDevicesExample`: Reception of emergency messages by means of `CiA405.RECV_EMCY`
- `ReceiveEMCYFromOneDeviceExample`: Reception of emergency messages for a specific device by means of `CIA405.RECV_EMCY_DEV`
- `GetCANopenKernelStateExample`: Detection of the CANopen kernel state by means of `CIA405.GET_CANOPEN_KERNEL_STATE`
- `GetStateExample`: Detection of the CANopen state of a slave by means of `CIA405.GET_STATE`

Network management

- `GetCANopenManagerNodeIDExample`: Detection of the CANopenManager node ID by means of `CiA405.GET_CANOPEN_KERNEL_STATE`
- `NMT_Example`: Sending of NMT requests by means of `CIA405.NMT`

SDO (Acyclic data transfer)

- `ExpeditedReadExample`: Reading of an object <= 4 bytes via SDO (expedited transfer) by means of `CiA405.SDO_READ4`
- `ExpeditedWriteExample`: Writing of an object <= 4 bytes via SDO (expedited transfer) by means of

CIA405.SDO_WRITE4

- SegmentedAndBlockReadExample: Reading of an object of any length via SDO (segmented or block transfer) via CiA405.SDO_READ_DATA
- SegmentedAndBlockWriteExample: Writing if an object of any length via SDO (segmented or block transfer) via CiA405.SDO_WRITE_DATA

General API functions

- CANopenManagerAPI: Demonstrates the use of all methods and properties of the CANopenManager instance
- CANopenManagerEventHandler: Exemplary implementation of a CANopen event handler
- CANopenRemoteDeviceAPI: Demonstrates the use of all methods and properties of a CANopenRemoteDevice instance

CANopen slave examples

Diagnostics

- GetLocalDeviceInfoExample: Demonstrates the reading of the CANopen state as well as the node ID

Network management

- ChangeNodeIDExample: Change of the node ID in runtime mode
- NMTExample: Change of the local CANopen state

Object directory

- ReadObjectExample1: Reading of an object from the local object directory
- ReadObjectExample2: Alternative example of ReadObjectExample1
- WriteObjectExample1: Writing an object to the object directory
- WriteObjectExample2: Alternative example of WriteObjectExample1
- CountObjects: Traversing of the complete object directory by means of _3SCSS.ObjectIterator
- GetPDOInfo: Additional example of traversing the object directory
- ObjectDictionaryEventHandler: Exemplary implementation of an object directory event handler

General information

Manufacturer:

3S-Smart Software Solutions GmbH
Memminger Strasse 151
87439 Kempten
Germany

Support:

https://support.codesys.com

Item:
CANopen Example
Item number:
000065
Sales:

CODESYS Store
https://store.codesys.com

Included in delivery:
Sample project: CANopenExamples.project

System requirements and restrictions

Programming System	CODESYS Development System V3.5.10.0 or higher
Runtime System	CODESYS Control V 3.5.10.0
Supported Platforms/ Devices	All platforms of the CODESYS Runtime environment Note: Use the project “Device Reader” to find out the supported features of your device. “Device Reader” is available for free in the CODESYS Store.
Additional Requirements	To run the sample project on CODESYS Control Win V3: • Two CAN interfaces from the company PEAK-System Technik GmbH • Activation of the runtime system component “CmpPCANBasicDrv” in the configuration file As an alternative, the project can also be run on another CODESYS controller with two CAN interfaces.
Restrictions	-
Licensing	Licensing not required
Required Accessories	Two CAN interfaces from the company PEAK-System Technik GmbH (e.g. two PCAN-USB or one PCAN-USB Pro)

Note: Not all CODESYS features are available in all territories. For more information on geographic restrictions, please contact sales@codesys.com.
Note: Technical specifications are subject to change. Errors and omissions excepted. The content of the current online version of this document applies.