

Manual

TC3 EtherCAT External Sync

TwinCAT 3

Version: 1.0
Date: 2018-11-30
Order No.: TF6225

BECKHOFF

Table of contents

1 Foreword	5
1.1 Notes on the documentation.....	5
1.2 Safety instructions	6
2 Overview	7
3 Installation	8
3.1 System requirements.....	8
3.2 Licensing	8
4 Technical introduction	14
5 PLC API	15
5.1 Function blocks.....	15
5.1.1 FB_EcExtSyncExtTimes.....	15
5.1.2 FB_EcExtSyncIsSynchronized	16
5.2 Data types	17
5.2.1 ST_ExtTimingTimes	17
6 Examples	18
6.1 EtherCAT External Sync.....	18
6.1.1 Configuration	18
6.1.2 Notes	20
6.1.3 Results	21
6.2 DCF77	22
7 Appendix	24
7.1 ADS Return Codes	24
7.2 Support and Service	26

1 Foreword

1.1 Notes on the documentation

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with the applicable national standards.

It is essential that the documentation and the following notes and explanations are followed when installing and commissioning the components.

It is the duty of the technical personnel to use the documentation published at the respective time of each installation and commissioning.

The responsible staff must ensure that the application or use of the products described satisfy all the requirements for safety, including all the relevant laws, regulations, guidelines and standards.

Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development.

We reserve the right to revise and change the documentation at any time and without prior announcement. No claims for the modification of products that have already been supplied may be made on the basis of the data, diagrams and descriptions in this documentation.

Trademarks

Beckhoff®, TwinCAT®, EtherCAT®, Safety over EtherCAT®, TwinSAFE®, XFC® and XTS® are registered trademarks of and licensed by Beckhoff Automation GmbH.

Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

Patent Pending

The EtherCAT Technology is covered, including but not limited to the following patent applications and patents:

EP1590927, EP1789857, DE102004044764, DE102007017835

with corresponding applications or registrations in various other countries.

The TwinCAT Technology is covered, including but not limited to the following patent applications and patents:

EP0851348, US6167425 with corresponding applications or registrations in various other countries.

EtherCAT 

EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany

Copyright

© Beckhoff Automation GmbH & Co. KG, Germany.

The reproduction, distribution and utilization of this document as well as the communication of its contents to others without express authorization are prohibited.

Offenders will be held liable for the payment of damages. All rights reserved in the event of the grant of a patent, utility model or design.

1.2 Safety instructions

Safety regulations

Please note the following safety instructions and explanations!
Product-specific safety instructions can be found on following pages or in the areas mounting, wiring, commissioning etc.

Exclusion of liability

All the components are supplied in particular hardware and software configurations appropriate for the application. Modifications to hardware or software configurations other than those described in the documentation are not permitted, and nullify the liability of Beckhoff Automation GmbH & Co. KG.

Personnel qualification

This description is only intended for trained specialists in control, automation and drive engineering who are familiar with the applicable national standards.

Description of symbols

In this documentation the following symbols are used with an accompanying safety instruction or note. The safety instructions must be read carefully and followed without fail!

DANGER

Serious risk of injury!

Failure to follow the safety instructions associated with this symbol directly endangers the life and health of persons.

WARNING

Risk of injury!

Failure to follow the safety instructions associated with this symbol endangers the life and health of persons.

CAUTION

Personal injuries!

Failure to follow the safety instructions associated with this symbol can lead to injuries to persons.

NOTE

Damage to the environment or devices

Failure to follow the instructions associated with this symbol can lead to damage to the environment or equipment.



Tip or pointer

This symbol indicates information that contributes to better understanding.

2 Overview

As a real-time software environment, TwinCAT offers the option of synchronization with an externally specified time, based on cycle/frequency or absolute time. Time access usually takes place via the EtherCAT fieldbus and the integrated distributed clocks functionality. The external time can be transferred to the TwinCAT system in different ways:

Method	Required devices
IEEE 1588, PTP	EL6688 EtherCAT Terminal
EtherCAT <-> EtherCAT	EL6692, EL6695 EtherCAT Terminal
Any clock pulse as electrical signal	EL1252 EtherCAT Terminal (as described in this document)
Other methods: See Beckhoff Information System	

The TwinCAT 3 EtherCAT External Sync function extends the TwinCAT EtherCAT master with the option of synchronizing the Beckhoff real-time with an external digital electrical clock signal and provides a library with various function blocks for this purpose. The signal must be connected to an EL1252 EtherCAT Terminal, which means that the EtherCAT fieldbus has to be used.

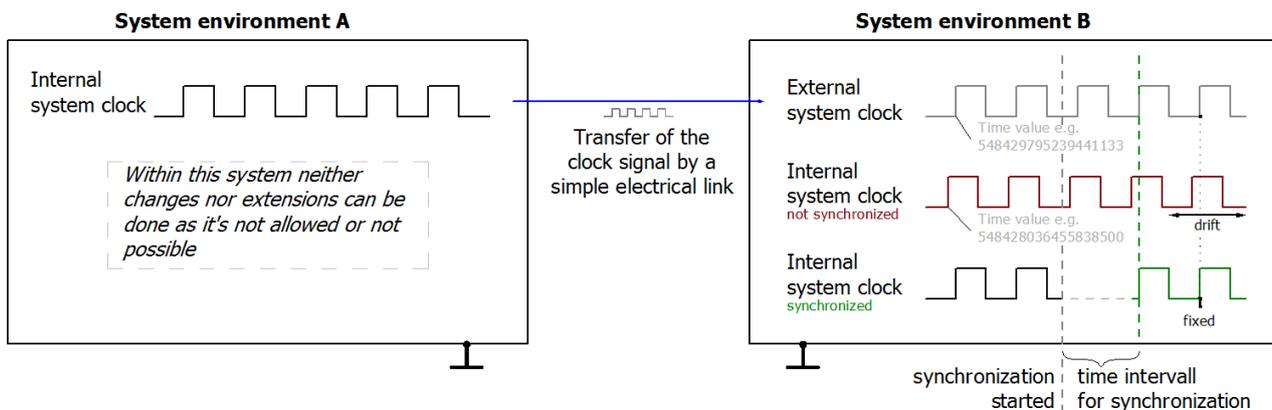
● EtherCAT System Documentation

i For further information on external synchronization see the Beckhoff Information System, section Fieldbus Components > EtherCAT Terminals > [EtherCAT System Documentation](#)

Application examples

- An existing system is to be expanded with an additional Beckhoff TwinCAT controller, which should be synchronized with the main controller. At the main controller, the clock signal that is picked up by the TwinCAT controller via the EL1252 can be generated by a toggling 24 V digital output, for example. (See [Examples > EtherCAT External Sync \[▶ 18\]](#))
- A TwinCAT system should generally run synchronously with an external cycle, e.g. a second pulse (PPS, pulses per second) or local time
- A TwinCAT system is to be supplied with the absolute time, in addition to the clock pulse. If an absolute time is also modulated onto an external PPS signal, e.g. via DCF77 coding, the TF6225 can be used in conjunction with a PLC function block to decode the DCF77 signal. The function blocks for coding/decoding DCF77 are included in the sample program. Other time codings can be implemented in the PLC itself. (See [Examples > DCF77 \[▶ 22\]](#))

Application example: external synchronization of a system environment (B) from a non-modifiable system environment (A)



i In the following sections, system environment (A) is regarded as the master system (from any manufacturer), system environment (B) is regarded as the slave system (TwinCAT only).

3 Installation

No separate setup is required for the function TF6225 TC3 EtherCAT External Sync. All the required components are supplied directly with the TwinCAT setup.

3.1 System requirements

Technical data	Description
Target system	Windows 7/8/10 Windows CE
TwinCAT version	TwinCAT 3.1 Build 4020.32
Minimum TwinCAT level	TC1200 TC3 PLC

3.2 Licensing

The TwinCAT 3 Function can be activated as a full version or as a 7-day test version. Both license types can be activated via the TwinCAT 3 development environment (XAE).

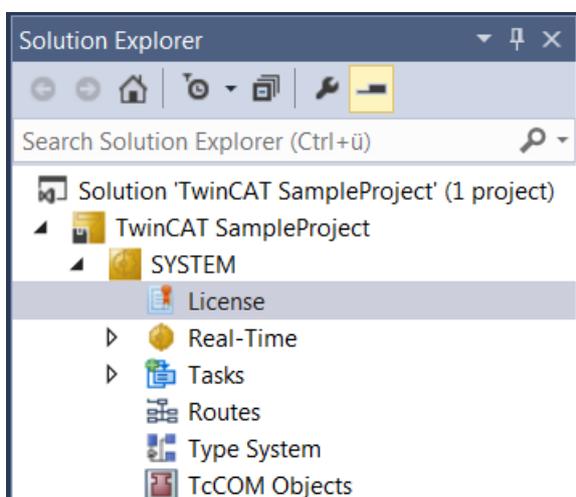
The licensing of a TwinCAT 3 Function is described below. The description is divided into the following sections:

- [Licensing a 7-day test version \[► 8\]](#)
- [Licensing a full version \[► 10\]](#)

Further information on TwinCAT 3 licensing can be found in the “Licensing” documentation in the Beckhoff Information System (TwinCAT 3 > [Licensing](#)).

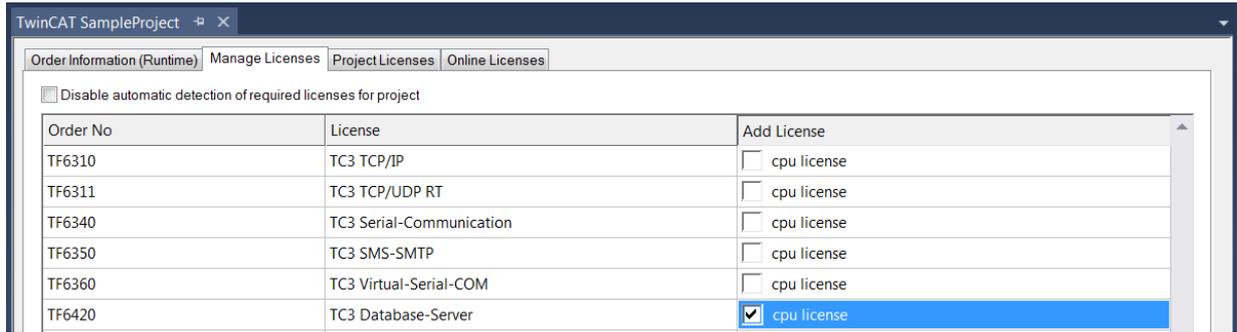
Licensing a 7-day test version

1. Start the TwinCAT 3 development environment (XAE).
2. Open an existing TwinCAT 3 project or create a new project.
3. If you want to activate the license for a remote device, set the desired target system. To do this, select the target system from the **Choose Target System** drop-down list in the toolbar.
 - ⇒ The licensing settings always refer to the selected target system. When the project is activated on the target system, the corresponding TwinCAT 3 licenses are automatically copied to this system.
4. In the **Solution Explorer**, double-click **License** in the **SYSTEM** subtree.

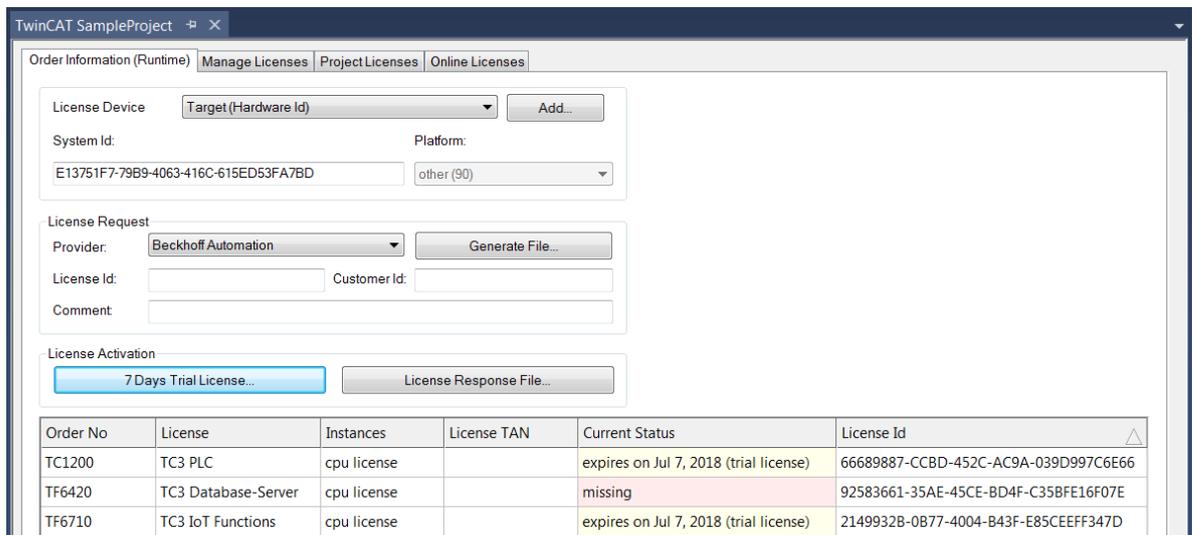


⇒ The TwinCAT 3 license manager opens.

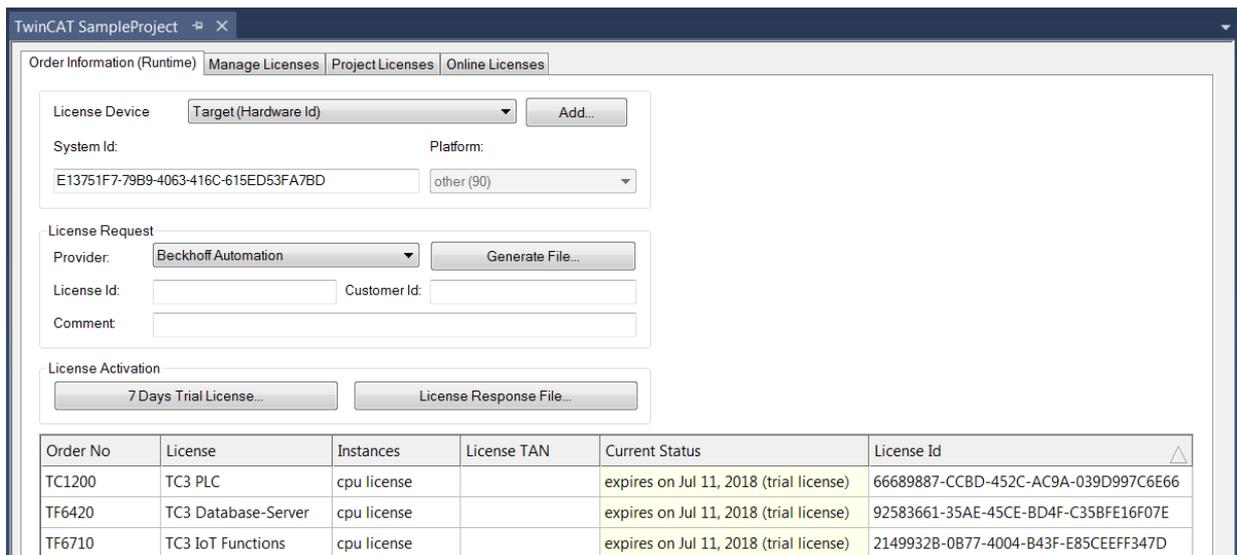
- Open the **Manage Licenses** tab. In the **Add License** column, check the check box for the license you want to add to your project (e.g. "TF6420: TC3 Database Server").



- Open the **Order Information (Runtime)** tab.
 - ⇒ In the tabular overview of licenses, the previously selected license is displayed with the status "missing".



- Click **7-Day Trial License...** to activate the 7-day trial license.



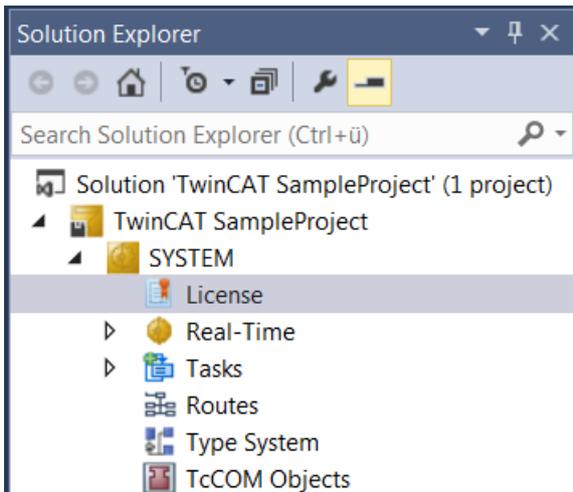
⇒ A dialog box opens, prompting you to enter the security code displayed in the dialog.

- Enter the code exactly as it appears, confirm it and acknowledge the subsequent dialog indicating successful activation.
 - ⇒ In the tabular overview of licenses, the license status now indicates the expiration date of the license.
- Restart the TwinCAT system.

⇒ The 7-day trial version is enabled.

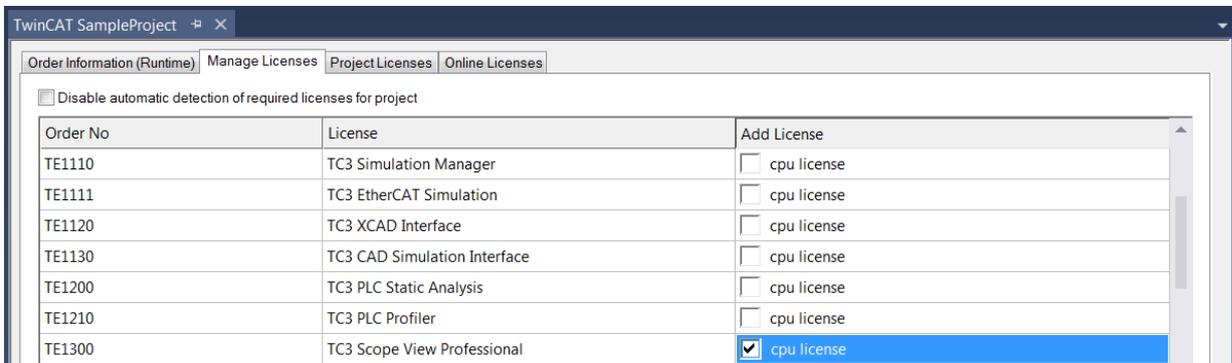
Licensing a full version

1. Start the TwinCAT 3 development environment (XAE).
2. Open an existing TwinCAT 3 project or create a new project.
3. If you want to activate the license for a remote device, set the desired target system. To do this, select the target system from the **Choose Target System** drop-down list in the toolbar.
 - ⇒ The licensing settings always refer to the selected target system. When the project is activated on the target system, the corresponding TwinCAT 3 licenses are automatically copied to this system.
4. In the **Solution Explorer**, double-click **License** in the **SYSTEM** subtree.



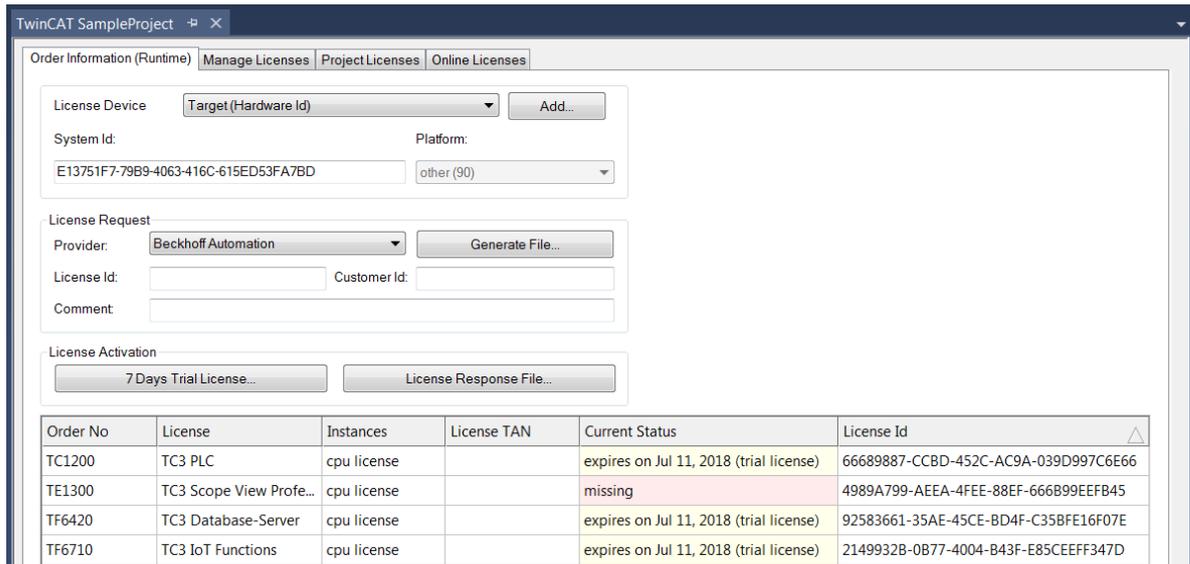
⇒ The TwinCAT 3 license manager opens.

5. Open the **Manage Licenses** tab. In the **Add License** column, check the check box for the license you want to add to your project (e.g. "TE1300: TC3 Scope View Professional").



6. Open the **Order Information** tab.

⇒ In the tabular overview of licenses, the previously selected license is displayed with the status “missing”.



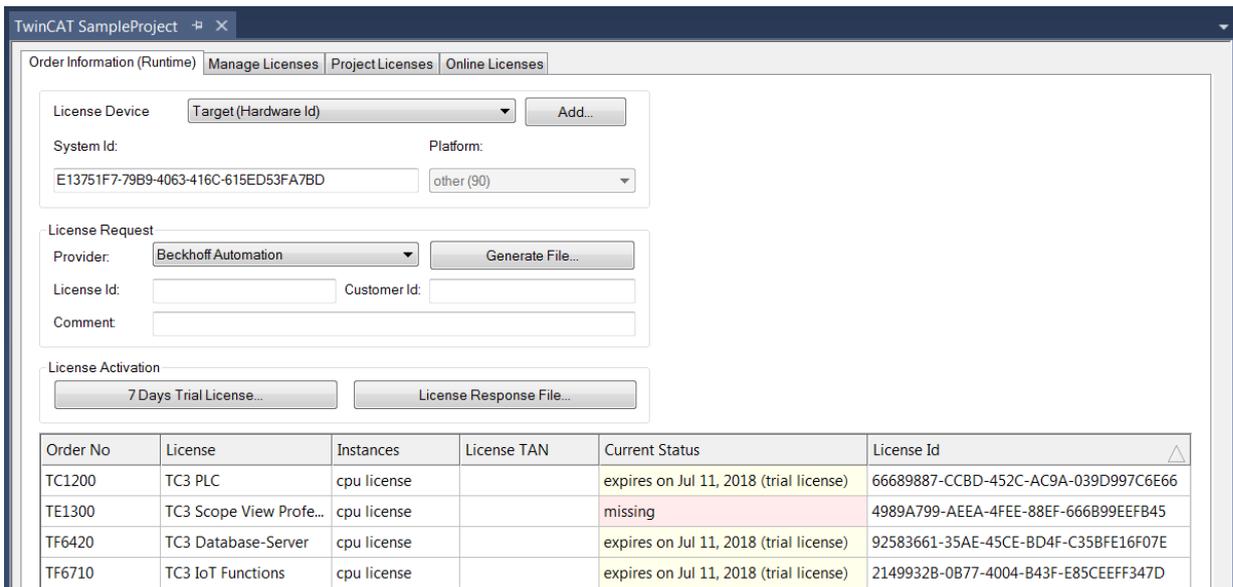
A TwinCAT 3 license is generally linked to two indices describing the platform to be licensed:

System ID: Uniquely identifies the device

Platform level: Defines the performance of the device

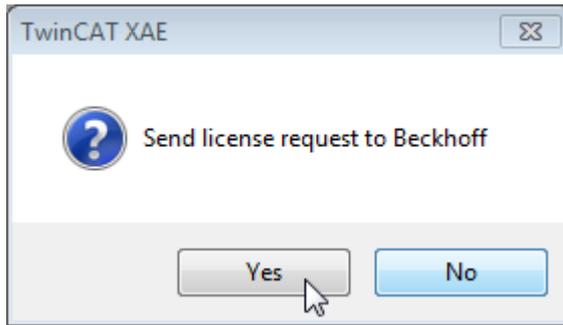
The corresponding **System Id** and **Platform** fields cannot be changed.

7. Enter the order number (**License Id**) for the license to be activated and optionally a separate order number (**Customer Id**), plus an optional comment for your own purposes (**Comment**). If you do not know your Beckhoff order number, please contact your Beckhoff sales contact.



8. Click the **Generate File...** button to create a License Request File for the listed missing license.
 - ⇒ A window opens, in which you can specify where the License Request File is to be stored. (We recommend accepting the default settings.)
9. Select a location and click **Save**.

- ⇒ A prompt appears asking whether you want to send the License Request File to the Beckhoff license server for verification:



- Click **Yes** to send the License Request File. A prerequisite is that an email program is installed on your computer and that your computer is connected to the internet. When you click **Yes**, the system automatically generates a draft email containing the License Request File with all the necessary information.
- Click **No** if your computer does not have an email program installed on it or is not connected to the internet. Copy the License Request File onto a data storage device (e.g. a USB stick) and send the file from a computer with internet access and an email program to the Beckhoff license server (tclicense@beckhoff.com) by email.

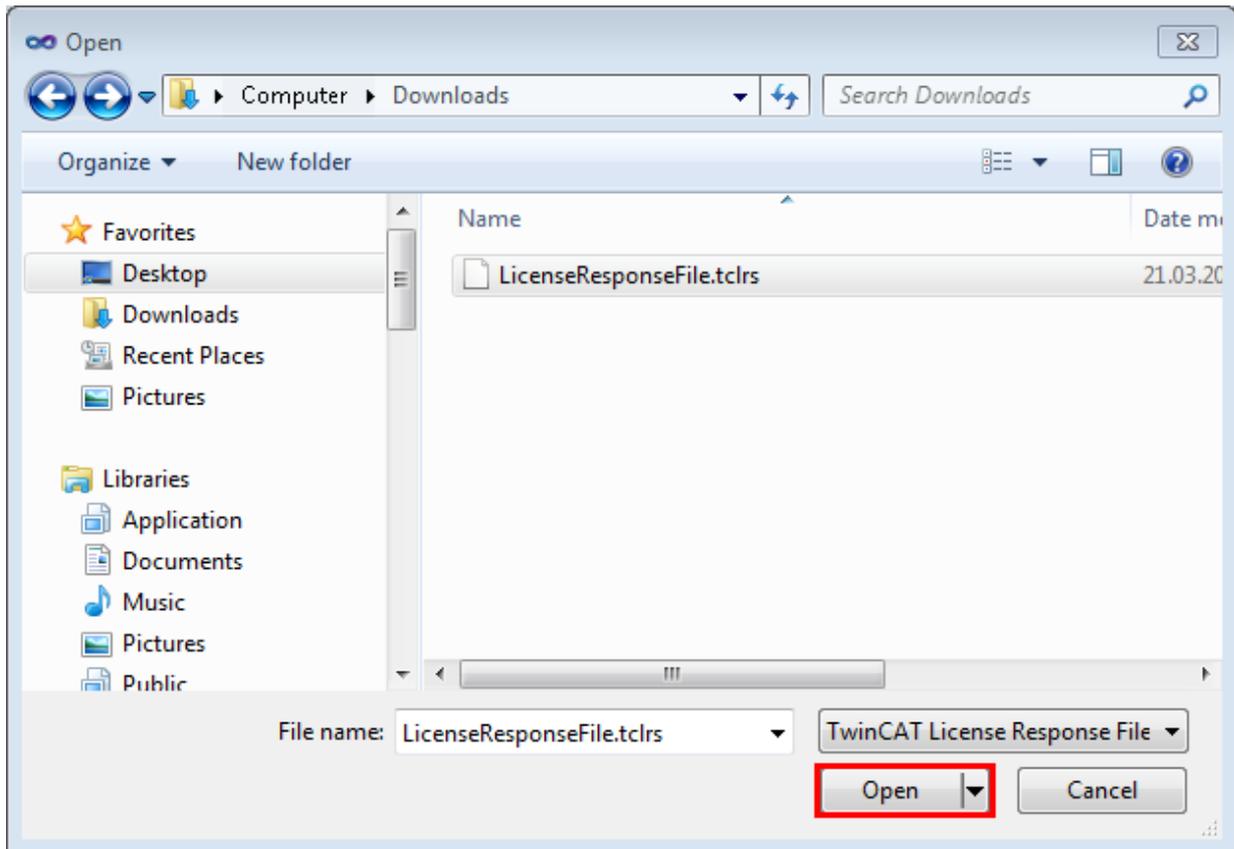
10. Send the License Request File.

- ⇒ The License Request File is sent to the Beckhoff license server. After receiving the email, the server compares your license request with the specified order number and returns a License Response File by email. The Beckhoff license server returns the License Response File to the same email address from which the License Request File was sent. The License Response File differs from the License Request File only by a signature that documents the validity of the license file content. You can view the contents of the License Response File with an editor suitable for XML files (e.g. "XML Notepad"). The contents of the License Response File must not be changed, otherwise the license file becomes invalid.

11. Save the License Response File.

12. To import the license file and activate the license, click **License Response File...** in the **Order Information** tab.

13. Select the License Response File in your file directory and confirm the dialog.



⇒ The License Response File is imported and the license it contains is activated.
Existing demo licenses will be removed.

14. Restart the TwinCAT system.

⇒ The license becomes active when TwinCAT is restarted. The product can be used as a full version.
During the TwinCAT restart the license file is automatically copied to the directory `...\\TwinCAT\\3.1\\Target\\License` on the respective target system.

4 Technical introduction

When external synchronization is applied, two system times meet: the controlled/time-receiving TwinCAT system with its distributed clock, and the time-sending system with its encoder clock.

Synchronization process:

- Two time stamps (one external and one internal) must be transferred periodically to the time-receiving TwinCAT system.
- To form a time stamp pair, the value of both times is recorded at the same time. The "internal clock" is always based on the system's own distributed clock time. The "external clock" is the time from the external, time-giving system.
- The time stamp pairs are determined periodically, e.g. every 100 ms.
- If these time stamp pairs are repeatedly sent to the receiving TwinCAT real-time system, the system is able to determine the deviation trend and can readjust itself until clock synchronization is achieved. The remaining offset is offered for linking through variables in the EtherCAT IO tree.

Essentially, TwinCAT synchronization is not designed to minimize or compensate the offset. If the initial difference between the two systems is large (weeks, months), this would mean a very long waiting time. The readjusting TwinCAT system therefore establishes clock synchronization within a few seconds and maintains the offset.

Offset step changes can occur if:

- the real-time behavior no longer permits tracking
- leap seconds/time changes etc. are received via the external time

The readjusted control should be designed in such a way that it can cope with offset step changes.

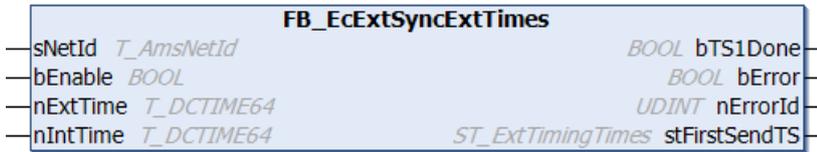
The quality of the time control that can be achieved depends, among other things, on how accurately the external clock can be read and how the "new signal" event can be fed into the time-receiving system. The jitter-free nature of this external time stamp must be magnitudes better than the intended control objective.

Example: The jitter associated with the edge output of commercially available direct DCF77 receivers (German radio time on longwave) can be such that the readjustment is disturbed up to the ms range, even taking into account the missing 59th second. This leads to step changes in the offset display.

5 PLC API

5.1 Function blocks

5.1.1 FB_EcExtSyncExtTimes



This function block enables or disables the synchronization of the internal and external clocks. The function block [FB_EcExtSyncIsSynchronized](#) [[16](#)] can be used to check the synchronization.

This function block is required for the synchronization. It:

- enables or disables the synchronization of the internal and external clocks;
- periodically sends the time stamp pair nExtTime / nIntTime to the TwinCAT real-time system via ADS.

The function block should be called at most every 2-5 ms. A new/fresh time stamp pair should be assigned to the function block every time it is called, otherwise the call is meaningless. The frequency of the function block call influences the quality of the time control. The more often the function block is called, the better the readjusted system can work. If the time stamp quality is good, one call per second, e.g. through a PPS signal (PulsePerSecond), may be sufficient.

VAR_INPUT

```
VAR_INPUT
  sNetId    : T_AmsNetId;
  bEnable   : BOOL;
  nExtTime  : T_DCTIME64;
  nIntTime  : T_DCTIME64;
END_VAR
```

sNetId: String containing the AMS network ID of the EtherCAT master device (type: T_AMSNetId)

bEnable: bEnable = TRUE activates the synchronization. bEnable = FALSE deactivates the synchronization.

nExtTime: external time stamp

nIntTime: internal time stamp

VAR_OUTPUT

```
VAR_OUTPUT
  bTS1Done   : BOOL;
  bError     : BOOL;
  nErrorId   : UDINT;
  stFirstSendTS : ST_ExtTimingTimes;
END_VAR
```

bTS1Done: This output is set when the synchronization is enabled for the first time and the internal and external time stamps are written to the EtherCAT master.

bError: This output is set if an error occurs during command execution.

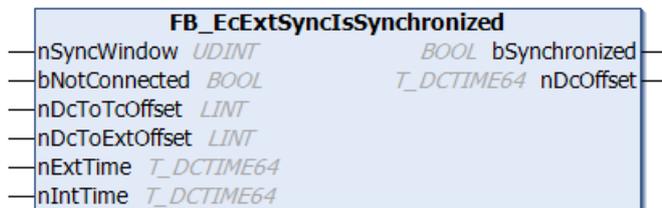
nErrorId: This output returns the error code, if the bError output is set.

stFirstSendTS: This output is set to the internal and external time stamps that are written to the EtherCAT master. (Type: [ST_ExtTimingTimes](#) [[17](#)])

Requirements

Development environment	Target platform	PLC libraries to be integrated (category group)
TwinCAT v3.1.0	PC or CX (x86, x64, ARM)	Tc3_EtherCATExtSync

5.1.2 FB_EcExtSyncIsSynchronized



This function block checks whether the internal and external clocks run synchronously (bSynchronized). For this purpose, the system checks whether the difference between the clocks over 10 cycles is within the specified synchronization window nSyncWindow. In the event of invalid data (bNotConnected = TRUE), the synchronization is reported as failed.

The function block is not absolutely necessary for readjustment, but it can be used for information purposes.

VAR_INPUT

```

VAR_INPUT
  nSyncWindow      : UDINT;
  bNotConnected    : BOOL;
  nDcToTcOffset    : LINT;
  nDcToExtOffset   : LINT;
  nExtTime         : T_DCTIME64;
  nIntTime         : T_DCTIME64;
END_VAR
    
```

nSyncWindow: Time window [ns] within which the internal and external time are regarded as synchronized

bNotConnected: TRUE, if the data is invalid (corresponds to the WcState of the EL6692)

nDcToTcOffset: Offset between the distributed clock time and the TwinCAT time [ns] from the InfoData of the EtherCAT master

nDcToExtOffset: Offset between the distributed clock time and the external time [ns] from the InfoData of the EtherCAT master

nExtTime: external time stamp

nIntTime: internal time stamp

VAR_OUTPUT

```

VAR_OUTPUT
  bSynchronized    : BOOL;
  nDcOffset        : T_DCTIME64;
END_VAR
    
```

bSynchronized: TRUE if the difference between the internal and external time over at least 10 cycles is within the synchronization window

nDcOffset: Magnitude of the current difference between the internal and external time [ns]

Requirements

Development environment	Target platform	PLC libraries to be integrated (category group)
TwinCAT v3.1.0	PC or CX (x86, x64, ARM)	Tc3_EtherCATExtSync

5.2 Data types

5.2.1 ST_ExtTimingTimes

Structure with the time stamps of the internal and external clock.

```
TYPE ST_ExtTimingStatus :  
STRUCT  
    nDcIntTs : T_DCTIME64;  
    nDcExtTs : T_DCTIME64;  
END_STRUCT  
END_TYPE
```

nDcIntTs: Time stamp of the internal clock

nDcExtTs: Time stamp of the external clock

6 Examples

6.1 EtherCAT External Sync

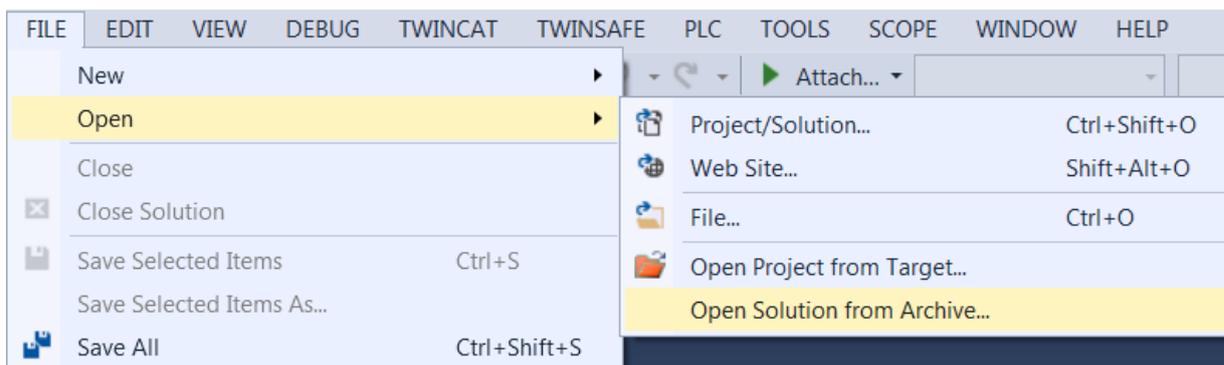
i Using the sample programs

This document contains sample applications of our products for certain areas of application. The application notices provided here are based on typical features of our products and only serve as samples. The notices contained in this document explicitly do not refer to specific applications. The user is therefore responsible for assessing and deciding whether the product is suitable for a particular application. We accept no responsibility for the completeness and correctness of the source code contained in this document. We reserve the right to modify the content of this document at any time and accept no responsibility for errors and missing information.

Download: https://infosys.beckhoff.com/content/1033/tf6225_tc3_ethercat_external_sync/Resources/zip/3864740235.zip

Load the sample program (tnzip file/TwinCAT 3)

1. Save the ZIP archive locally on your hard disk and extract the file.
2. Open TwinCAT 3 Engineering.
3. In the **File > Open** menu, select the command **Open Solution from Archive**.



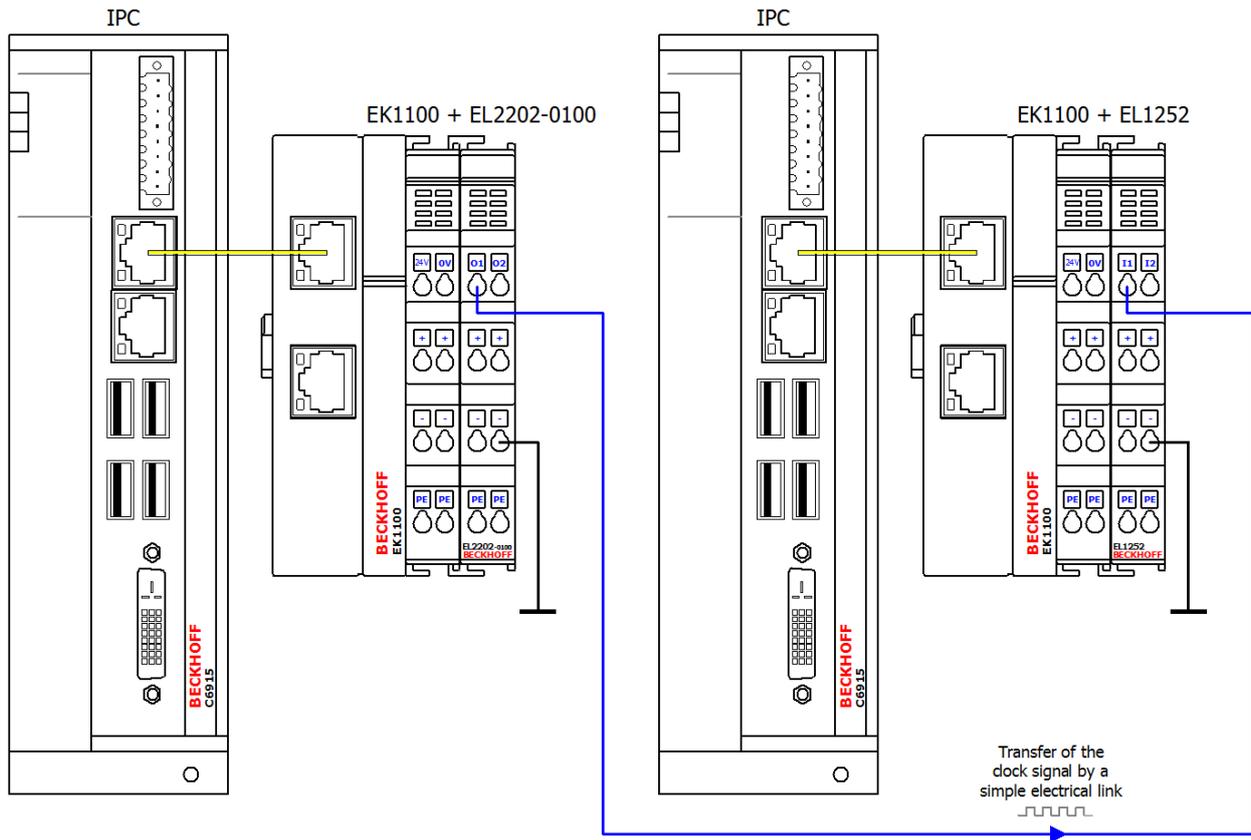
4. In the **Open** dialog that opens, select the previously unpacked .tnzip file (sample program) and confirm the dialog with **Open**.
 - ⇒ The **Select folder for new solution** selection window opens.
5. Select a destination directory for saving the project and confirm the dialog with **Select folder**.
 - ⇒ The Tc3_EtherCATExtSyncSample solution is loaded.

Further information on the general procedure for commissioning the PLC or starting the program can be found in the terminal documentation and in the [EtherCAT System Documentation](#).

6.1.1 Configuration

Sample configuration for TF6225 TC3 EtherCAT External Sync

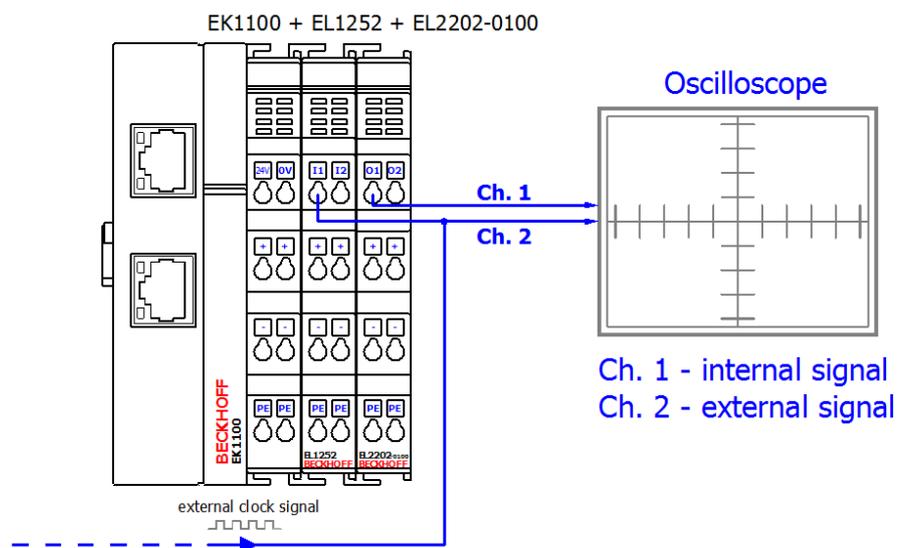
The operating principle of the TF6225 can be illustrated through the configuration of two independent IPC systems. The diagram shows the hardware structure of the master system (clock generator) on the left and the slave system (clock receiver) on the right.



Recording of the internal and external clock with an oscilloscope

After commissioning the system, you can monitor the synchronization via the TwinCAT 3 Scope View, for example.

You can also use an oscilloscope to record the internal and external clocks on a time base. To do this, add an EL2202-0100 EtherCAT Terminal to the slave system. The external clock of the master system can be recorded via the input of the EL1252 EtherCAT Terminal and can serve as a reference (trigger), for example. The internal clock of the slave system can be output via a toggling bit at the additionally connected EL2202-0100 EtherCAT Terminal. If external synchronization is not active, the time drift can be recognized by the fact that the two signals are not fixed in time relative to each other.

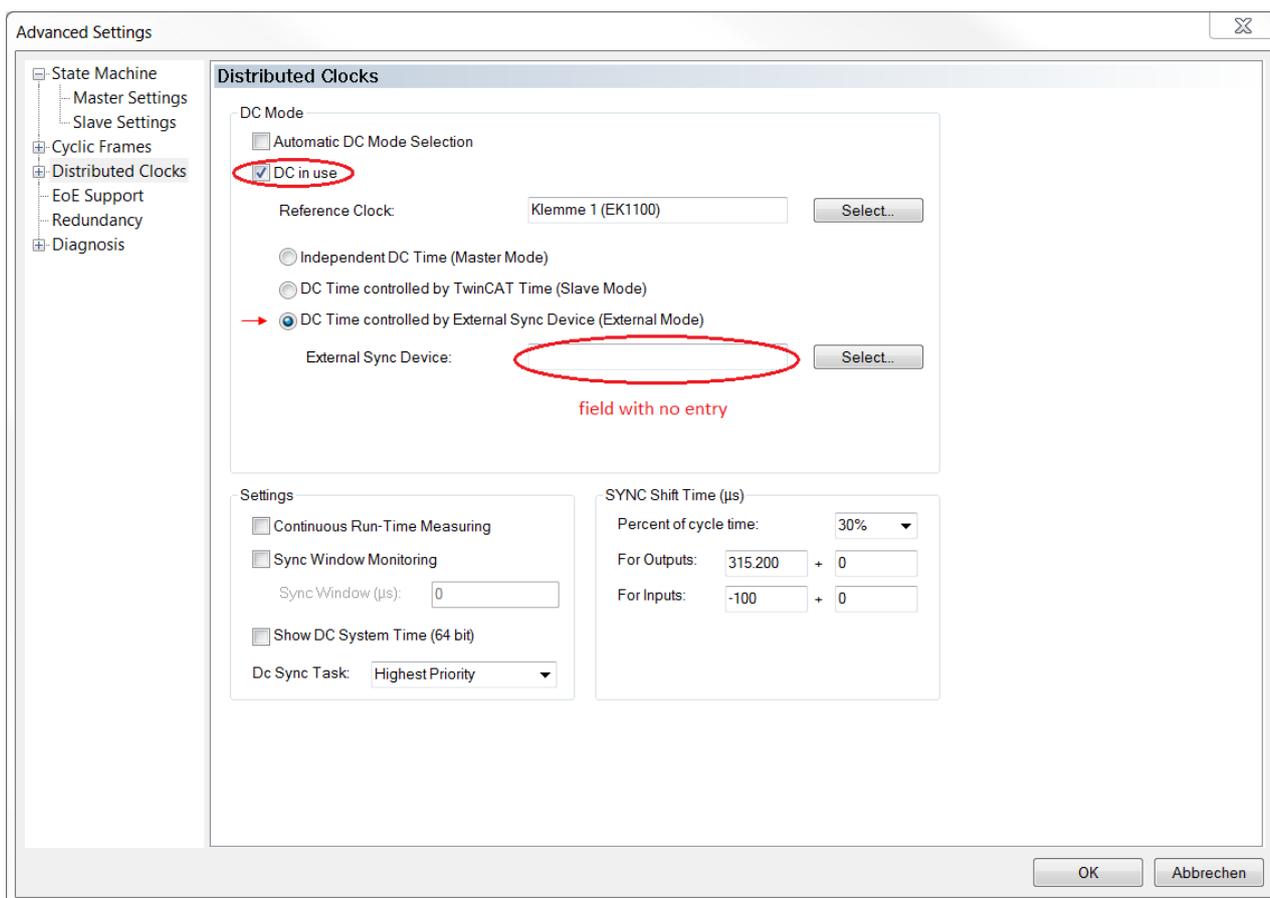




The EL2202-0100 EtherCAT Terminal is the extended distributed clocks version of the EL2202. It can be obtained from Beckhoff, or it may be possible to set it up by reprogramming the EL2202. Further information can be found in the documentation [EL2202, EL2252 – two-channel digital output terminal](#).

EtherCAT master configuration of the slave system for external DC synchronization via the EL1252 in TwinCAT 3 Engineering

The EtherCAT master of the slave system for external synchronization can be configured in TwinCAT 3 Engineering under advanced device settings. To open the advanced settings, double-click on the EtherCAT device in the TwinCAT project tree. Select the **EtherCAT** tab and click **Advanced Settings...** Select the **Distributed Clocks** entry in the navigation tree in the dialog that opens. Configure the EtherCAT master of the slave system according to the following diagram.



6.1.2 Notes

Notes on the sample program of the slave system

- The sample program shows the slave system (system environment B). This requires a controller with an EL1252 EtherCAT Terminal. You can either use an Embedded PC, to which the terminal is connected on the right-hand side, or an IPC with an EtherCAT connection, e.g. an RJ-45 connection to the EK1100 Coupler with the terminal. (See also [Configuration](#) [▶ 18])
- If necessary, read in the I/O configuration again and link the variables as follows:
 - nIntTime → EL1252, PDO: Latch/ LatchPos1
 - aEcMasterAmsNetId → device (EtherCAT), PDO: InfoData/ AmsNetId
 - bOut → (optional) EL2202-0100, PDO: Channel 1/ output

- In the sample program, the internal time value (nExtTime) is incremented according to the expected input signal (50 ms period → rEventTimeStep = 5·10⁷) when an incoming positive edge event is detected at the EL1252 input. Select the appropriate value for rEventTimeStep, depending on the expected cycle duration:

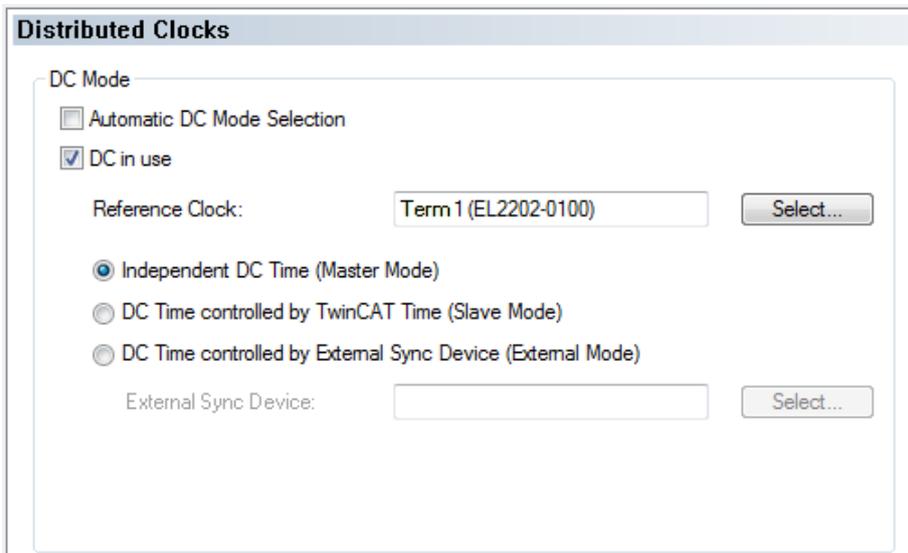
```
nExtTime := nExtTime + LREAL_TO_ULINT(rEventTimeStep);
```

 The addition of a constant time value to a base time value for the time stamp of the external time depends on an incoming (positive) edge. This causes an external time stamp to be mapped by the external clock. An edge is always determined by comparing the current internal time stamp supplied by the EL1252 with the last time stamp in each task run:

```
IF(current_intTimeSatmp<>last_intTimeSatmp) THEN
```
- Optionally, you can add a YT-Scope-View to visualize the project and record the following variables from the MAIN program:
 - nDcOffset (the result of FB_EcExtSynclsSynchronized from nExtTime and nIntTime)
 - bExtTime_Digits and blntTime_Digits for visualizing the temporal drift or synchronous operation after activation of the synchronization process
 - bSynchronized to show that the synchronization has taken place
- The variable bEnableExtSync can be set to TRUE at a desired time.

General information on the master system

- The master system is not included in this sample. In principle, any timer that can provide a 24 V square wave signal could be used as a master system. The master system shown in the diagram has enabled distributed clocks (DC) for the EL2202-0100 EtherCAT Terminal and set a corresponding task cycle time of 25 ms for an output signal with T = 50 ms. The following code line in the (POU) MAIN is used for generating the output signal: bOut := NOT bOut;
- The EtherCAT master of the master system has distributed clocks (DC) enabled. Further information regarding the setting can be found in the [EtherCAT System Documentation](#) in the Setup section under TwinCAT System Manager > Notes on Distributed Clocks. An EL2202-0100 EtherCAT Terminal in the master system acts as DC timer and also issues the output signal.

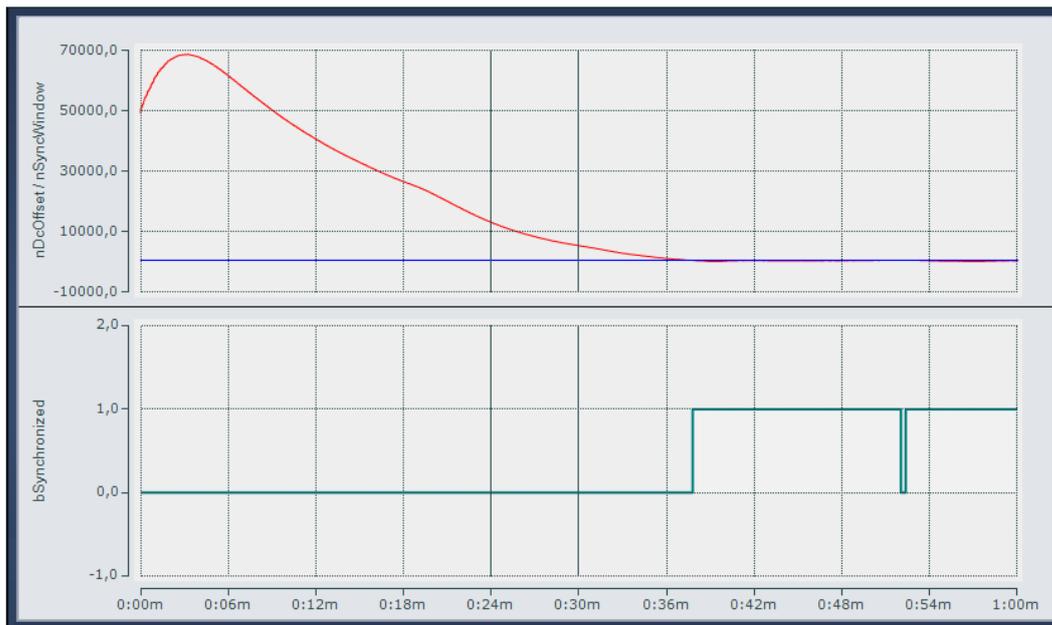


- Larger cycle durations, e.g. 1 s, can be converted using a counter variable.

6.1.3 Results

ScopeView recording

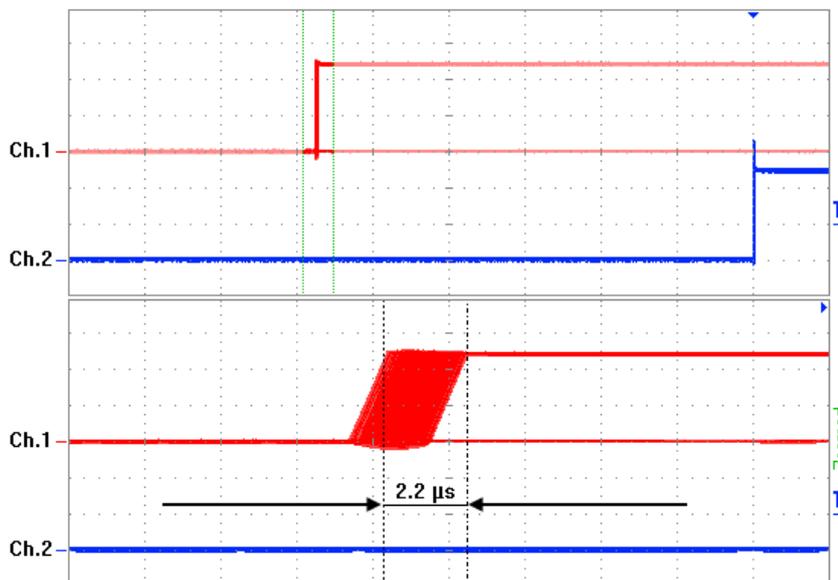
After the activation of the external synchronization on the part of the slave system (TRUE => bEnableExtSync), the deviation between the internal and external time values (nDcOffset), the synchronization window nSyncWindow and the actual synchronization is recorded with ScopeView (bSynchronized):



Oscilloscope recording

The upper part of the following diagram shows the time offset between the two signals. The lower part shows an enlarged section relating to channel 1, to illustrate the remaining jitter of the sample slave system. In this example a clock synchronization of $< \pm 1.5 \mu\text{s}$ is achieved.

- Channel 1: Internal slave PLC clock pulse
- Channel 2: External master PLC clock pulse



6.2 DCF77

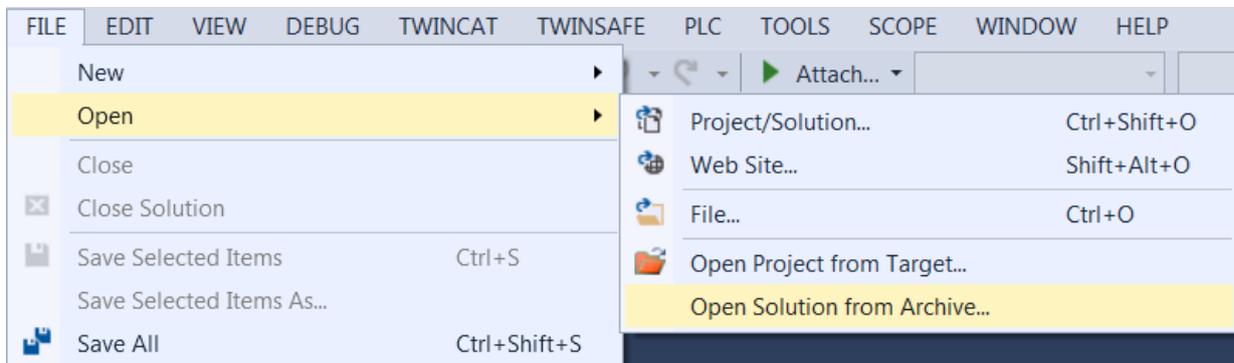
i Using the sample programs

This document contains sample applications of our products for certain areas of application. The application notices provided here are based on typical features of our products and only serve as samples. The notices contained in this document explicitly do not refer to specific applications. The user is therefore responsible for assessing and deciding whether the product is suitable for a particular application. We accept no responsibility for the completeness and correctness of the source code contained in this document. We reserve the right to modify the content of this document at any time and accept no responsibility for errors and missing information.

Download: https://infosys.beckhoff.com/content/1033/tf6225_tc3_ethercat_external_sync/Resources/zip/4001783435.zip

Load the sample program (tnzip file/TwinCAT 3)

1. Save the ZIP archive locally on your hard disk and extract the file.
2. Open TwinCAT 3 Engineering.
3. In the **File > Open** menu, select the command **Open Solution from Archive**.



4. In the **Open** dialog that opens, select the previously unpacked .tnzip file (sample program) and confirm the dialog with **Open**.
 - ⇒ The **Select folder for new solution** selection window opens.
5. Select a destination directory for saving the project and confirm the dialog with **Select folder**.
 - ⇒ The Tc3_EtherCATExtSyncSample solution is loaded.

Further information on the general procedure for commissioning the PLC or starting the program can be found in the terminal documentation and in the [EtherCAT System Documentation](#).

7 Appendix

7.1 ADS Return Codes

Error codes: [0x000 \[▶ 24\]...](#), [0x500 \[▶ 24\]...](#), [0x700 \[▶ 25\]...](#), [0x1000 \[▶ 26\]...](#)

Global Error Codes

Hex	Dec	Description
0x0	0	no error
0x1	1	Internal error
0x2	2	No Rtime
0x3	3	Allocation locked memory error
0x4	4	Insert mailbox error
0x5	5	Wrong receive HMSG
0x6	6	target port not found
0x7	7	target machine not found
0x8	8	Unknown command ID
0x9	9	Bad task ID
0xA	10	No IO
0xB	11	Unknown ADS command
0xC	12	Win 32 error
0xD	13	Port not connected
0xE	14	Invalid ADS length
0xF	15	Invalid AMS Net ID
0x10	16	Low Installation level
0x11	17	No debug available
0x12	18	Port disabled
0x13	19	Port already connected
0x14	20	ADS Sync Win32 error
0x15	21	ADS Sync Timeout
0x16	22	ADS Sync AMS error
0x17	23	ADS Sync no index map
0x18	24	Invalid ADS port
0x19	25	No memory
0x1A	26	TCP send error
0x1B	27	Host unreachable
0x1C	28	Invalid AMS fragment

Router Error Codes

Hex	Dec	Name	Description
0x500	1280	ROUTERERR_NOLOCKEDMEMORY	No locked memory can be allocated
0x501	1281	ROUTERERR_RESIZEMEMORY	The size of the router memory could not be changed
0x502	1282	ROUTERERR_MAILBOXFULL	The mailbox has reached the maximum number of possible messages. The current sent message was rejected
0x503	1283	ROUTERERR_DEBUGBOXFULL	The mailbox has reached the maximum number of possible messages. The sent message will not be displayed in the debug monitor
0x504	1284	ROUTERERR_UNKNOWNPORTTYPE	Unknown port type
0x505	1285	ROUTERERR_NOTINITIALIZED	Router is not initialized
0x506	1286	ROUTERERR_PORTALREADYINUSE	The desired port number is already assigned
0x507	1287	ROUTERERR_NOTREGISTERED	Port not registered
0x508	1288	ROUTERERR_NOMOREQUEUES	The maximum number of Ports reached
0x509	1289	ROUTERERR_INVALIDPORT	Invalid port
0x50A	1290	ROUTERERR_NOTACTIVATED	TwinCAT Router not active

General ADS Error Codes

Hex	Dec	Name	Description
0x700	1792	ADSERR_DEVICE_ERROR	error class <device error>
0x701	1793	ADSERR_DEVICE_SRVNOTSUPP	Service is not supported by server
0x702	1794	ADSERR_DEVICE_INVALIDGRP	invalid index group
0x703	1795	ADSERR_DEVICE_INVALIDOFFSET	invalid index offset
0x704	1796	ADSERR_DEVICE_INVALIDACCESS	reading/writing not permitted
0x705	1797	ADSERR_DEVICE_INVALIDSIZE	parameter size not correct
0x706	1798	ADSERR_DEVICE_INVALIDDATA	invalid parameter value(s)
0x707	1799	ADSERR_DEVICE_NOTREADY	device is not in a ready state
0x708	1800	ADSERR_DEVICE_BUSY	device is busy
0x709	1801	ADSERR_DEVICE_INVALIDCONTEXT	invalid context (must be in Windows)
0x70A	1802	ADSERR_DEVICE_NOMEMORY	out of memory
0x70B	1803	ADSERR_DEVICE_INVALIDPARM	invalid parameter value(s)
0x70C	1804	ADSERR_DEVICE_NOTFOUND	not found (files, ...)
0x70D	1805	ADSERR_DEVICE_SYNTAX	syntax error in command or file
0x70E	1806	ADSERR_DEVICE_INCOMPATIBLE	objects do not match
0x70F	1807	ADSERR_DEVICE_EXISTS	object already exists
0x710	1808	ADSERR_DEVICE_SYMBOLNOTFOUND	symbol not found
0x711	1809	ADSERR_DEVICE_SYMBOLVERSIONINVAL	symbol version invalid
0x712	1810	ADSERR_DEVICE_INVALIDSTATE	server is in invalid state
0x713	1811	ADSERR_DEVICE_TRANSMODENOTSUPP	AdsTransMode not supported
0x714	1812	ADSERR_DEVICE_NOTIFYHNDINVALID	Notification handle is invalid
0x715	1813	ADSERR_DEVICE_CLIENTUNKNOWN	Notification client not registered
0x716	1814	ADSERR_DEVICE_NOMOREHDLS	no more notification handles
0x717	1815	ADSERR_DEVICE_INVALIDWATCHSIZE	size for watch too big
0x718	1816	ADSERR_DEVICE_NOTINIT	device not initialized
0x719	1817	ADSERR_DEVICE_TIMEOUT	device has a timeout
0x71A	1818	ADSERR_DEVICE_NOINTERFACE	query interface failed
0x71B	1819	ADSERR_DEVICE_INVALIDINTERFACE	wrong interface required
0x71C	1820	ADSERR_DEVICE_INVALIDCLSID	class ID is invalid
0x71D	1821	ADSERR_DEVICE_INVALIDOBJID	object ID is invalid
0x71E	1822	ADSERR_DEVICE_PENDING	request is pending
0x71F	1823	ADSERR_DEVICE_ABORTED	request is aborted
0x720	1824	ADSERR_DEVICE_WARNING	signal warning
0x721	1825	ADSERR_DEVICE_INVALIDARRAYIDX	invalid array index
0x722	1826	ADSERR_DEVICE_SYMBOLNOTACTIVE	symbol not active
0x723	1827	ADSERR_DEVICE_ACCESSDENIED	access denied
0x724	1828	ADSERR_DEVICE_LICENSENOTFOUND	missing license
0x725	1829	ADSERR_DEVICE_LICENSEEXPIRED	license expired
0x726	1830	ADSERR_DEVICE_LICENSEEXCEEDED	license exceeded
0x727	1831	ADSERR_DEVICE_LICENSEINVALID	license invalid
0x728	1832	ADSERR_DEVICE_LICENSESYSTEMID	license invalid system id
0x729	1833	ADSERR_DEVICE_LICENSENOTTIMELIMIT	license not time limited
0x72A	1834	ADSERR_DEVICE_LICENSEFUTUREISSUE	license issue time in the future
0x72B	1835	ADSERR_DEVICE_LICENSESETIMETOLONG	license time period to long
0x72c	1836	ADSERR_DEVICE_EXCEPTION	exception occurred during system start
0x72D	1837	ADSERR_DEVICE_LICENSEDUPPLICATED	License file read twice
0x72E	1838	ADSERR_DEVICE_SIGNATUREINVALID	invalid signature
0x72F	1839	ADSERR_DEVICE_CERTIFICATEINVALID	public key certificate
0x740	1856	ADSERR_CLIENT_ERROR	Error class <client error>
0x741	1857	ADSERR_CLIENT_INVALIDPARM	invalid parameter at service
0x742	1858	ADSERR_CLIENT_LISTEMPTY	polling list is empty
0x743	1859	ADSERR_CLIENT_VARUSED	var connection already in use
0x744	1860	ADSERR_CLIENT_DUPLINVOKEID	invoke ID in use
0x745	1861	ADSERR_CLIENT_SYNCTIMEOUT	timeout elapsed

Hex	Dec	Name	Description
0x746	1862	ADSERR_CLIENT_W32ERROR	error in win32 subsystem
0x747	1863	ADSERR_CLIENT_TIMEOUTINVALID	Invalid client timeout value
0x748	1864	ADSERR_CLIENT_PORTNOTOPEN	ads-port not opened
0x750	1872	ADSERR_CLIENT_NOAMSADDR	internal error in ads sync
0x751	1873	ADSERR_CLIENT_SYNCINTERNAL	hash table overflow
0x752	1874	ADSERR_CLIENT_ADDHASH	key not found in hash
0x753	1875	ADSERR_CLIENT_REMOVEHASH	no more symbols in cache
0x754	1876	ADSERR_CLIENT_NOMORESVM	invalid response received
0x755	1877	ADSERR_CLIENT_SYNCRESINVALID	sync port is locked

RTime Error Codes

Hex	Dec	Name	Description
0x1000	4096	RTERR_INTERNAL	Internal fatal error in the TwinCAT real-time system
0x1001	4097	RTERR_BADTIMERPERIODS	Timer value not valid
0x1002	4098	RTERR_INVALIDTASKPTR	Task pointer has the invalid value ZERO
0x1003	4099	RTERR_INVALIDSTACKPTR	Task stack pointer has the invalid value ZERO
0x1004	4100	RTERR_PRIOEXISTS	The demand task priority is already assigned
0x1005	4101	RTERR_NOMORETCB	No more free TCB (Task Control Block) available. Maximum number of TCBs is 64
0x1006	4102	RTERR_NOMORESEMAS	No more free semaphores available. Maximum number of semaphores is 64
0x1007	4103	RTERR_NOMOREQUEUES	No more free queue available. Maximum number of queue is 64
0x100D	4109	RTERR_EXTIRQALREADYDEF	An external synchronization interrupt is already applied
0x100E	4110	RTERR_EXTIRQNOTDEF	No external synchronization interrupt applied
0x100F	4111	RTERR_EXTIRQINSTALLFAILED	The apply of the external synchronization interrupt failed
0x1010	4112	RTERR_IRQNOTLESSORQUAL	Call of a service function in the wrong context
0x1017	4119	RTERR_VMXNOTSUPPORTED	Intel VT-x extension is not supported
0x1018	4120	RTERR_VMXDISABLED	Intel VT-x extension is not enabled in system BIOS
0x1019	4121	RTERR_VMXCONTROLSSMISSING	Missing function in Intel VT-x extension
0x101A	4122	RTERR_VMXENABLEFAILS	Enabling Intel VT-x fails

TCP Winsock Error Codes

Hex	Dec	Description
0x274d	10061	A connection attempt failed because the connected party did not properly respond after a period of time, or established connection failed because connected host has failed to respond.
0x2751	10065	No connection could be made because the target machine actively refused it. This error normally occurs when you try to connect to a service which is inactive on a different host - a service without a server application.
0x274c	10060	No route to a host. A socket operation was attempted to an unreachable host
		Further Winsock error codes: Win32 Error Codes

7.2 Support and Service

Beckhoff and their partners around the world offer comprehensive support and service, making available fast and competent assistance with all questions related to Beckhoff products and system solutions.

Beckhoff's branch offices and representatives

Please contact your Beckhoff branch office or representative for [local support and service](#) on Beckhoff products!

The addresses of Beckhoff's branch offices and representatives round the world can be found on her internet pages:

<http://www.beckhoff.com>

You will also find further [documentation](#) for Beckhoff components there.

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

Phone: +49(0)5246/963-0
Fax: +49(0)5246/963-198
e-mail: info@beckhoff.com

Beckhoff Support

Support offers you comprehensive technical assistance, helping you not only with the application of individual Beckhoff products, but also with other, wide-ranging services:

- support
- design, programming and commissioning of complex automation systems
- and extensive training program for Beckhoff system components

Hotline: +49(0)5246/963-157
Fax: +49(0)5246/963-9157
e-mail: support@beckhoff.com

Beckhoff Service

The Beckhoff Service Center supports you in all matters of after-sales service:

- on-site service
- repair service
- spare parts service
- hotline service

Hotline: +49(0)5246/963-460
Fax: +49(0)5246/963-479
e-mail: service@beckhoff.com