



Application Note

EtherCAT Slave Programming Example

How to configure and use CoE Communication

Beckhoff TwinCAT Master to Hilscher Slave

Hilscher Gesellschaft für Systemautomation mbH

www.hilscher.com

DOC100708AN04EN | Revision 4 | English | 2013-07 | Released | Public

Table of Contents

1	Introduction.....	3
1.1	About this Document.....	3
1.2	List of Revisions	3
1.3	Features of EtherCAT CoE Basic Sample Application	3
2	Set-up the EtherCAT Network.....	4
2.1	Set-up the Hardware	4
2.2	Configuring the Hilscher EtherCAT CoE Slave.....	4
2.3	Set-up a TwinCAT Project.....	7
2.4	Running Tests	11
2.4.1	General Test.....	11
2.4.2	Using the Demo Object for Testing	12
2.4.3	Process Data Tests	13
3	Appendix	16
3.1	List of Tables	16
3.2	List of Figures.....	16
3.3	Contacts	17

1 Introduction

1.1 About this Document

This manual describes how to run the basic sample Application for Hilscher EtherCAT Slave CoE stack in combination with:

- Beckhoff TwinCAT as EtherCAT Master

The used components are

- 1 PC with 1 Standard Ethernet Card and 1 PC Card cifX
- Hilscher EtherCAT Slave Stack Version 4.2.x
- Beckhoff TwinCAT Version 2.11.1552
- Microsoft Visual Studio 6

1.2 List of Revisions

Rev	Date	Name	Chapter	Revision
1	2010-07-07	EO	all	Created
2	2010-04-19	EO	all	Updated for new Stack Version V2.5.16.0
3	2012-06-26	RW	1.1	Updated for new Stack Version V4.2.x
4	2013-07-17	HH	all	Revised

Table 1: List of Revisions

1.3 Features of EtherCAT CoE Basic Sample Application

The EtherCAT CoE basic sample Application is a straight forward implementation which shows how to use CifX Driver API and how to communicate using packets.

The application configures a slave with following features:

- 4 byte PDO input
- 4 byte PDO output
- 1 SDO simple variable object – non indexed access (0x6000)
- 1 SDO record object – 16 Elements (0x6010)
- 1 SDO array object – 16 Elements (0x6020)

The application will simply mirror the received data. This means, that the application will copy the output data to the input data and send it back to the master.

Furthermore, the application registers for write indications of the demo object 0x6000. Each time the object is written from master, this will be indicated within the console output.

2 Set-up the EtherCAT Network

2.1 Set-up the Hardware

Use the following environment:

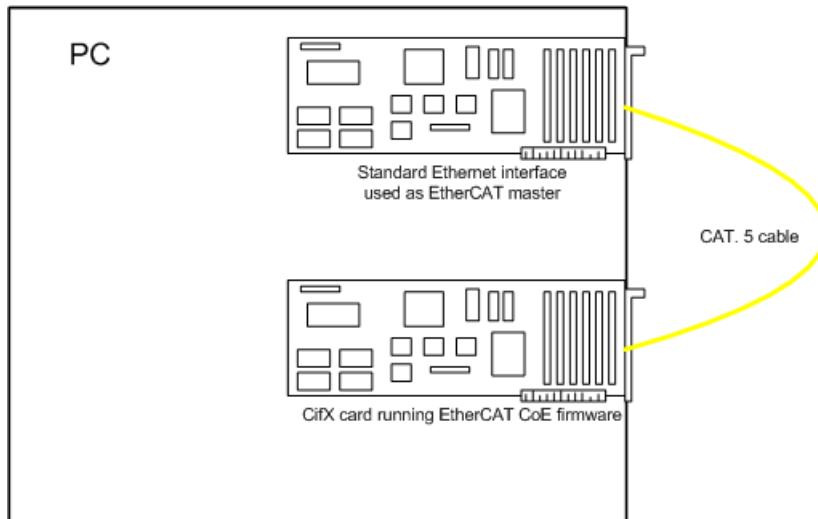


Figure 1: Hardware Setup

2.2 Configuring the Hilscher EtherCAT CoE Slave

- Load EtherCAT slave firmware file via cifX Driver Setup Utility on your PC card cifX.

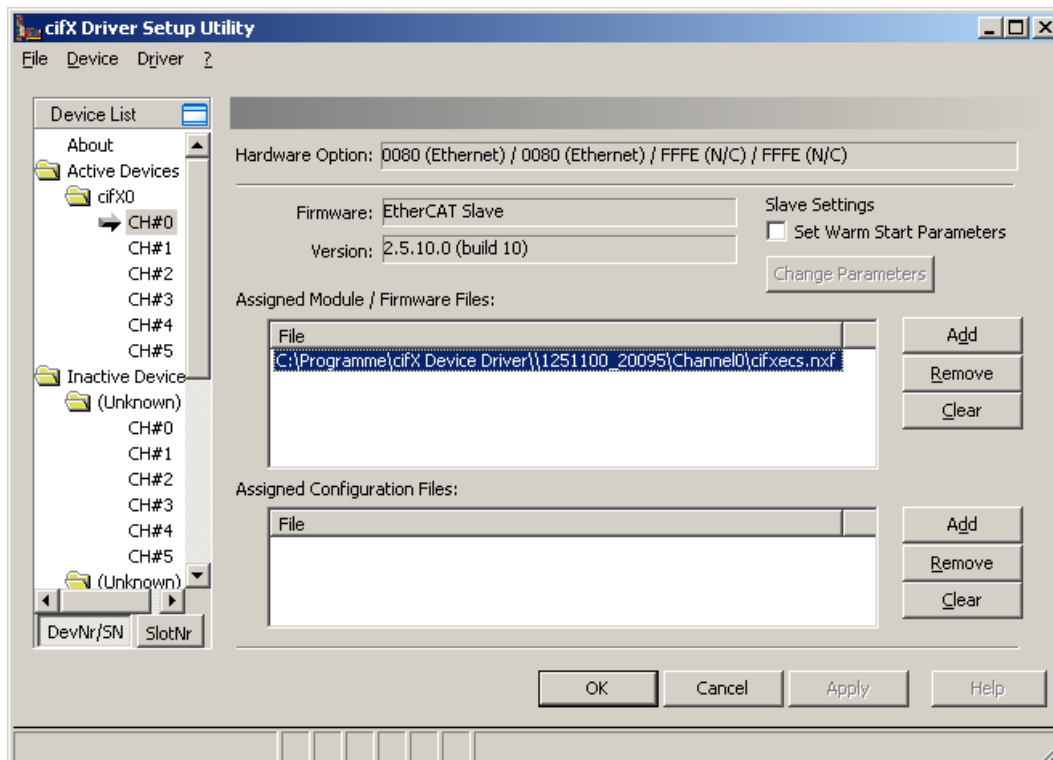


Figure 2: CifX Setup to load the EtherCAT Slave Firmware

- Open EtherCAT basic sample Application (CifX_ApDemo.dsw) using Microsoft Visual Studio 6 or higher.
- Open file Application.c and adjust the board name at the beginning of the App_Main() function. Set the board instance of the PC card cifX to which you have loaded the firmware.

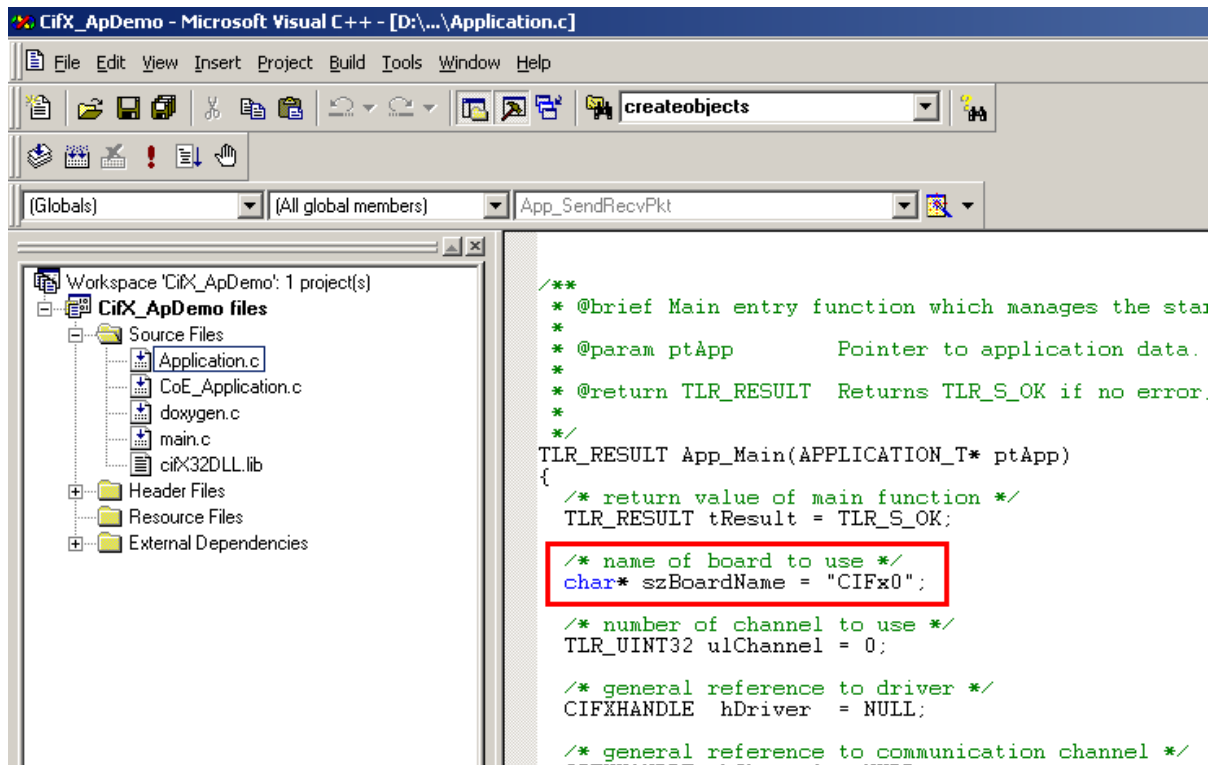


Figure 3: Adjust the cifX Board Instance

- Build and start the application.

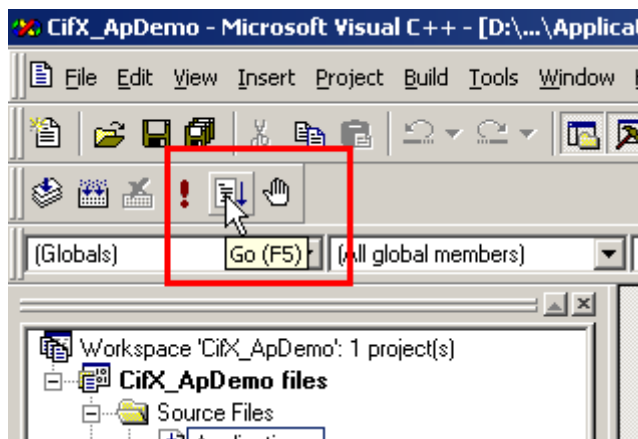


Figure 4: Starting the Example

- You can monitor the system startup by checking the console output.

```
C:\D:\Workspace_EtherCAT\EtherCAT-Slave_branch2.5.x\Examples_CoE\Y56\Debug\...
Allocating resources...
Successful.

*****
*
*   Basic sample application for EtherCAT CoE
*
*   Copyright (c) Hilscher GmbH. All Rights Reserved.
*
*****

Opening driver...
Opening channel 0 on board CIfx0...
Processing system restart...
Sending configuration request...
Processing channel init...
```

Figure 5: Startup

- After the startup, the application runs into an endless loop. Within this loop, packets (acyclic data) as well as process data (cyclic) are handled. Every time a packet is handled, the application prints a "-". When the handling of process data fails, e.g. if there is no master communication, a "F" will be printed. When process data exchange was successful, a "X" will be printed.

```
C:\Workspace_EtherCAT\EtherCAT-Slave_branch2.5.x\Examples\CifX_CoE\V56\Debug\... - _ X
```

```
Allocating resources...  
Successful.  
  
*****  
*                                                                 *  
*   Basic sample application for EtherCAT CoE                    *  
*                                                                 *  
*   Copyright (c) Hilscher GmbH. All Rights Reserved.          *  
*                                                                 *  
*****  
  
Opening driver...  
Opening channel 0 on board CIfx0...  
Processing system restart...  
Sending configuration request...  
Processing channel init...  
Initializing application...  
Setting bus state on...  
Entering endless loop...  
  
>> Press any key to leave. <<  
  
F-F-F-FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF  
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

Figure 6: Entering Cyclic IO Exchange

- As soon as the master writes the Demo Object (0x6000), the new value will also be shown in the console output (see *Using the Demo Object for Testing* on page 12).

2.3 Set-up a TwinCAT Project

- Install TwinCAT on your PC (if not already done).
- Add the device description file which comes with the example to the TwinCAT system. Therefore copy file “Hilscher CoE Dynamic Mapping Example.xml” to the following folder:
<TwinCATInstallDir> \ Io \ EtherCAT
- Start TwinCAT System in Config-Mode.

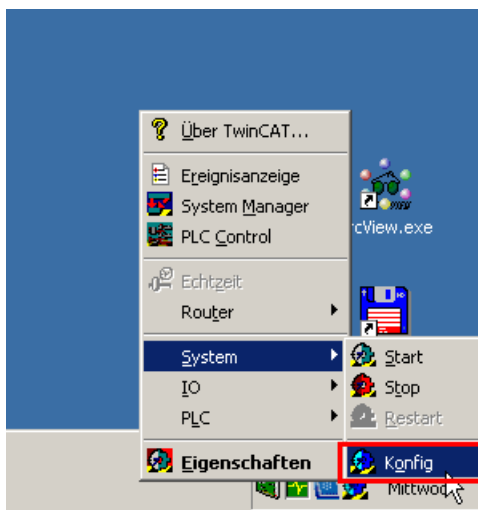


Figure 7: TwinCAT System Start

- Open TwinCAT System Manager and create a new project.

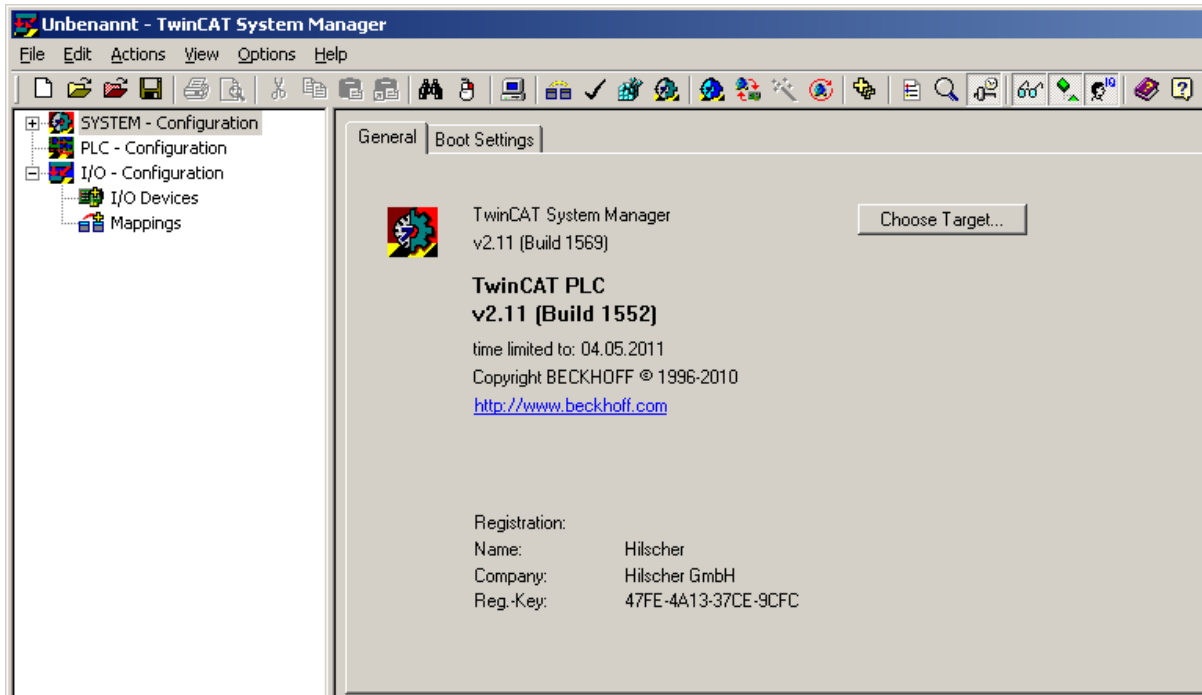


Figure 8: TwinCAT System Manager

- Append a new I/O Device

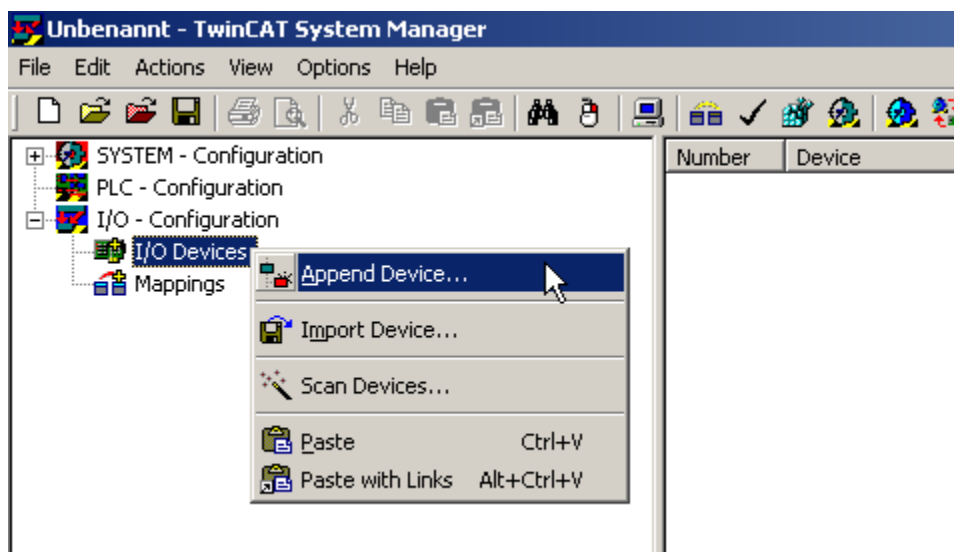


Figure 9: Append I/O Device

- Select EtherCAT as device and the LAN interface which you are using.

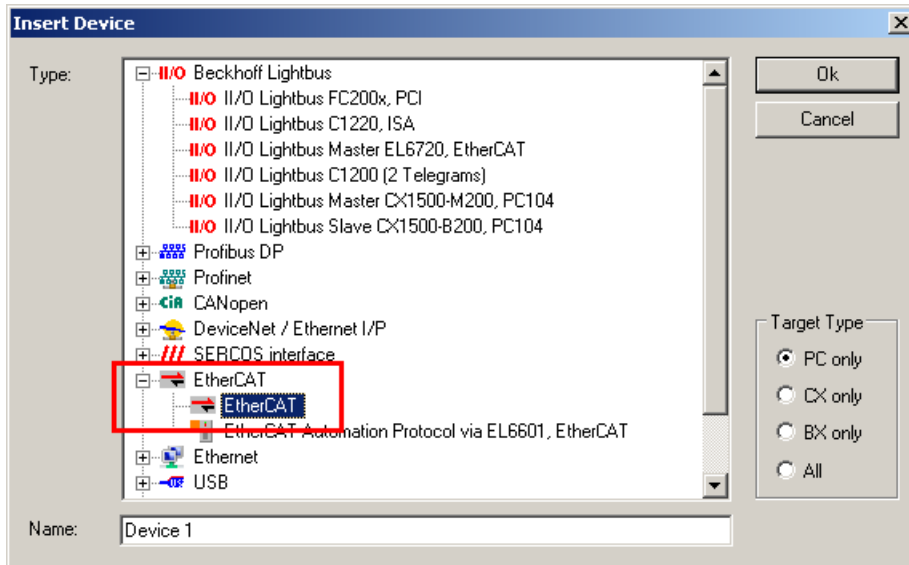


Figure 10: Device Selection

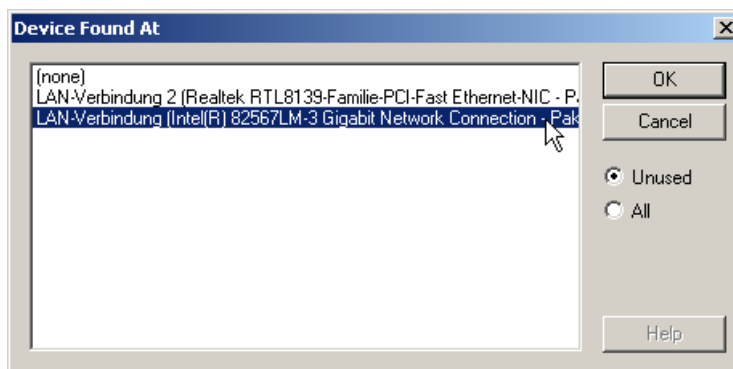


Figure 11: Interface Selection

- Open the context menu of the new device and select **Scan Boxes** in order to find the new EtherCAT slave. Note: The basic sample Application must have been started at this time.

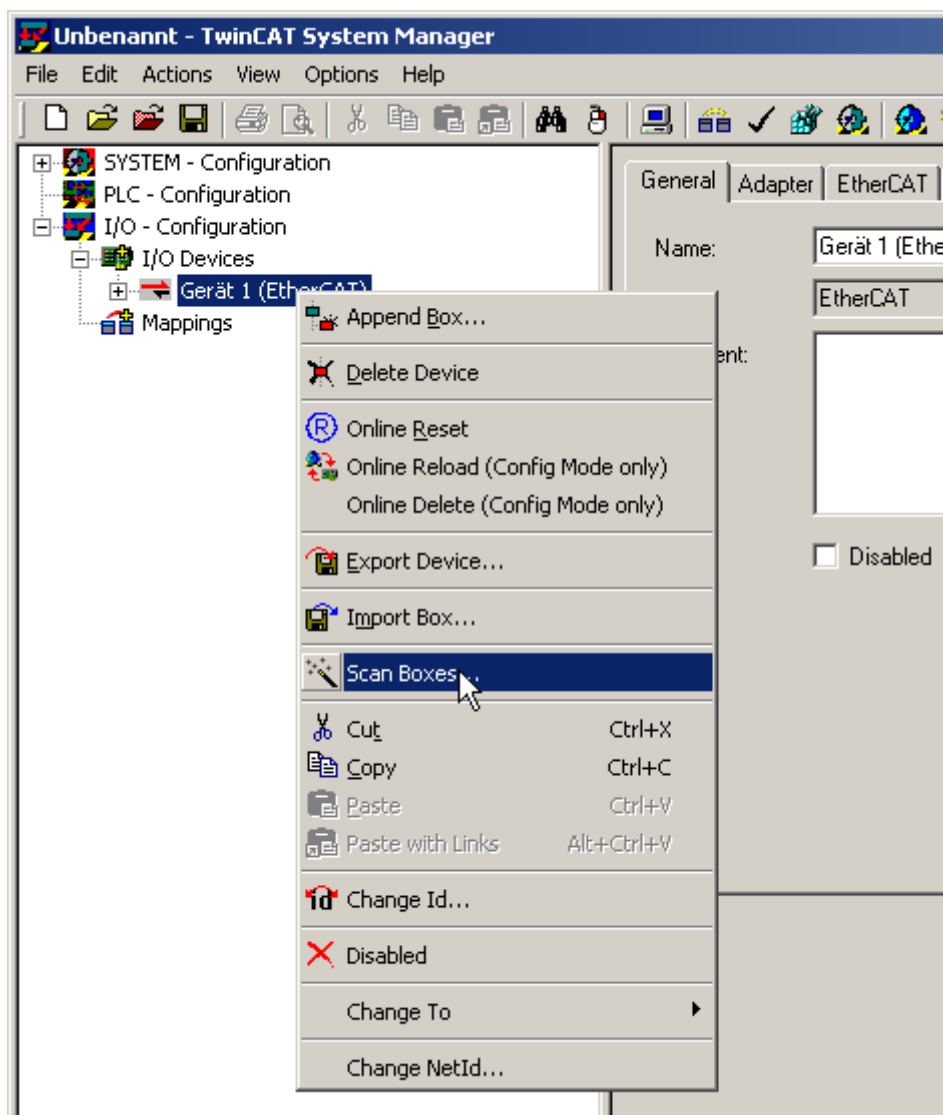


Figure 12: Scan Boxes

- TwinCAT will find the new slave device and add it to the project.

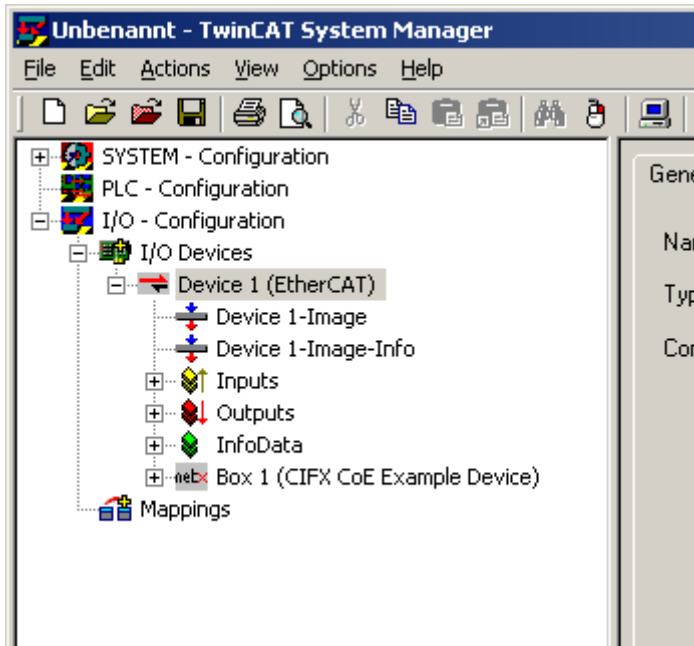


Figure 13: Project with EtherCAT Slave Device

2.4 Running Tests

2.4.1 General Test

- Start the EtherCAT basic sample Application as described in section *Features of EtherCAT CoE Basic Sample Application* on page 3.
- Switch TwinCAT to **Free Run** state

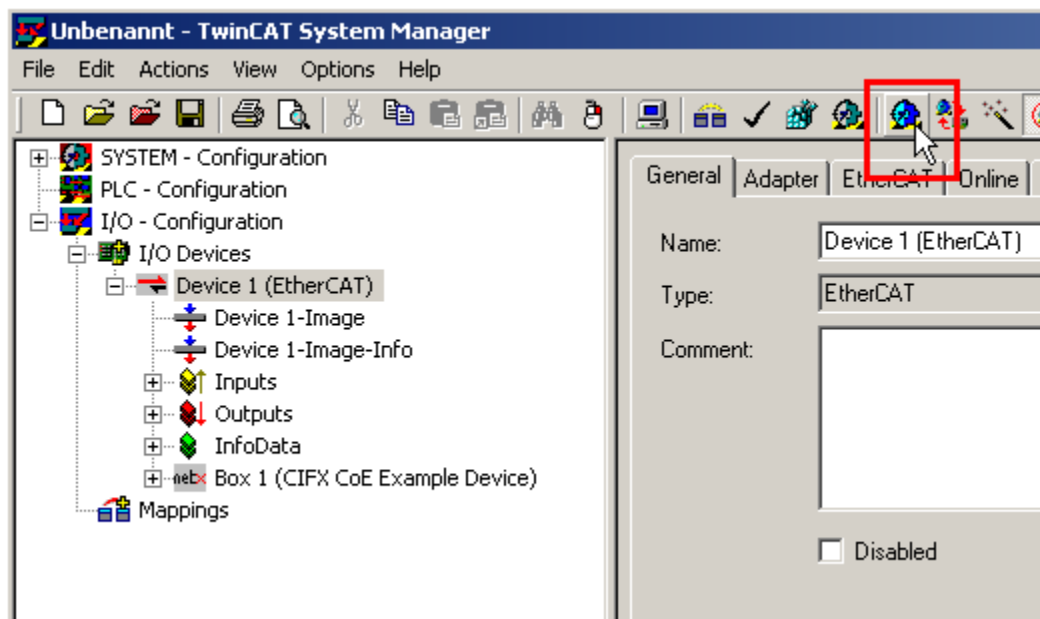


Figure 14: Activate "Free Run"

- The EtherCAT slave device will now switch to OP-State.

- You can verify this via opening tab page **Online** of the EtherCAT slave device.

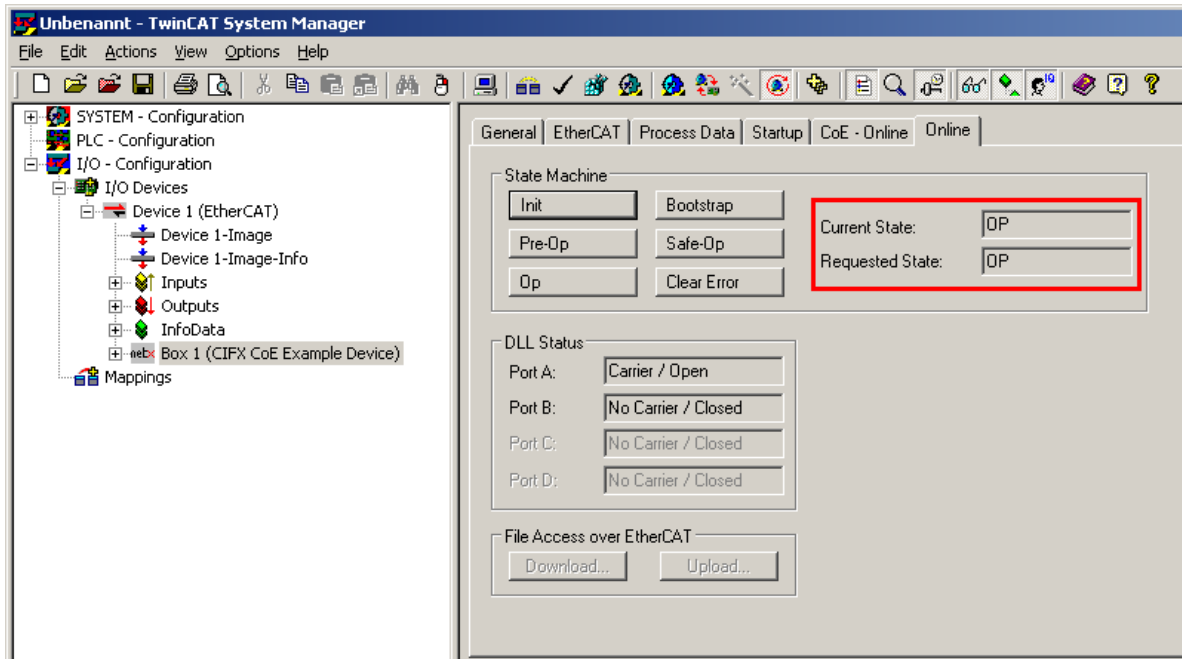


Figure 15: Verify the State

2.4.2 Using the Demo Object for Testing

- Open tab page **CoE - Online** of the EtherCAT slave device.

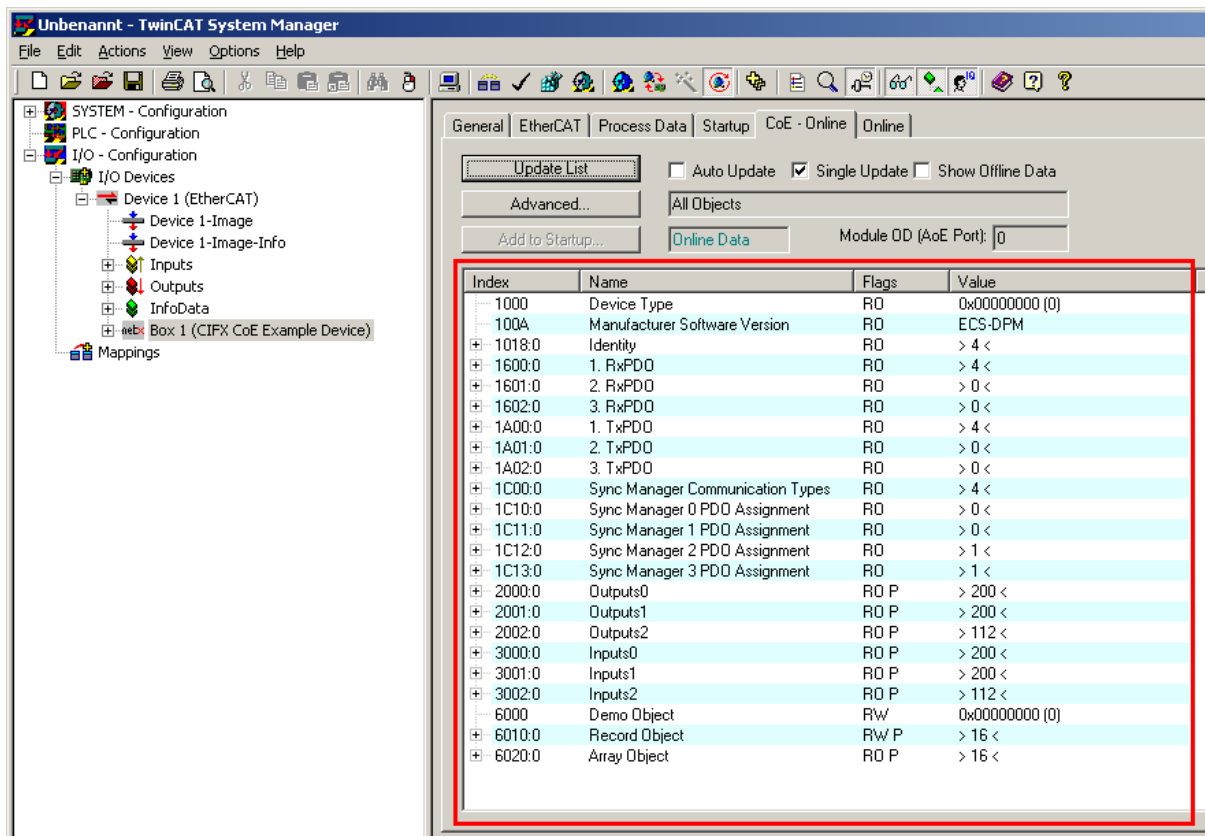


Figure 16: Tab CoE-Online

- At the bottom of the list, you will find the demo object 0x6000.
- Double click on the line of the demo object.
- A new window appears. This window allows writing a new value into the object.

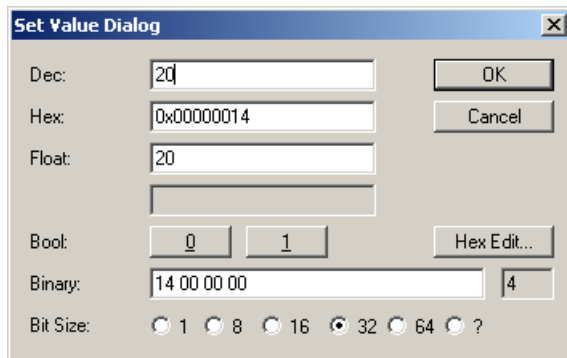


Figure 17: Set Value Dialog (SDO 0x6000)

- Write a new value, click **OK** and monitor the console output.
- The write access itself as well as the new value of the object will be indicated within the console output.



Figure 18: Write Indication Within Console Output

2.4.3 Process Data Tests

As mentioned above (see section Features of EtherCAT CoE Basic Sample Application on page 3), the application will mirror the output data and send it back to the master as input data. This behaviour can be verified using TwinCAT.

- Open tab page **Online** of the first output byte.

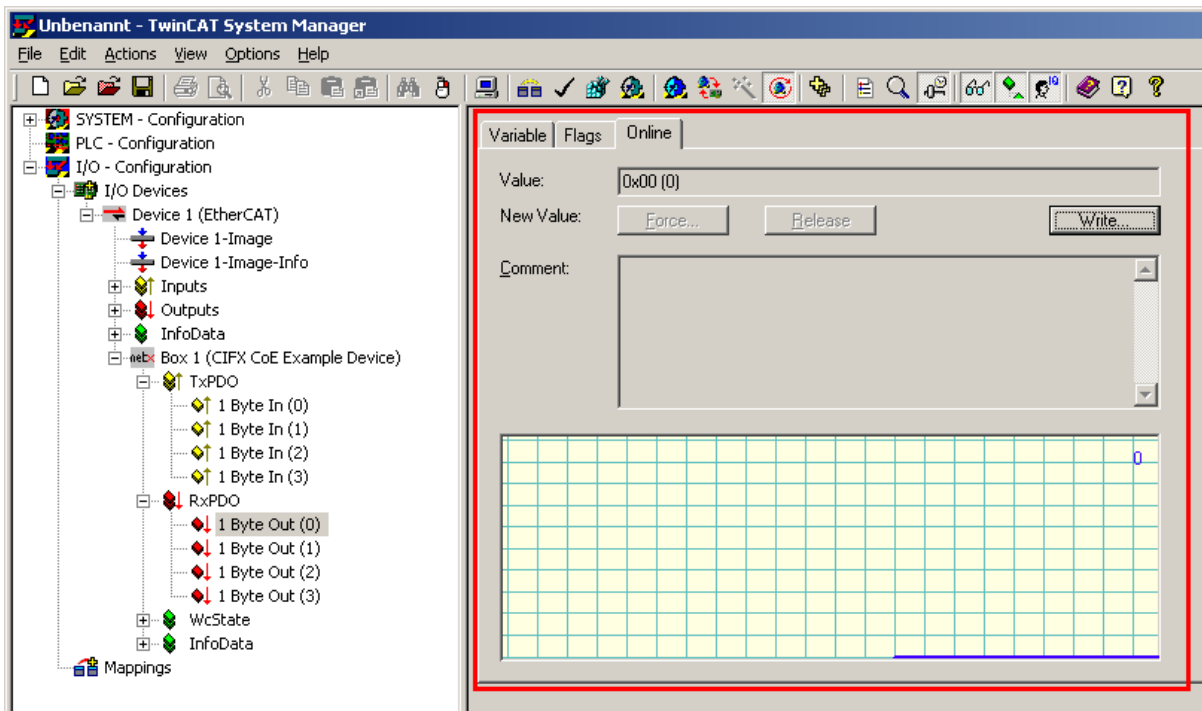


Figure 19: Tab Online (Output Byte)

- The output will initially set to zero (0x00).
- Click the **Write** button and write a new pattern (e.g. 0xFF).

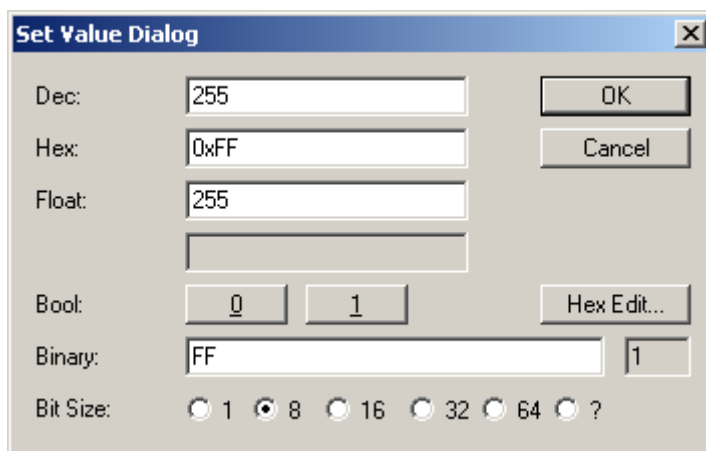


Figure 20: Set Value Dialog (Output Byte)

- The application will mirror this pattern to the first of the input bytes.
- Verify this by opening tab page **Online** of the first input byte.

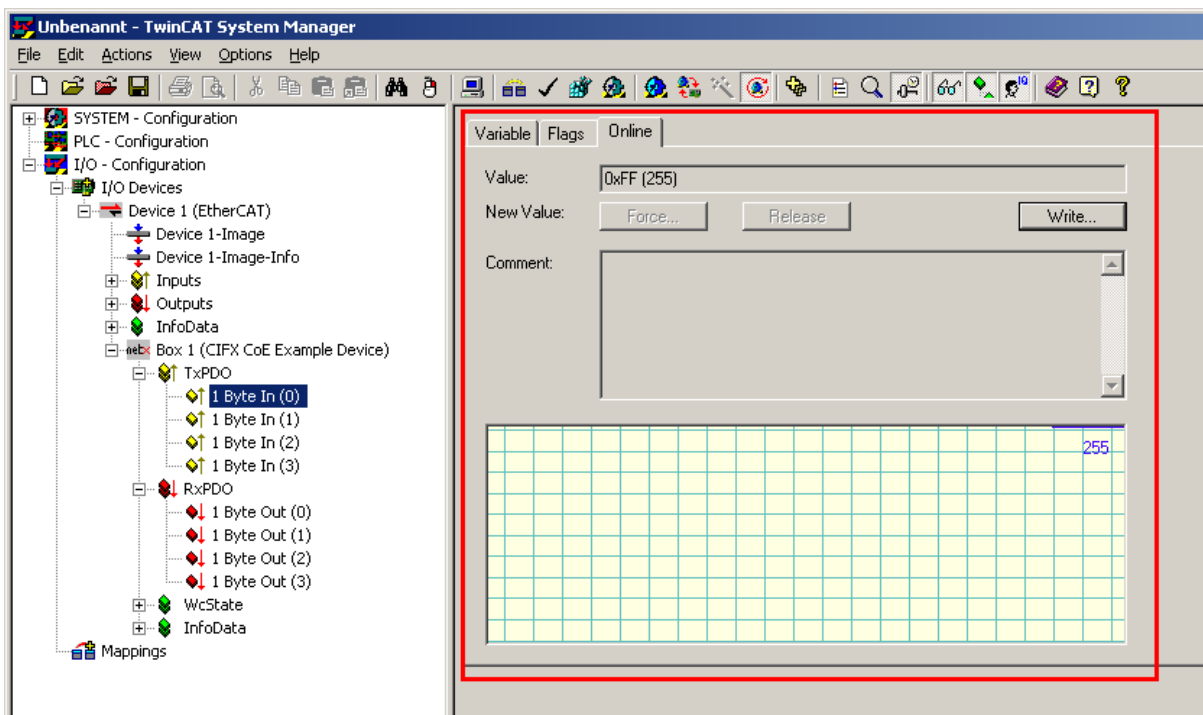


Figure 21: Tab Online (Input Byte)

- The value must equal the value you have written into the corresponding output byte (0xFF).

3 Appendix

3.1 List of Tables

Table 1: List of Revisions	3
----------------------------------	---

3.2 List of Figures

Figure 1: Hardware Setup	4
Figure 2: CifX Setup to load the EtherCAT Slave Firmware.....	4
Figure 3: Adjust the cifX Board Instance	5
Figure 4: Starting the Example.....	5
Figure 5: Startup	6
Figure 6: Entering Cyclic IO Exchange	6
Figure 7: TwinCAT System Start.....	7
Figure 8: TwinCAT System Manager	8
Figure 9: Append I/O Device	8
Figure 10: Device Selection	9
Figure 11: Interface Selection	9
Figure 12: Scan Boxes	10
Figure 13: Project with EtherCAT Slave Device	11
Figure 14: Activate "Free Run"	11
Figure 15: Verify the State.....	12
Figure 16: Tab CoE-Online	12
Figure 17: Set Value Dialog (SDO 0x6000).....	13
Figure 18: Write Indication Within Console Output.....	13
Figure 19: Tab Online (Output Byte)	14
Figure 20: Set Value Dialog (Output Byte)	14
Figure 21: Tab Online (Input Byte)	15

3.3 Contacts

Headquarters

Germany

Hilscher Gesellschaft für
Systemautomation mbH
Rheinstrasse 15
65795 Hattersheim
Phone: +49 (0) 6190 9907-0
Fax: +49 (0) 6190 9907-50
E-Mail: info@hilscher.com

Support

Phone: +49 (0) 6190 9907-99
E-Mail: de.support@hilscher.com

Subsidiaries

China

Hilscher Systemautomation (Shanghai) Co. Ltd.
200010 Shanghai
Phone: +86 (0) 21-6355-5161
E-Mail: info@hilscher.cn

Support

Phone: +86 (0) 21-6355-5161
E-Mail: cn.support@hilscher.com

France

Hilscher France S.a.r.l.
69500 Bron
Phone: +33 (0) 4 72 37 98 40
E-Mail: info@hilscher.fr

Support

Phone: +33 (0) 4 72 37 98 40
E-Mail: fr.support@hilscher.com

India

Hilscher India Pvt. Ltd.
New Delhi - 110 065
Phone: +91 11 26915430
E-Mail: info@hilscher.in

Italy

Hilscher Italia S.r.l.
20090 Vimodrone (MI)
Phone: +39 02 25007068
E-Mail: info@hilscher.it

Support

Phone: +39 02 25007068
E-Mail: it.support@hilscher.com

Japan

Hilscher Japan KK
Tokyo, 160-0022
Phone: +81 (0) 3-5362-0521
E-Mail: info@hilscher.jp

Support

Phone: +81 (0) 3-5362-0521
E-Mail: jp.support@hilscher.com

Korea

Hilscher Korea Inc.
Seongnam, Gyeonggi, 463-400
Phone: +82 (0) 31-789-3715
E-Mail: info@hilscher.kr

Switzerland

Hilscher Swiss GmbH
4500 Solothurn
Phone: +41 (0) 32 623 6633
E-Mail: info@hilscher.ch

Support

Phone: +49 (0) 6190 9907-99
E-Mail: ch.support@hilscher.com

USA

Hilscher North America, Inc.
Lisle, IL 60532
Phone: +1 630-505-5301
E-Mail: info@hilscher.us

Support

Phone: +1 630-505-5301
E-Mail: us.support@hilscher.com