Your Global Automation Partner

TURCK

EtherCAT Encoders
RS-25/33
RM-29/36

Manual

1 Firmware and XML file versions

Versions of the firmware and of the XML file at the date of release of this documentation:
Firmware version V3.4 visible under TwinCAT with Object 0x100A.
TurckEtherCAT.xml in Version 8, visible in the file as tag <Vendor FileVersion="8">

2 Technical Details and Encoder Features

Mechanical characteristics

Shock resistance according to EN 60068-2-27:	2500 m/s2, 6ms for Singleturn
	2000 m/s2, 6ms for Multiturn
Vibration resistance according to EN 60068-2-6:	100m/s2, 10.....2000 Hz

Operating temperature range
– 40…..+85 °C

Supply voltage and power consumption
10…30 VDC
< 160 mA under 10 VDC
< 70 mA under 24 VDC
< 60 mA under 30 VDC

Hardware-characteristics
EtherCAT IO ASIC: ET1100
Auto-Negotiation
Auto-Crossover
Function display and diagnostics with LED's

Supported standards and protocols
CAN over EtherCAT: CoE

Implemented encoder profile
CiA 406 Work Draft Version 3.2.10 of 18 February 2011

Compliance ensured with
EN 61000-4-2:2001
EN 61000-4-3:2006
EN 61000-4-4:2005
EN 61000-4-5:2007
EN 61000-4-6:2008
EN 61000-4-7:2004

EN 61000-6-4:2007
EN 61000-6-2:2006

3 Installation

Commissioning the encoder as a part of a plant consists in five steps:
1. Installation of the wiring.
2. Installation of the supply voltage.
3. Installation and project implementation using TwinCAT
4. Setting up a communication between the encoder and TwinCAT or the control.
5. Start of the application

4 Installation of the wiring

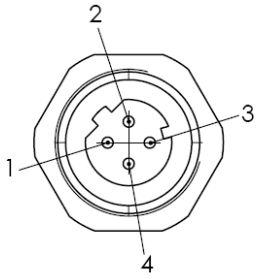
The encoder has three connections, of which two are the two Ethernet ports. They are referred to as Port IN and Port OUT in this documentation.

In picture 1 below, the position of these two ports is given by arrows „PORT IN“ and „PORT OUT“ of the sticker. The central connector is the power supply connector, which will be described in the following chapter.

Picture 1

The two Ethernet connections are D-coded M12 sockets. The assignment of the signals to the pins is represented in picture 2 and in the following table.

Signal assignment of a D coded M12-socket



Picture 2: D-coded M12 socket of the encoder

Signal name of an M12 D-coded socket	Function	Wire color	Pin number
TD+	Transmit data +	Yellow	1
TD-	Transmit data -	Orange	3
RD+	Receive data +	White	2
RD-	Receive data -	Blue	4

Signal assignment of an RJ45 to M12 cable

M12 to RJ45 direct

Signal	M12 Pin number	RJ45 Pin number
TD+	1	1
TD-	3	2
RD+	2	3
RD-	4	6

M12 to RJ45 crossover

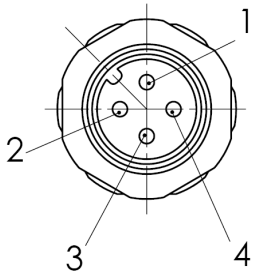
Signal	M12 Pin number	RJ45 Pin number
TD+	1	3
TD-	3	6
RD+	2	1
RD-	4	2

Recommended cable for EtherCAT – network wiring:
M12: RSSD 441-*M
M12 to RJ45 Direct: RSSD RJ45S 441-*M
M12 to RJ45 Crossover: RSSD RJ45S CR 441-*M
RJ45 Field Wireable: CONNECTOR, RJ45S

Caution:
Since EtherCAT is based on the Fast Ethernet technology, care must be taken so that the segments do in no case exceed a length of 100 m. For lengths exceeding 100m, switches must be connected in between.

5 Installation of the supply voltage

Picture 3 and the following table show the signal assignment for the pins of an A-coded power supply plug on the encoder.



Picture 3: A-coded M12-plug of the encoder

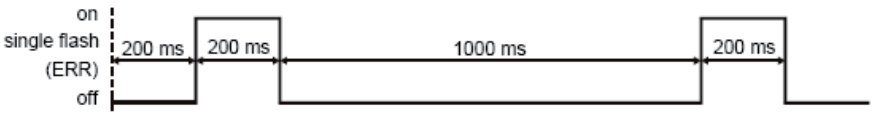
Signal on the M12 plug A-coded	Function	Pin number
PWR	10 – 30 V DC	1
		2
GND	0V	3
		4

Diagnostic - LEDs
The encoder is equipped with four diagnostic LED's, whose functionality is described below.

Picture 4

LED designation	Colour	Function description
L/A IN	Yellow	L/A IN is an LED of Port IN that is ON when the link is established, flickers on data transmission and is OFF otherwise.
L/A OUT	Yellow	L/A OUT is an LED of Port OUT that is ON when the link is established, flickers on data transmission and is OFF otherwise.
RUN	Green	The RUN LED shows the current status of the EtherCAT status machine, which can have the following values: Init, PreOperational, SafeOperational, Operational. These statuses are explicitly switchable within TwinCAT. Picture 5 shows the correspondence of the statuses of the status machine with the blinking codes.
ERROR	Red	Lights up in case of a failure.

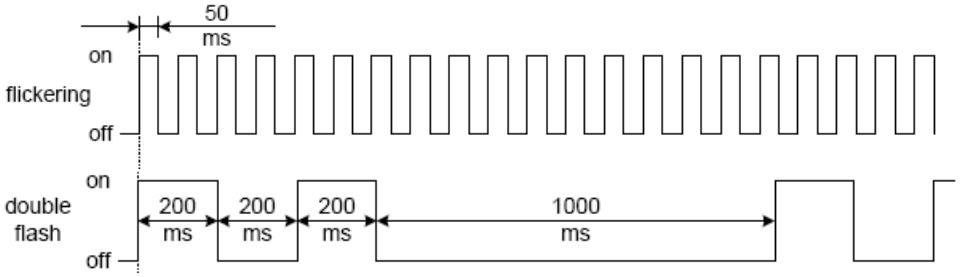
Description of the RUN LED (green)

Status	Description
Init	 LED is constantly off. Neither cyclic (process data) nor acyclic (parameters) communication is possible between the master and the encoder.
Preoperational	 In this status, no cyclic communication is possible. The green LED is blinking.
SafeOperational	The actual values are transmitted from the encoder to the master. The acyclic data communication is also operating here.
Operational	LED is constantly on. Now the cyclic data channel, which sends the encoder process data to the master, is also operating.

Picture 5

Description of the ERR LED (red)

The two pictures below illustrate graphically the definition of the ERR-LED in „flickering“ and „double flash“ statuses. These pictures are followed by a table which describes the behaviour of the ERROR-LED according to the situation.



ERR LED Status	Failure designation	Description	Example
On	Failure display caused by a failure on the microcontroller of the application	Communication error or critical application error. It a bit is set in Object 0x1001 (error register), the ERR LED is switched to permanently On.	Position error, Temperature limit value exceeded, Commissioning error, Watchdog of the process data interface between the microcontroller and the EtherCAT slave
Double Flash	Process data or EtherCAT watchdog timeout	An application watchdog timeout occurred	Sync Manager Watchdog timeout
Single Flash	Local error	The slave changed the EtherCAT status autonomously because of a local error	Device changes its EtherCAT status from Operational to SafeOperationalError because of a synchronisation error.
Blinking	Invalid configuration	General configuration error	Error leading to the fact that the master cannot modify a status change because of invalid register values or an invalid hardware configuration.
Flickering	Boot error	Boot error detected, even if INIT status has been reached.	Check sum error in the Flash of the microcontroller.
Off	No failure		

6 Installation and project implementation with TwinCAT

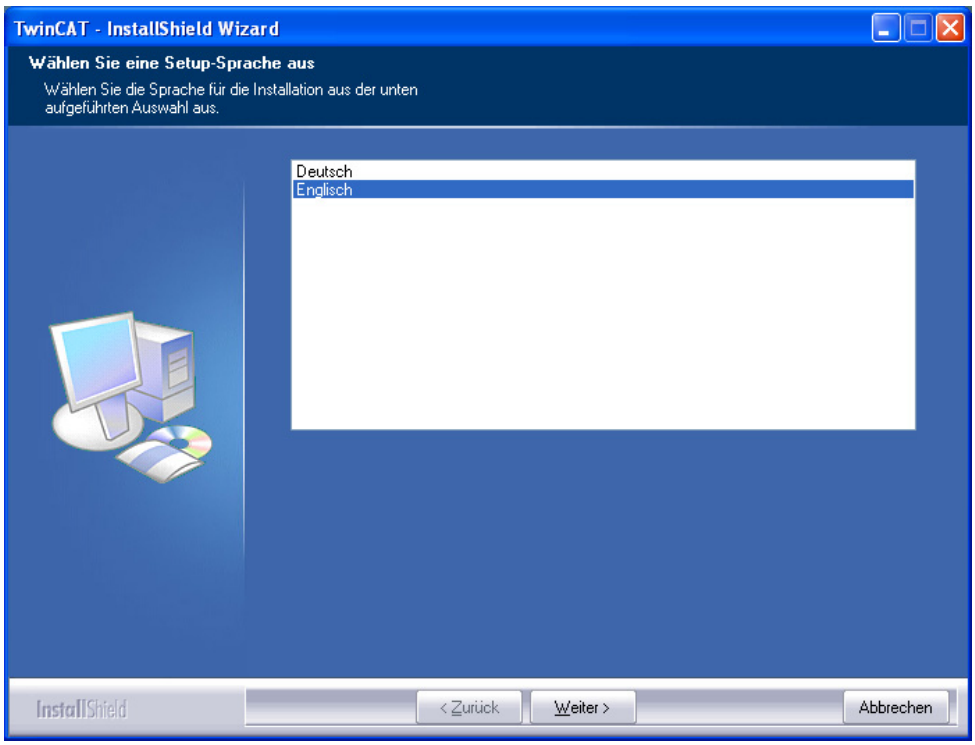
Hardware-requirements

Important:
Absolutely make sure that the PC on which you install TwinCAT includes an Ethernet chipset (LAN controller) that appears on the attached list „Network controllers supported by Beckhoff's Software Ethernet Driver". This is a necessary prerequisite for the communication between TwinCAT and the EtherCAT device.

System requirements for TwinCAT installation:
WinXP (32 bits) or Win7 (32 bits). Please note that TwinCAT cannot be installed under the 64 bits version of Win7.

Important:
Install TwinCAT imperatively in version v2.11 Build 2226 and higher! TwinCAT is stored on the DVD supplied with the encoder in this version. It is to be installed as follows.

Start file tcat_2110_2226.exe with a double click and select the language of the InstallShield as shown in picture 6.

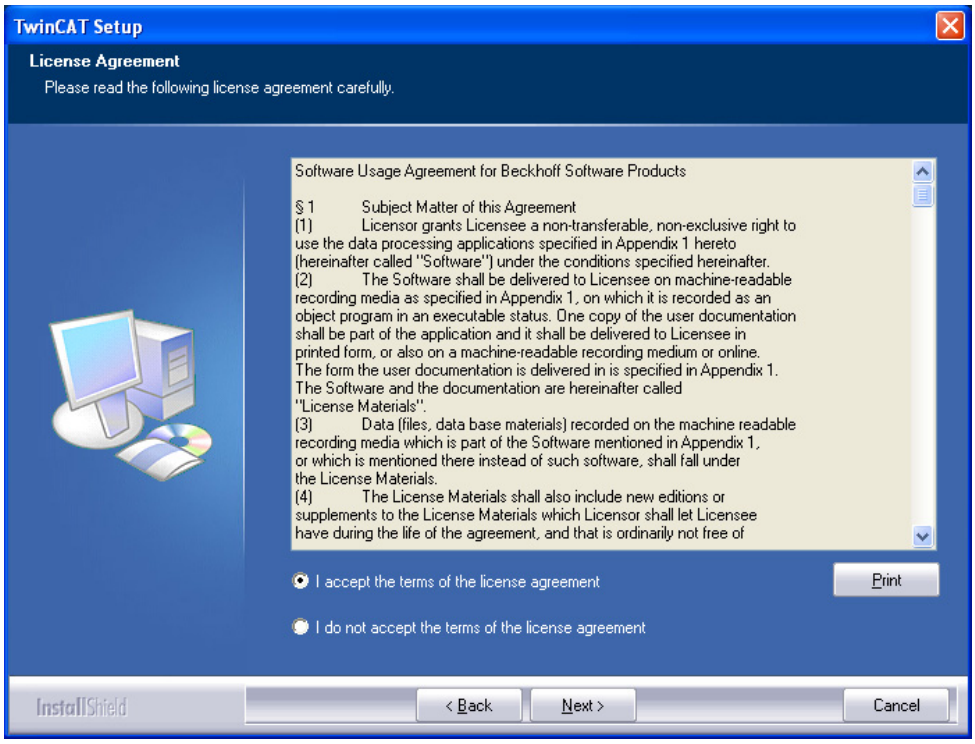


Picture 6

The **CAN bus termination** must be set **by software on Object 2102h.**

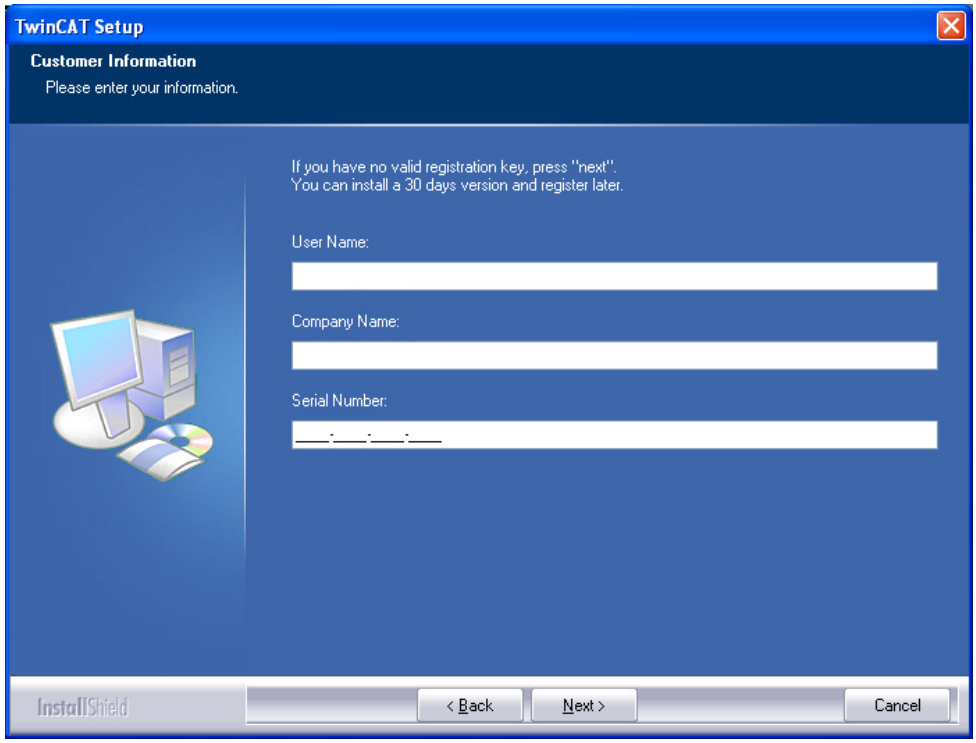
Since the CAN signals are represented on the bus as differential levels, the CAN line is comparatively insensitive to interference (EMI). Interferences always affect both lines, and therefore they almost do not modify the differential level.

Accept the license terms.



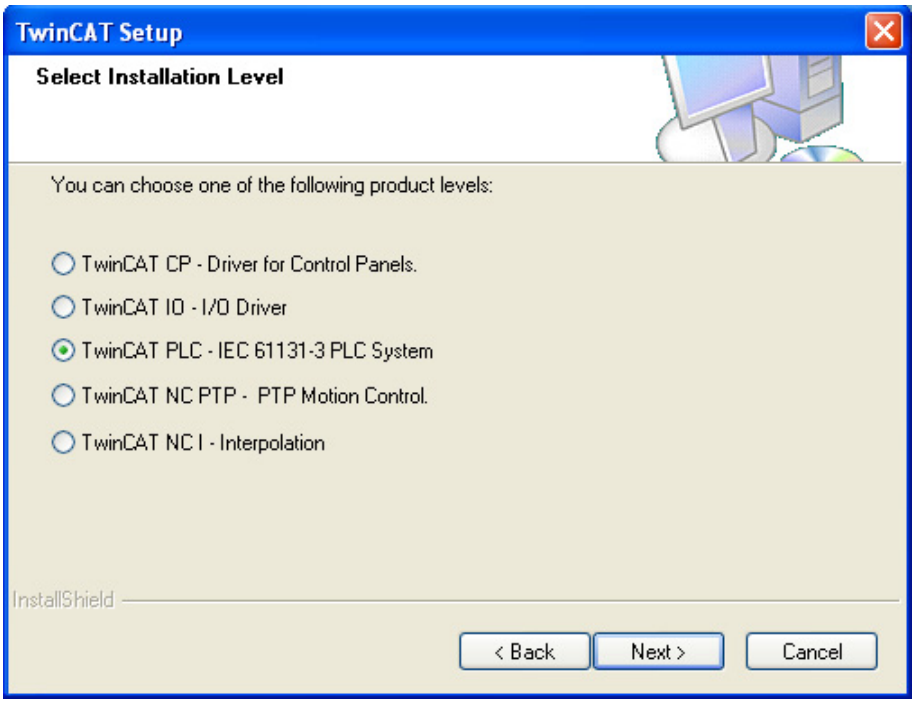
Picture 7

Input a name and the name of your company. Leave the Serial Number empty. This example installs the 30-day demo version.



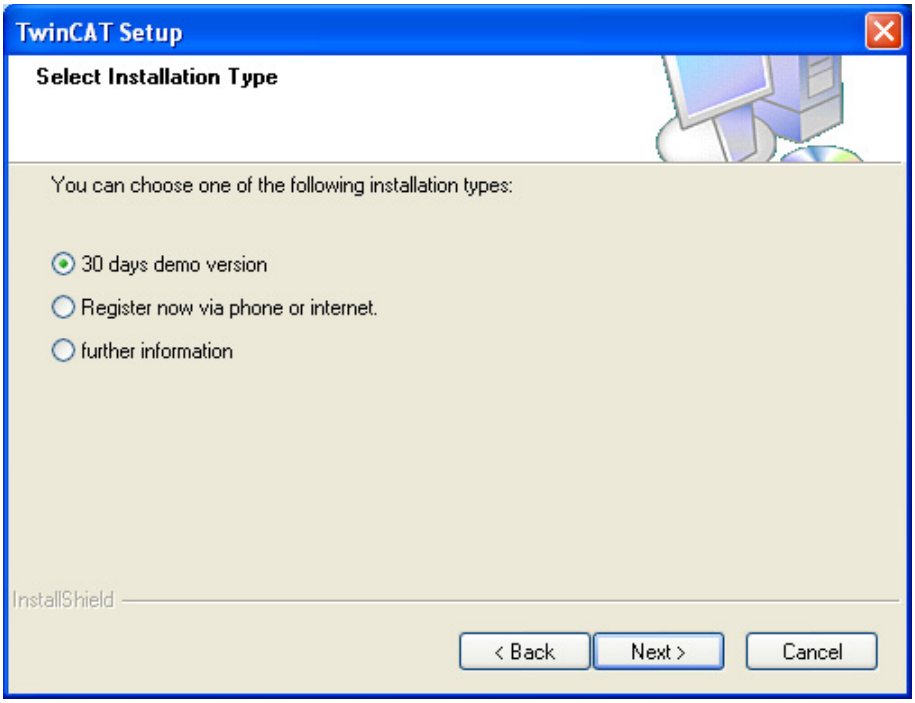
Picture 8

Select the Installation Level as shown in picture 9.



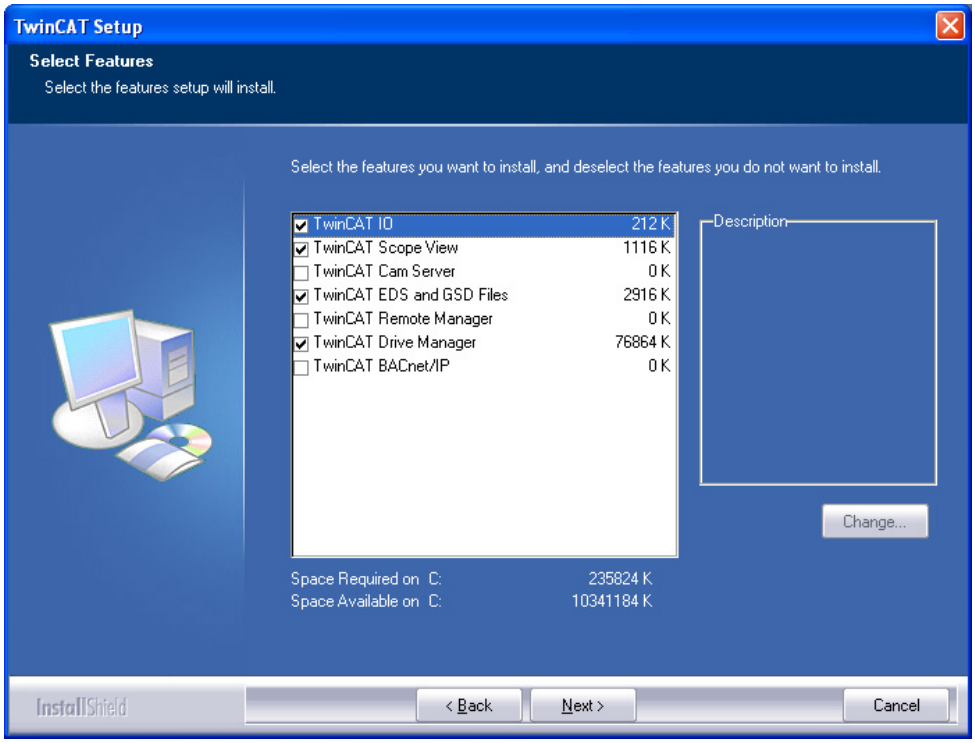
Picture 9

Select the 30-day demo version of the installation.



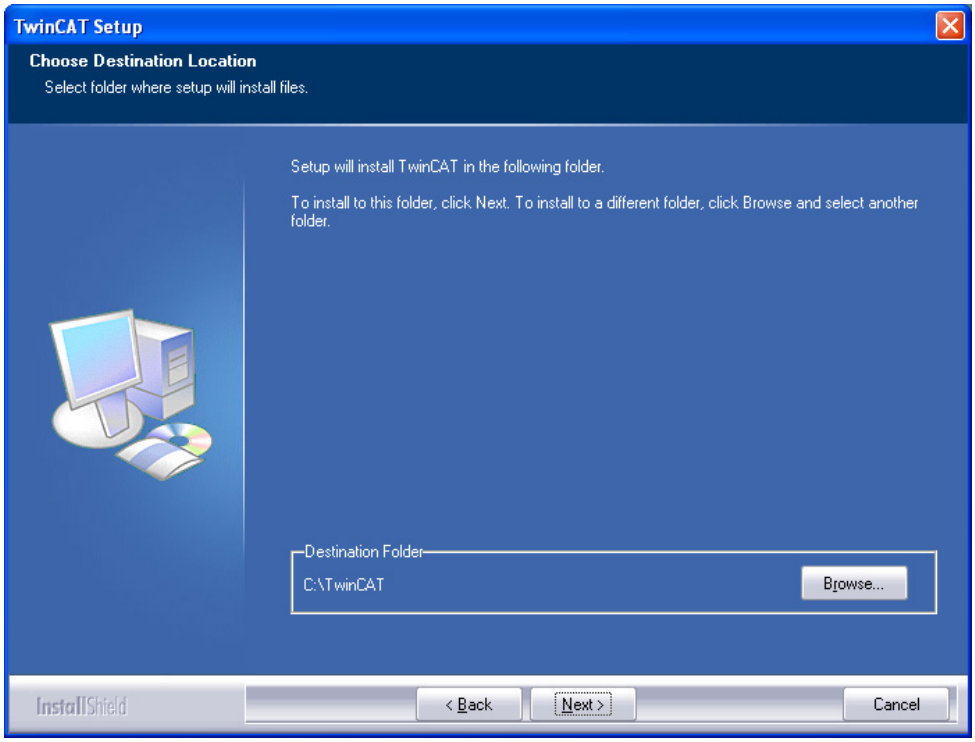
Picture 10

Select the components to be installed as shown in picture 11 hereafter.



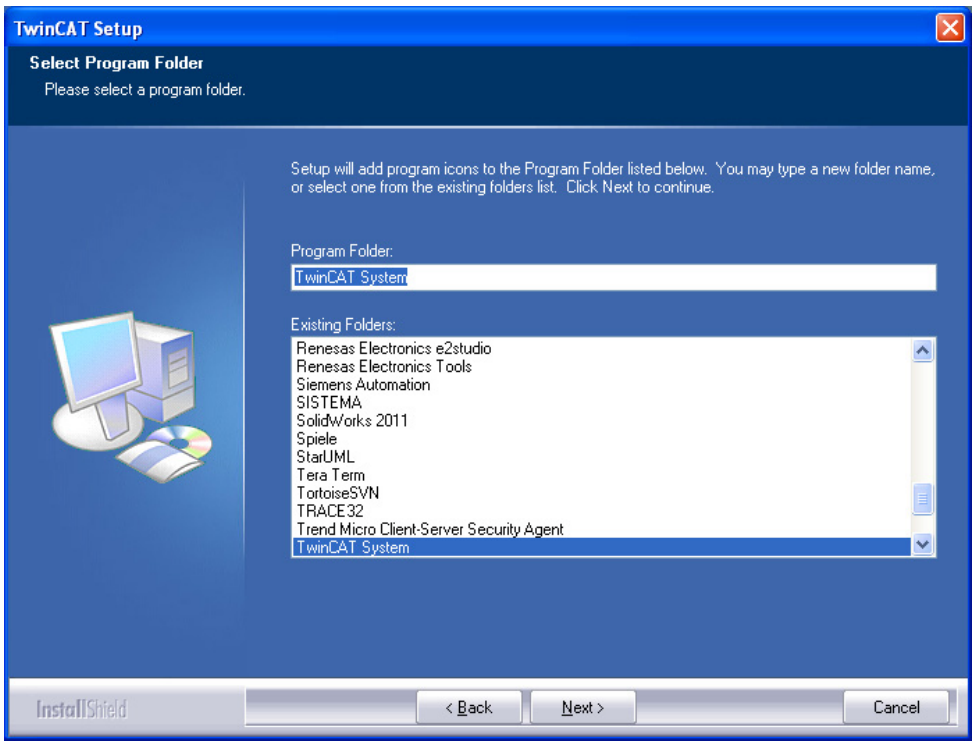
Picture 11

Select the installation folder as shown in picture 12.



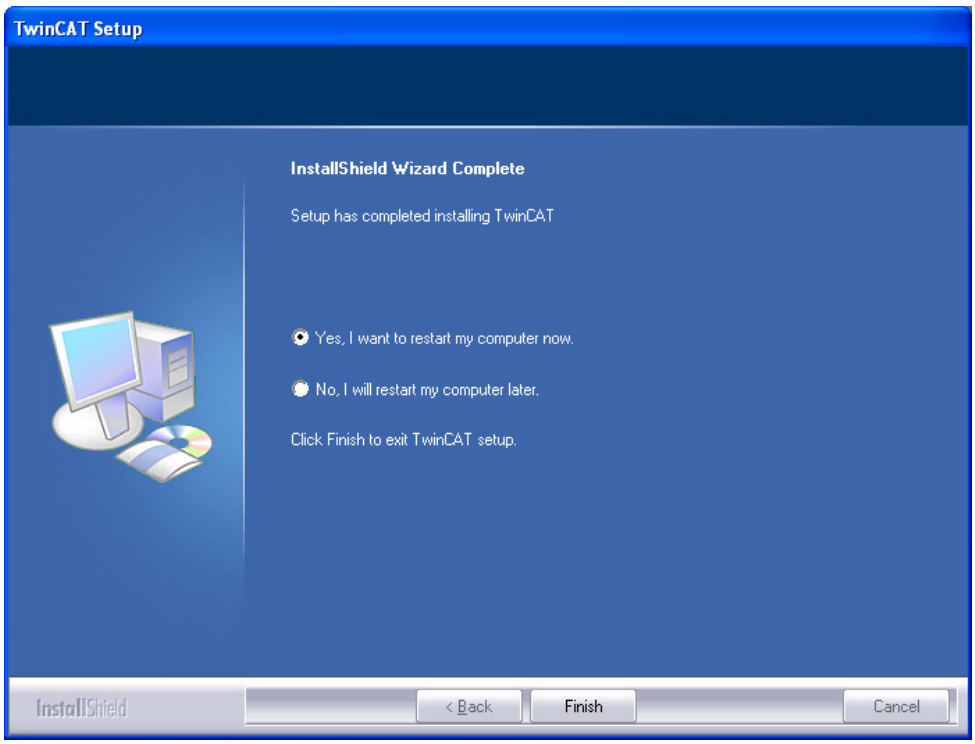
Picture 12

Select the program group name as shown in picture 13.



Picture13

Reboot your computer. This will make the TwinCAT installation usable.



Picture 14

Commissioning the encoder

The DVD supplied with the EtherCAT encoder contains the EtherCAT slave information file (ESI) in XML form. Its name is „TurckEtherCAT.xml“, and it must be installed in the TwinCAT installation folder.

Copying the ESI file

Once you have installed TwinCAT in C:\TwinCAT, you have to copy the file TurckEtherCAT.xml in the folder C:\TwinCAT\Io\EtherCAT\.

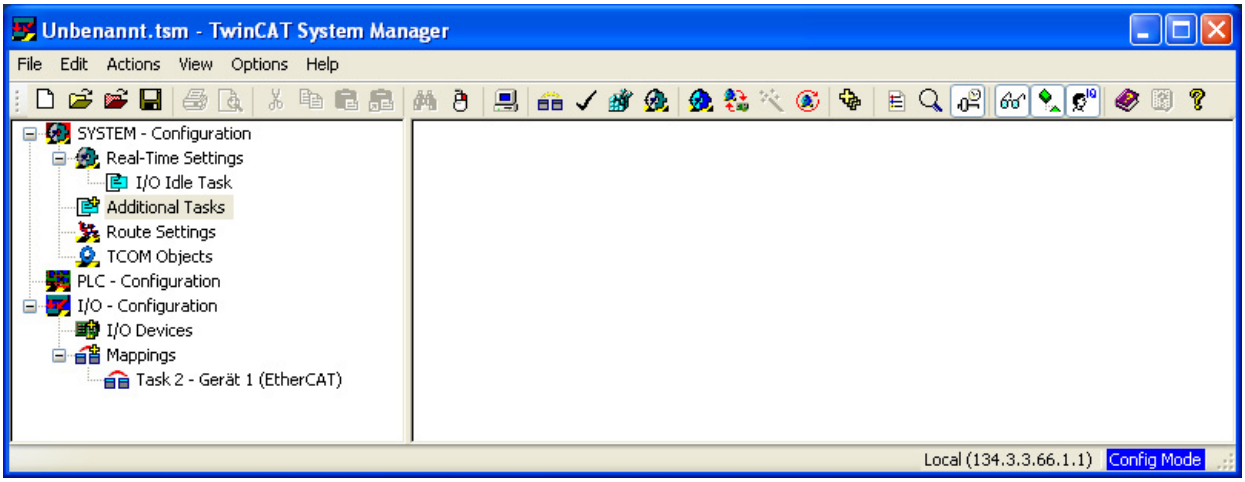
Configuration of the TwinCAT System Manager

Start the TwinCAT System Manager in the program group or in the Quick Start bar.

Picture 15

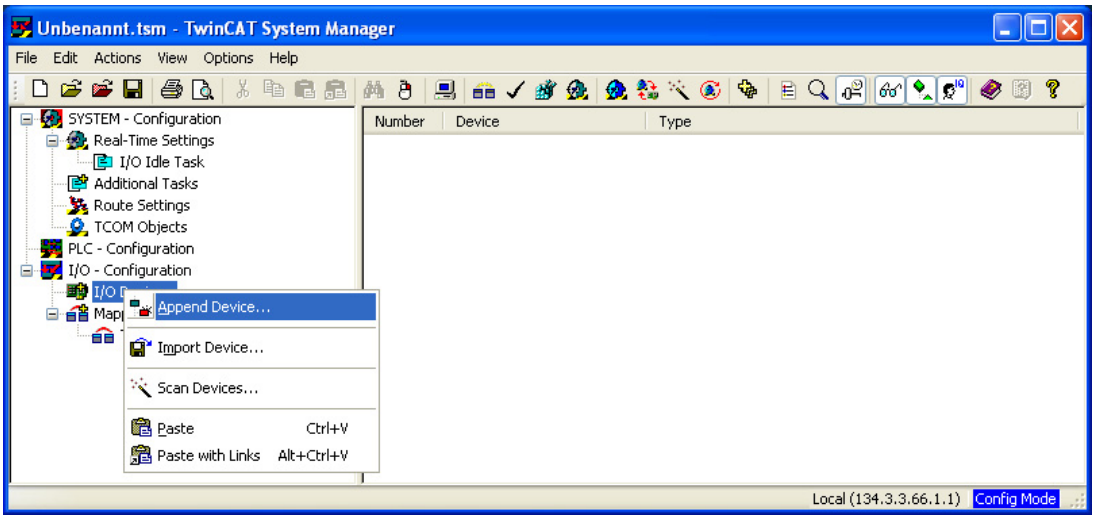
The first start requires more time, since the System Manager analyses all XML files in the installation folder. The progress can be followed on the progress bar in the left lower corner of the System Manager window.

The graphic front end of the TwinCAT-System Managers corresponds to picture 16 hereafter.



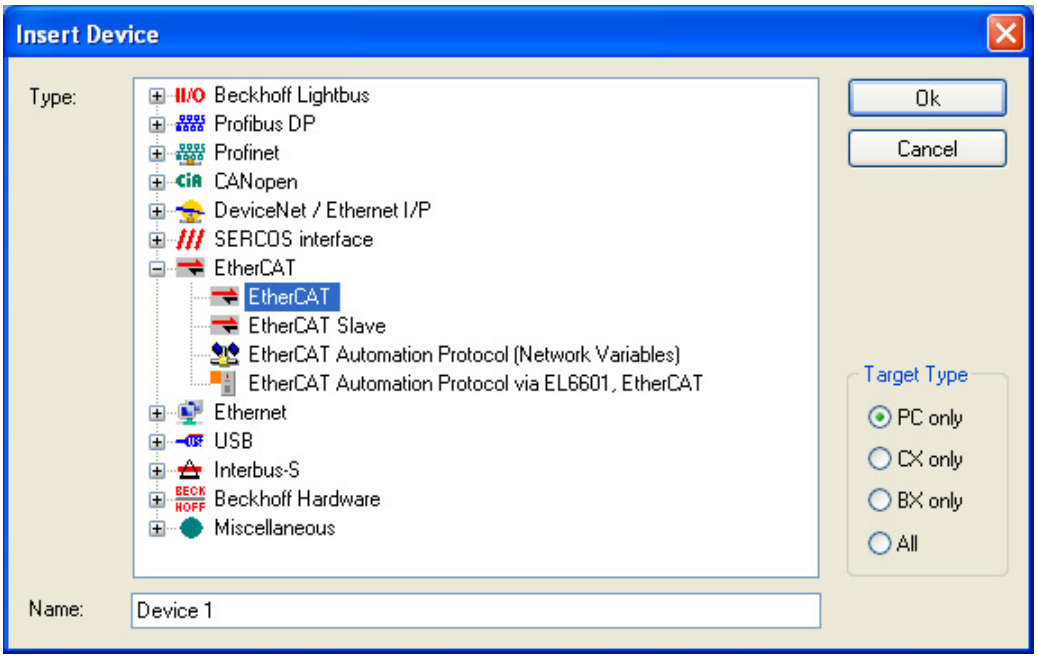
Picture 16

As shown in picture 17, click with the right mouse key the „I/O Devices“ line and select in the displayed menu the „Append Device...“ item.



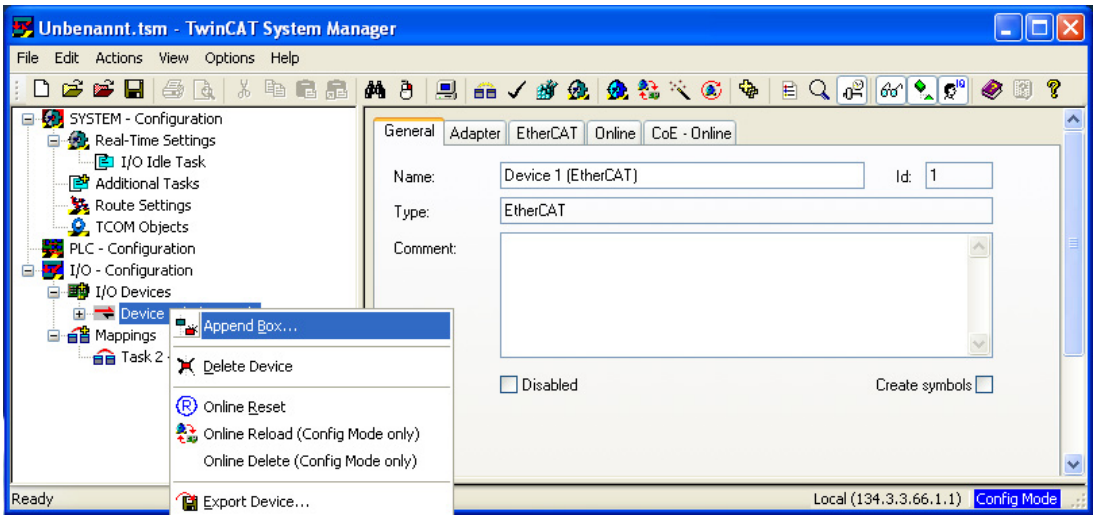
Picture 17

In the window that then opens, select menu „EtherCAT“, submenu „EtherCAT“, as shown in picture 18, and click on Ok.



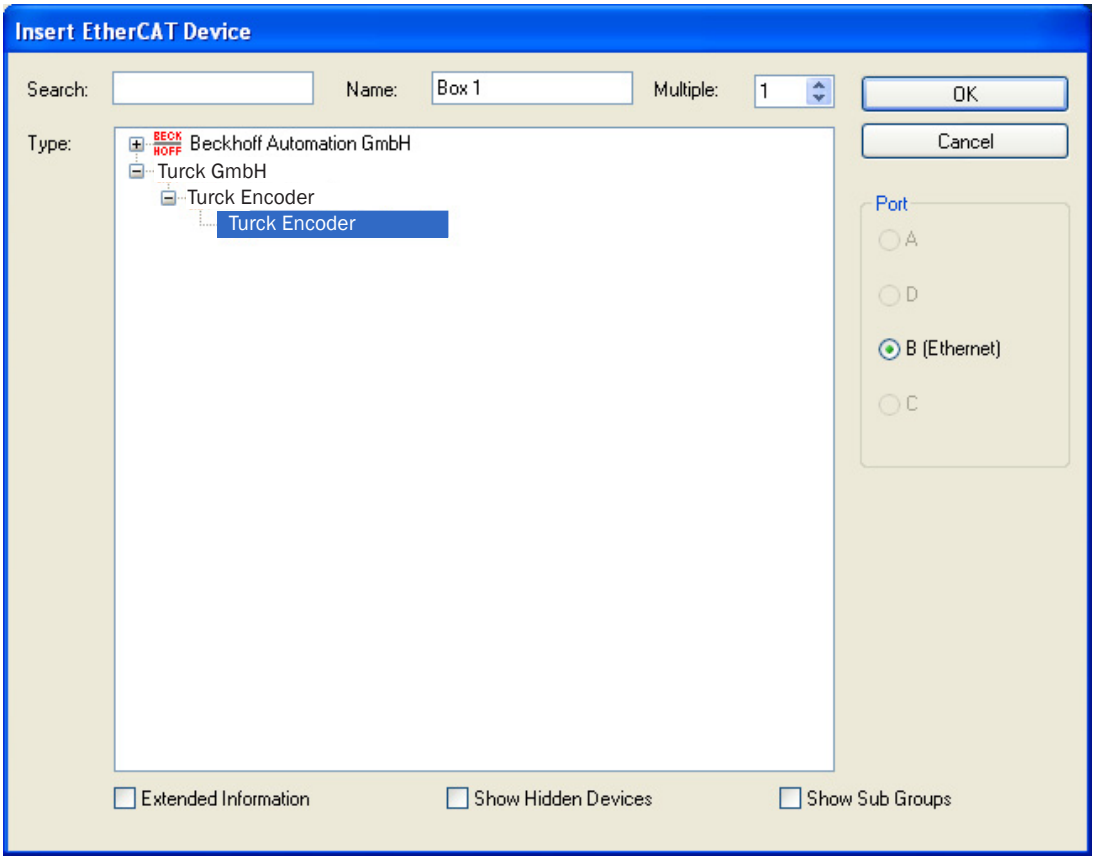
Picture 18

Click with the right mouse button on the newly displayed menu item „Device 1 (EtherCAT)“ and select the submenu „Append Box...“.



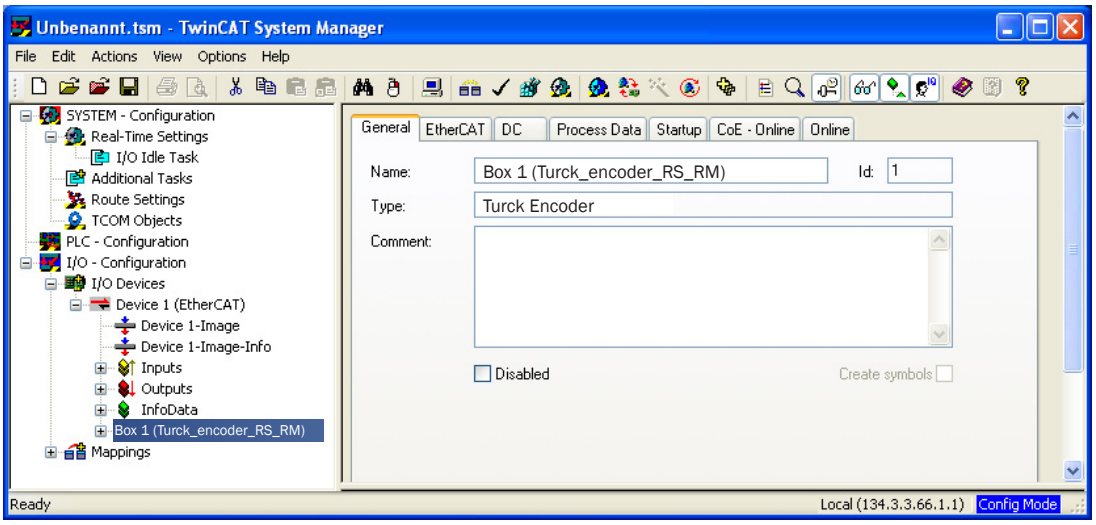
Picture 19

In the window that then opens, select Turck Encoder as shown in the picture below.



Picture 20

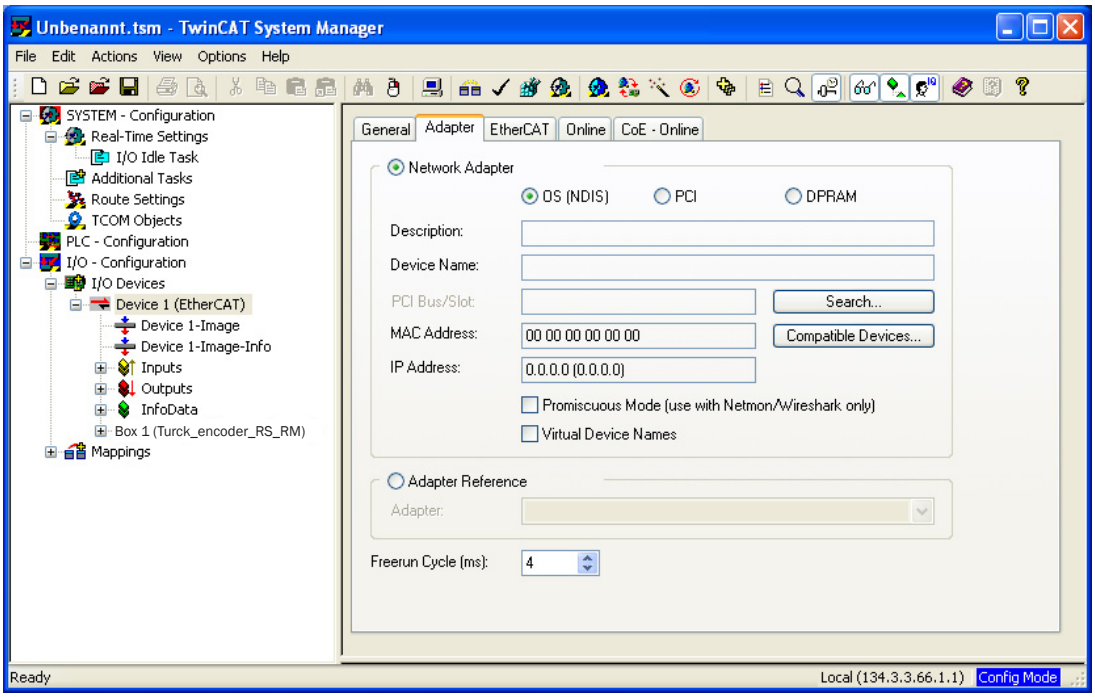
The System Manager window should now look as follows:



Picture 21

Configuration of the network board

As shown in picture 22, select the menu „Device 1 (EtherCAT)“ and select, in the right-hand area, the tab „Adapter“.

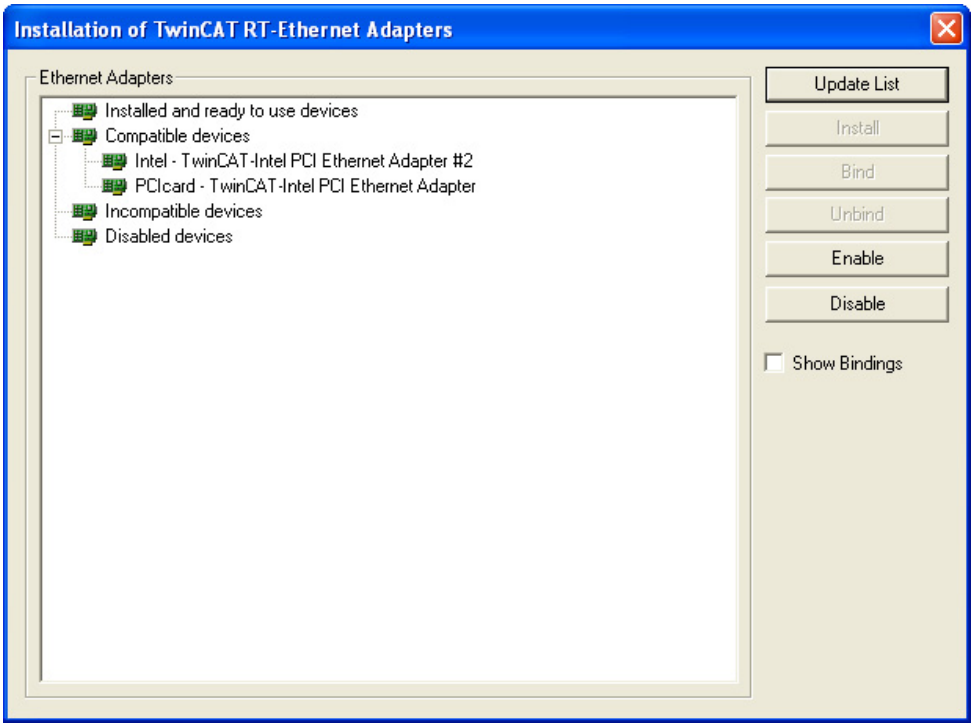


Picture 22

Click on the button „Compatible Devices...“, which opens the following window (picture 23). It shows on the first line the adapters that are installed and ready-to-use for TwinCAT („Installed and ready to use devices“). In this example, no ready-to-use adapter is available yet.

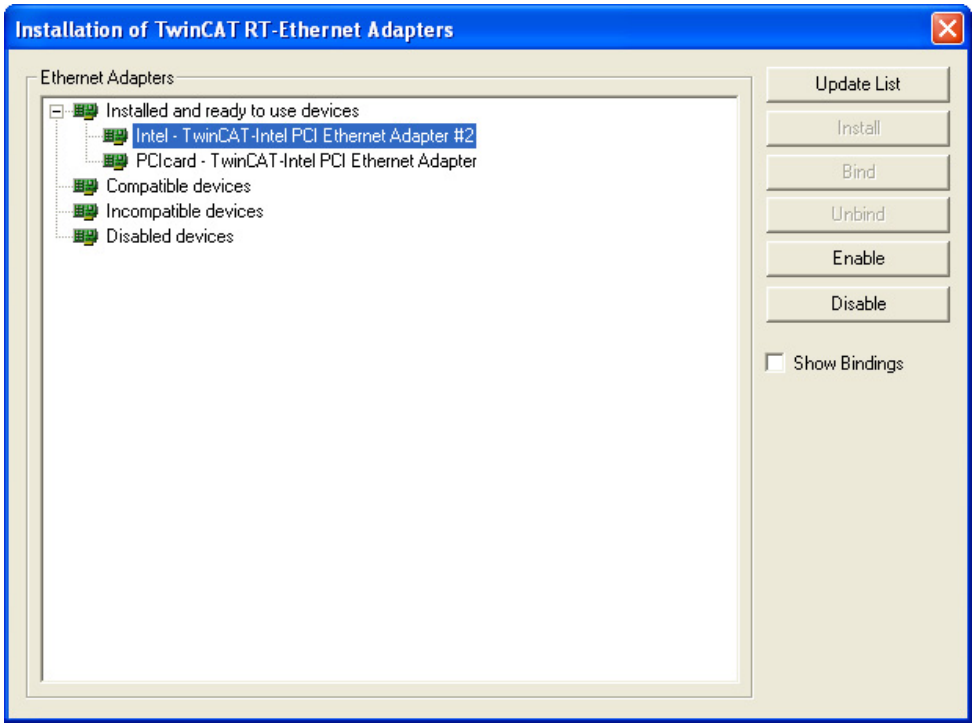
The second line, with its sub-items shows all TwinCAT-compatible adapters of the PC, from which one or several may be selected for the installation of the TwinCAT real time extension.

The third line shows all incompatible adapters. These adapters are not suitable for use with TwinCAT.
The fourth line shows all adapters that have already been operated successfully, but have been disabled.



Picture 23

Select at least one adapter from the category of the compatible adapters and click on the „Install“ button.
The adapters are now displayed as sub-items of the installed and ready-to-use adapters. See picture 24.

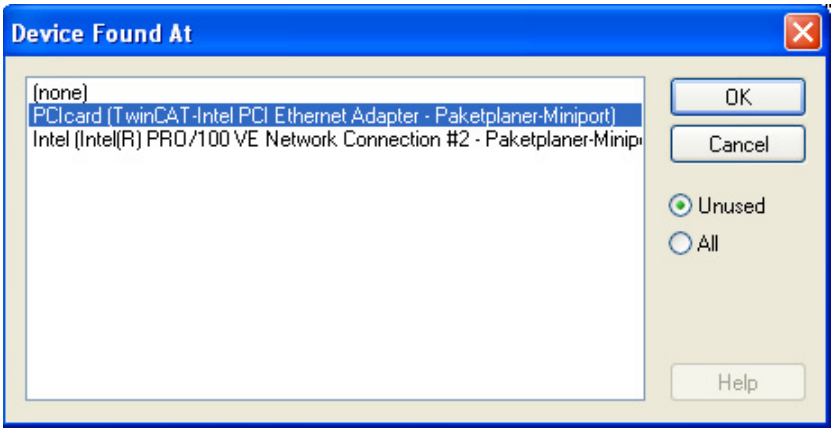


Picture 24

If, due to an update of the TwinCAT Software, there is already a disabled adapter, select it and click on „Enable“. This adapter then also appears in the category of the installed and ready-to-use adapters.

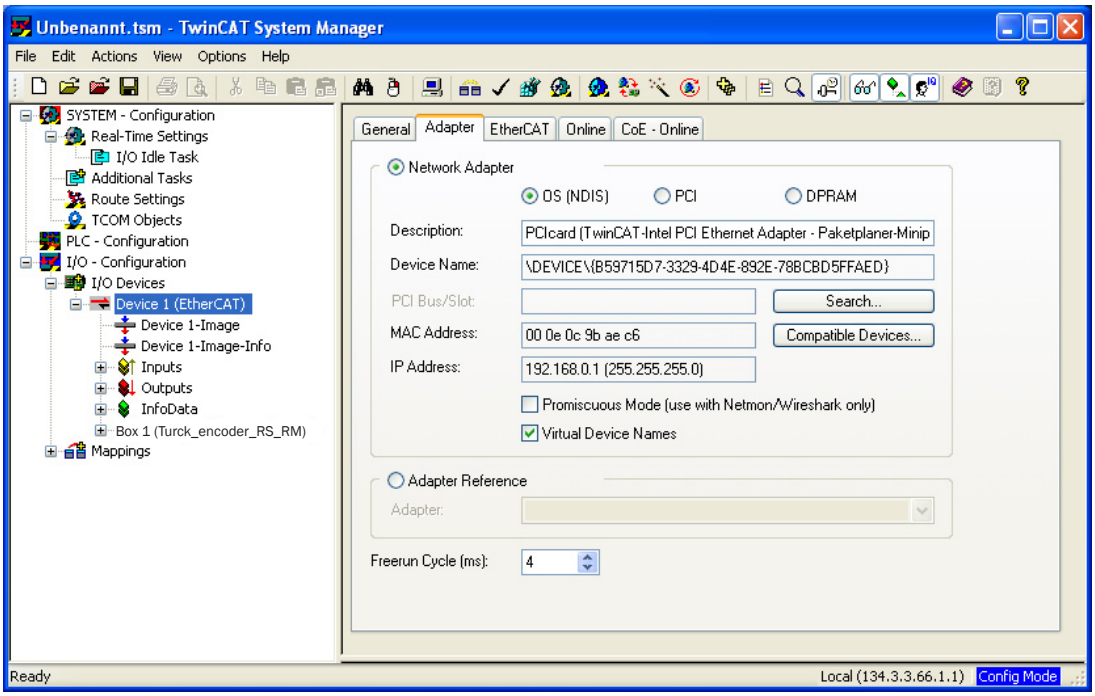
Now close the window and click on the „Search...“ button. A selection window opens, allowing selecting an adapter for the future TwinCAT communication with the encoder. In the present example, the „PCIcard“ adapter has been selected (picture 25).

Important!
If no input is displayed in the dialogue box of picture 25, even though the installation of a network card according to picture 24 was completed successfully, this network card is not suitable for operation with TwinCAT.



Picture 25

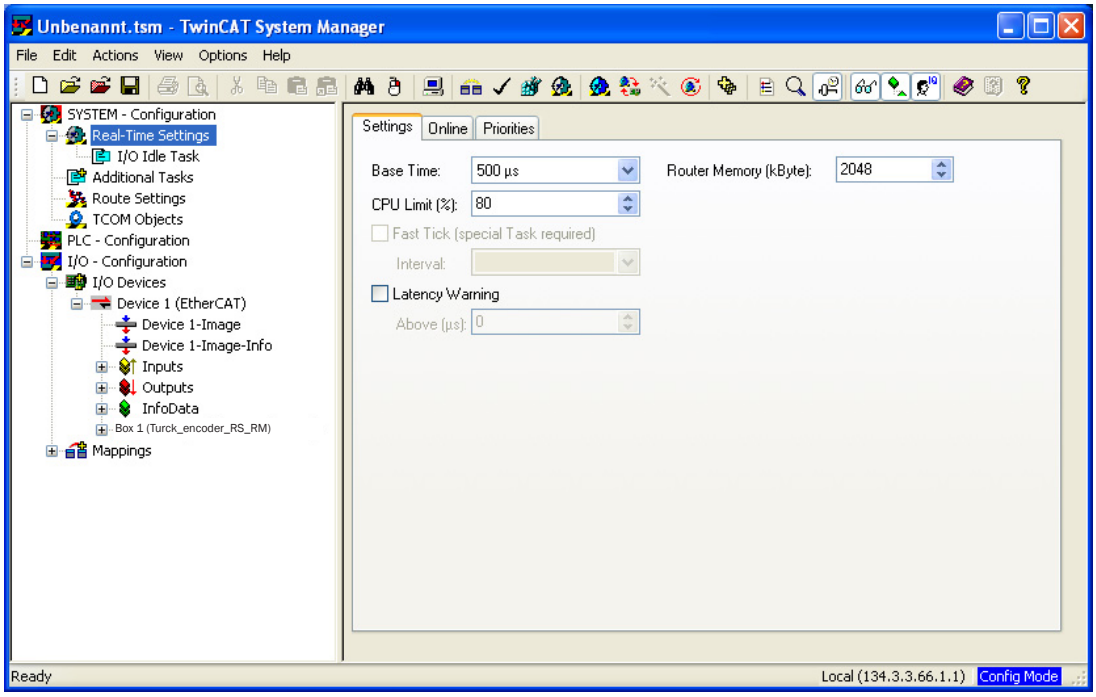
The „Adapter“ tab finally corresponds to this selection of picture 26



Picture 26

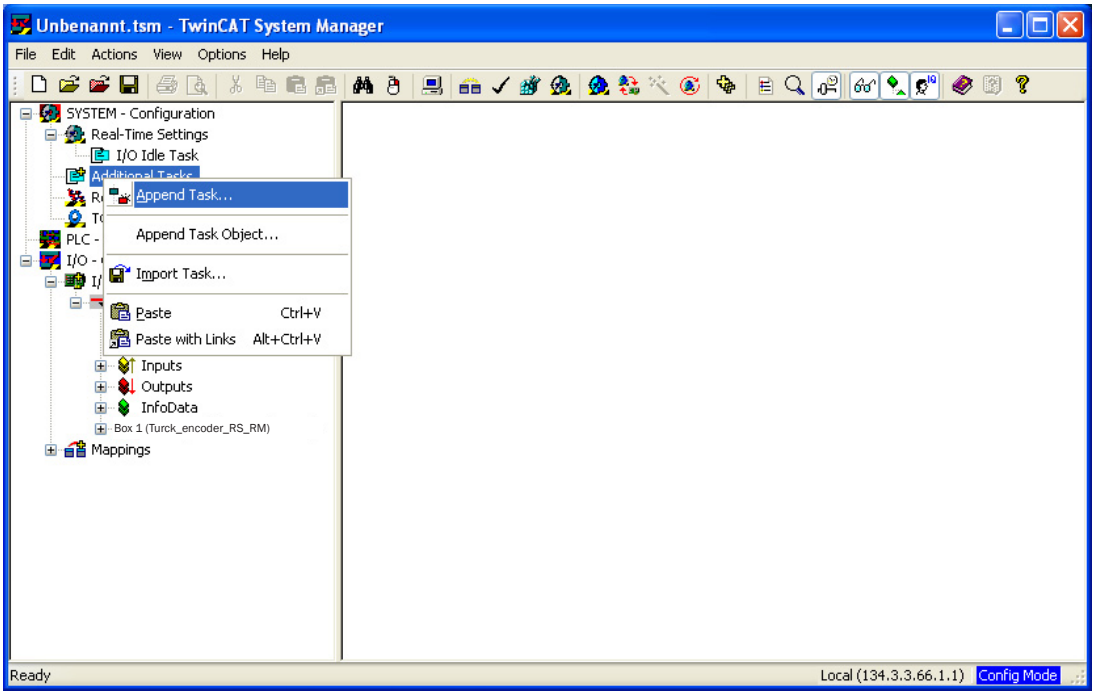
Configuration of the DC operating mode (Distributed Clocks)

In the System Configuration, select the submenu „Real-Time Settings“ as shown in the picture below and set the „Base time“ to 500µs.



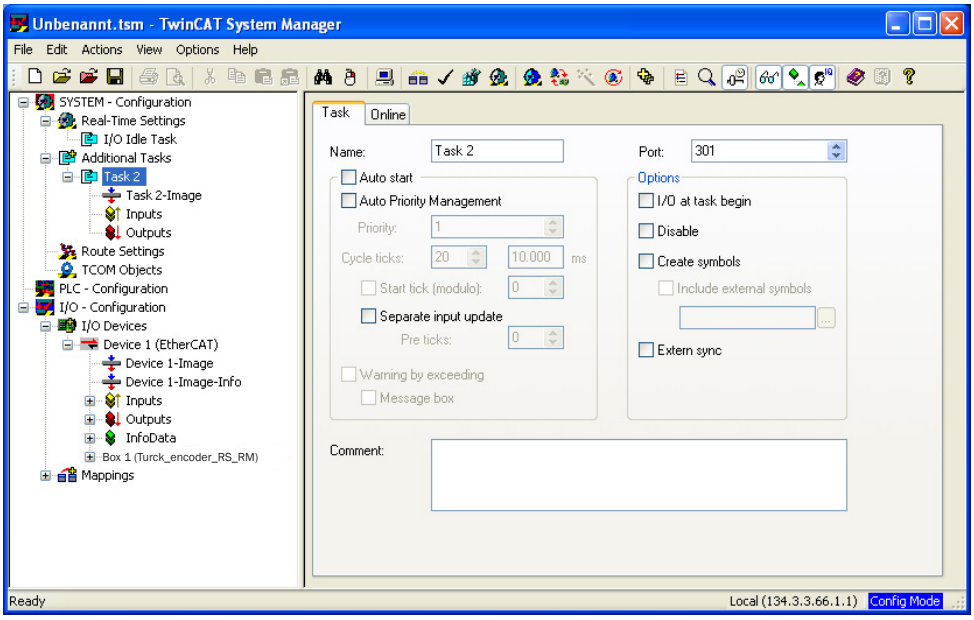
Picture 27

As shown in picture 28, select now the menu „Additional Tasks“ with the right mouse button and select the submenu item „Append Task...“. Click OK in the dialogue box that is then displayed.



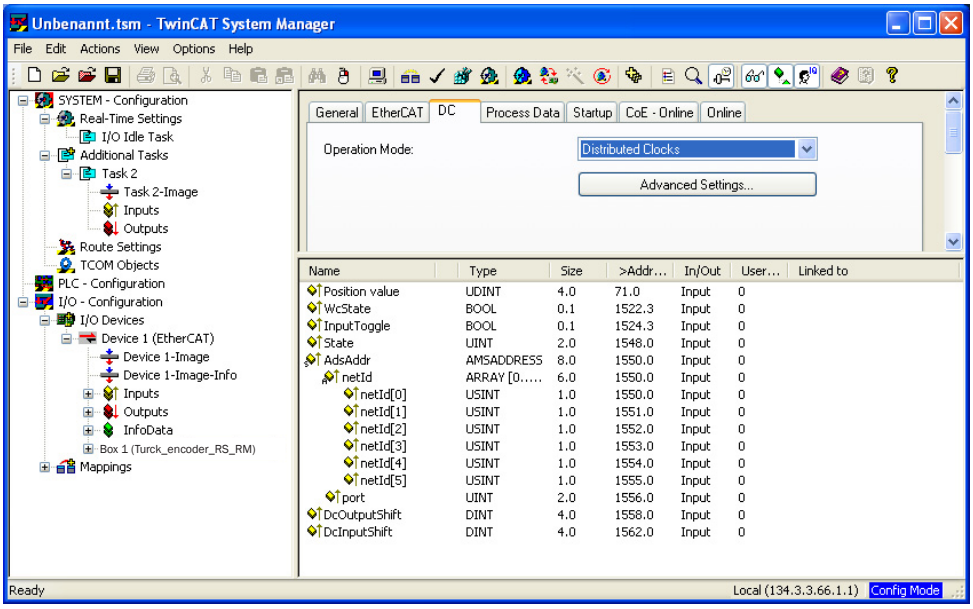
Picture 28

In the dialogue window of Task 2, select now all settings as shown in picture 29.



Picture 29

Select now the menu item „Box 1 (Turck_encoder_RS_RM)“ and select the „DC“ tab. Select the Operation Mode Distributed Clocks (picture 30).



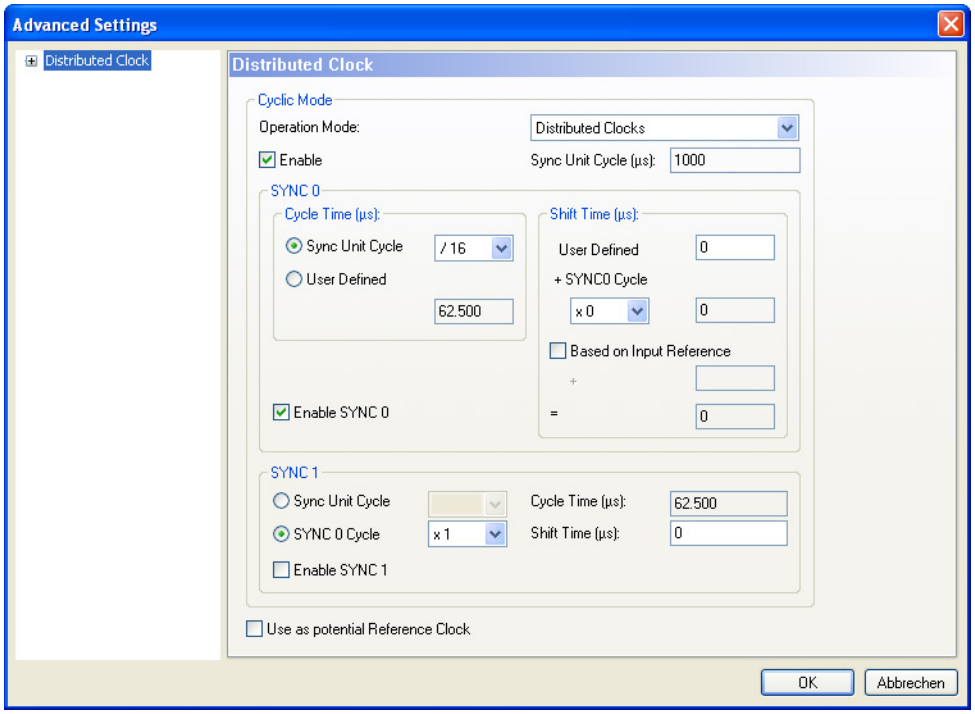
Picture 30

Click the „Advanced settings...“ button and check whether the settings correspond to picture 31. In particular the SYNC 0 Cycle Time should be 62.500 µs or more. The cycle time of 62.5 µs may only be used if no other process data than the position is to be transmitted. In other words, with the 62.5 µs cycle time, only one of Object 0x6004 or 0x2004 may be mapped, but not both.

Important!

The duration of the transmission, and thus the duration of the DC cycle, depends on the number of mapped bytes. If the DC cycle time is too short, there will be a communication break. Therefore, be sure to observe Annex „DC cycle times“.

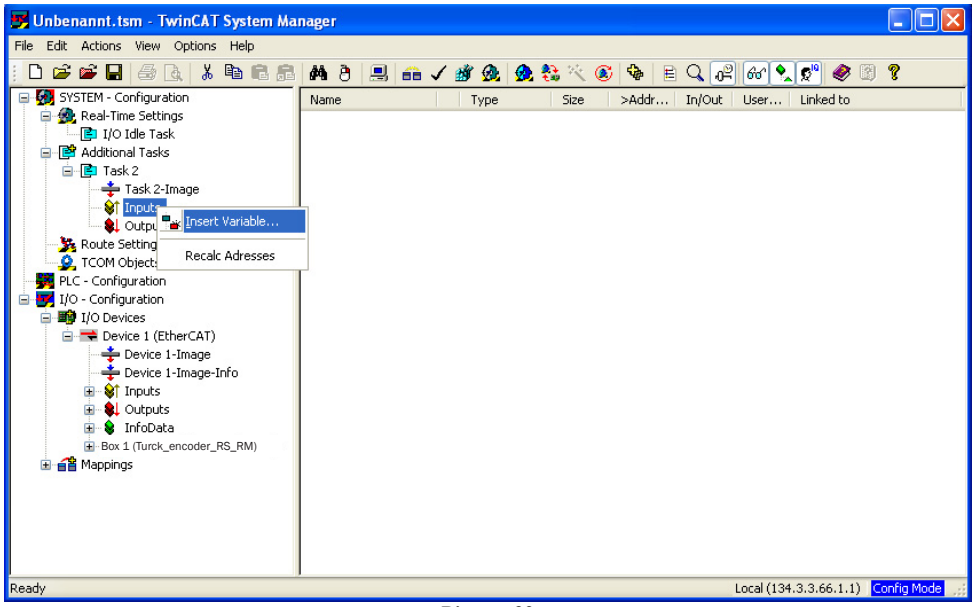
As shown in picture 30, the System Manager is in the Configuration mode. This is indicated by the „Config Mode“ message in the blue field in the lower right corner.



Picture 31

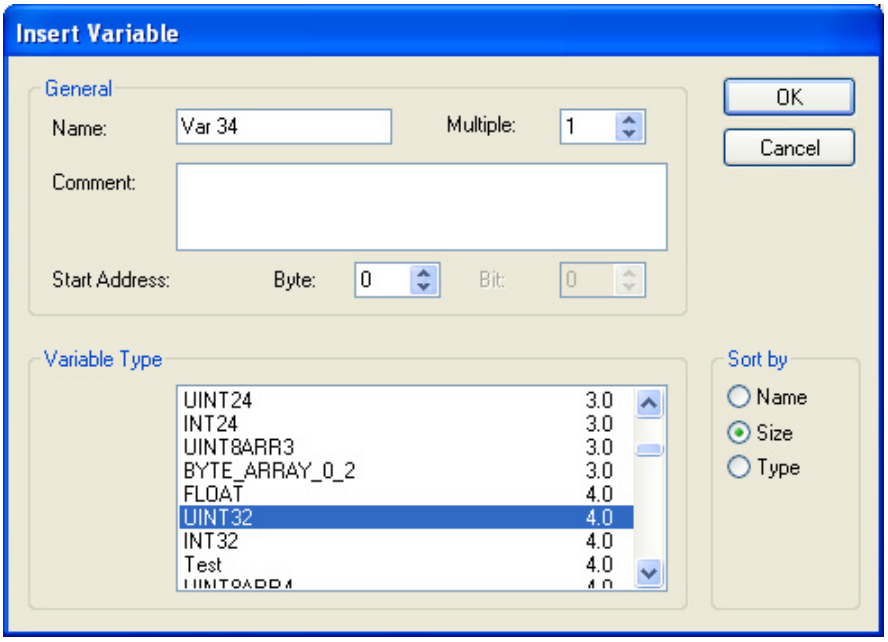
Representation of the process data in variables in TwinCAT

As shown in picture 32, select with the right mouse button the „Inputs“ item and select the submenu item „Insert Variable...“.



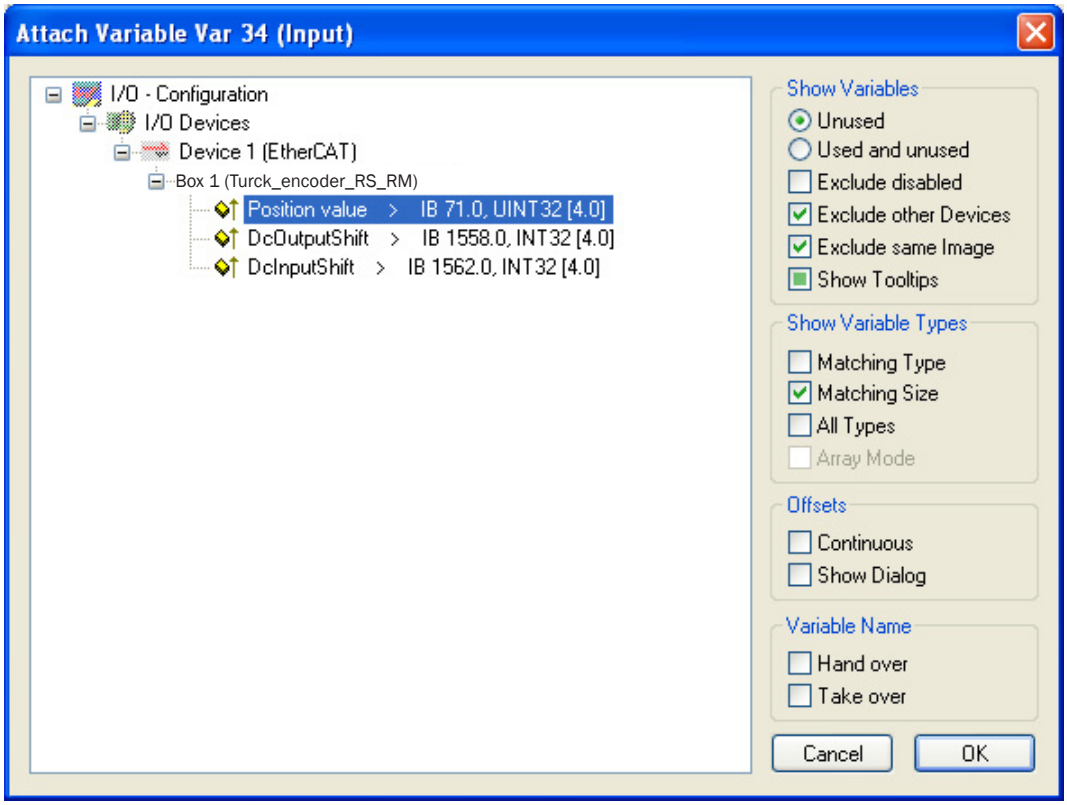
Picture 32

In the dialogue window that is displayed then, see following picture, select a variable of the UINT32 type. The value of the encoder position will be reproduced in this variable. If necessary, input a suitable comment at the location provided to that purpose and click on OK



Picture 33

Select now the variable „Var 34“ and click on the „Connect...“ button.
Select now the „Position value“ of the encoder and click on the OK button (picture 34).



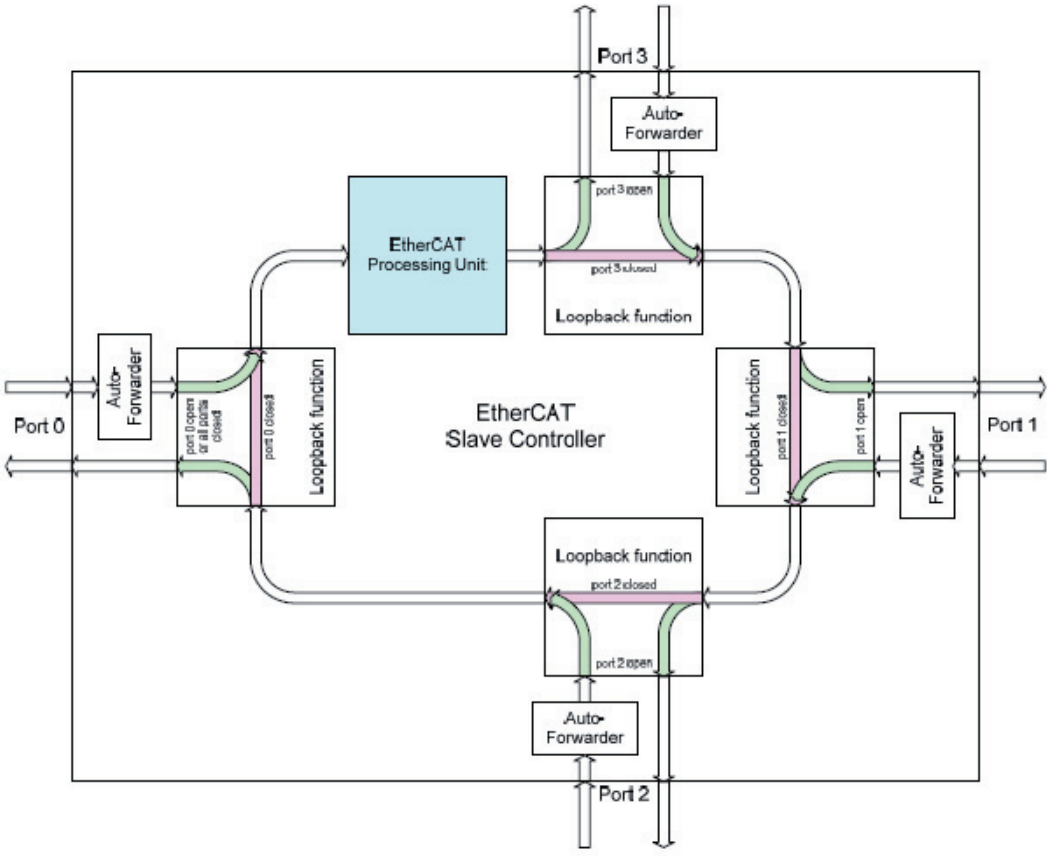
Picture 34

7 Setting up a communication between the encoder and TwinCAT

Now the encoder is to be put in communication with the PC via the network card that has been configured for TwinCAT.

Caution!

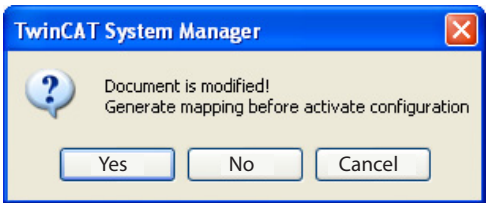
If you selected a star topology and thus use only one port of the encoder, imperatively connect the encoder via Port IN. This is an indispensable prerequisite for the good operation of an EtherCAT device. As shown in the following picture, Port IN corresponds to Port0 of the ET1100, which represents the EtherCAT slave controller of the encoder. Port OUT of the encoder corresponds to Port1 of the ET1100.



Switch the supply voltage of the encoder on.

In case of a trouble-free start-up of the encoder firmware, the red LED only lights up for a short period. Then the yellow LED of the port connecting the encoder to TwinCAT / to the control switches to permanently on.

Activate now the DC mode of TwinCAT with the „Actions“ menu, submenu „Activate Configuration...“.
Answer the following dialogues with „Yes“.



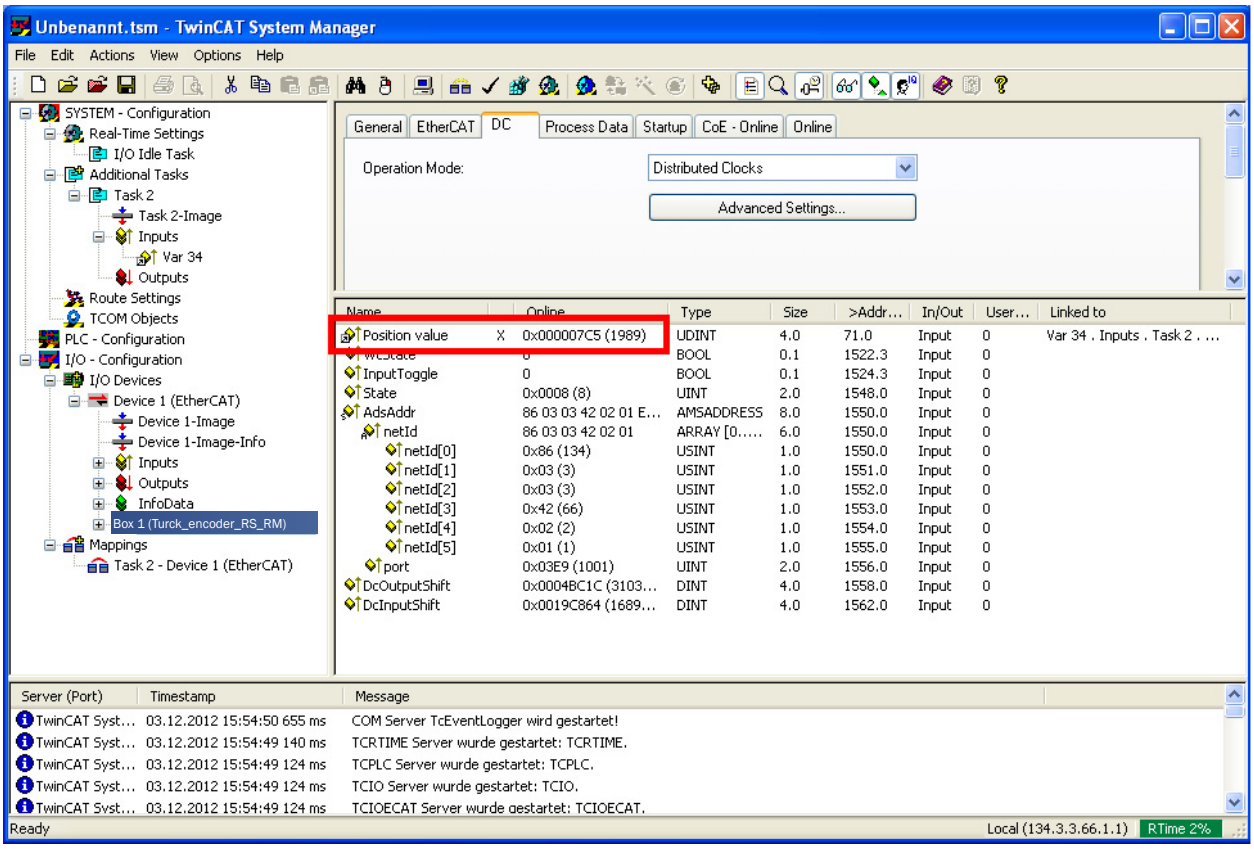
Picture 35

And confirm the two following dialogues with OK.

Picture 36

Picture 37

If all conditions for the Operational Mode are met, TwinCAT switches to the Real-Time display and shows the position value. This value is highlighted with a red rectangle in the next picture 38. In the present example, the position has the value 4317.



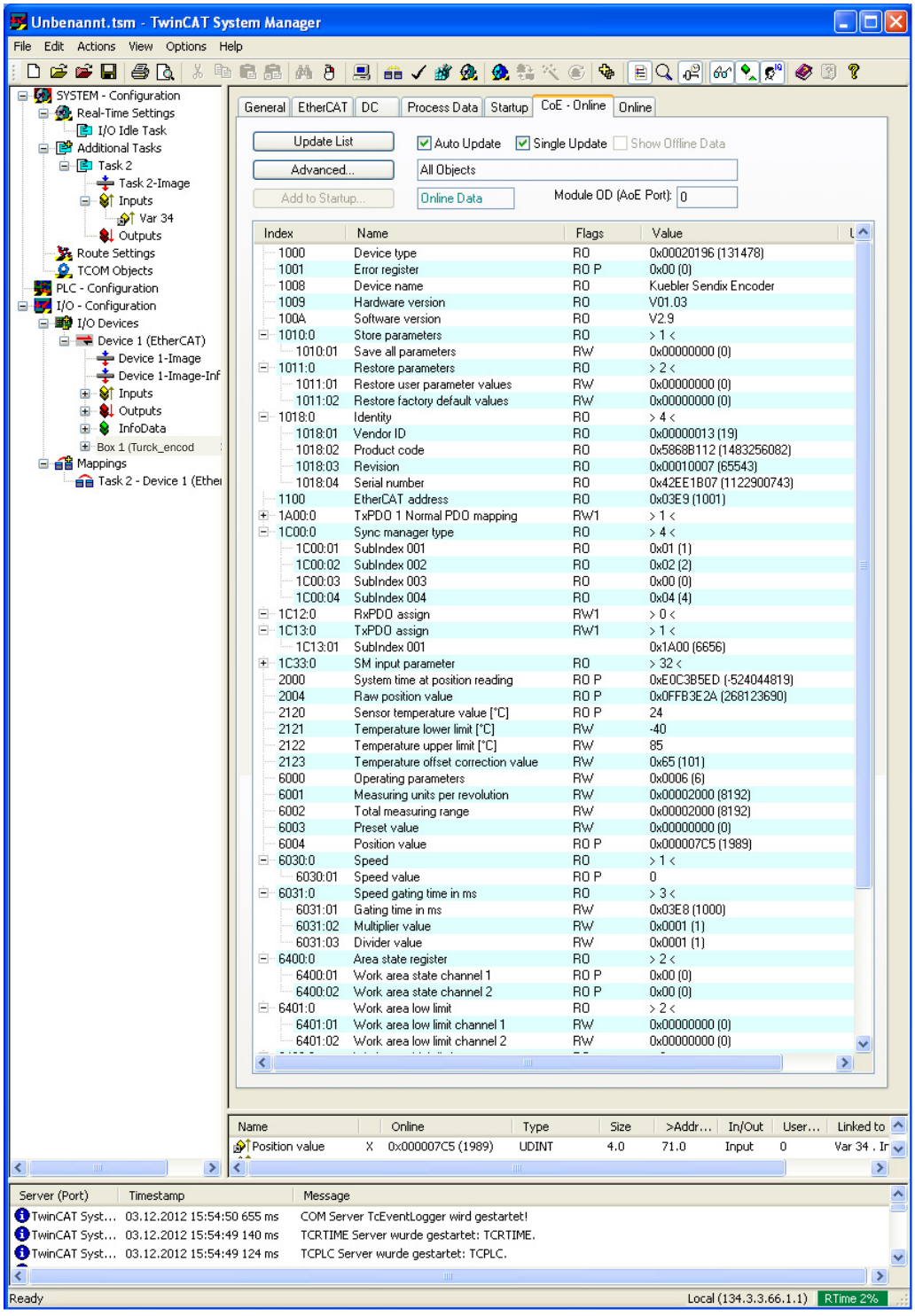
Picture 38

The yellow LED of the associated ports is now blinking. The green RUN LED is permanently on.

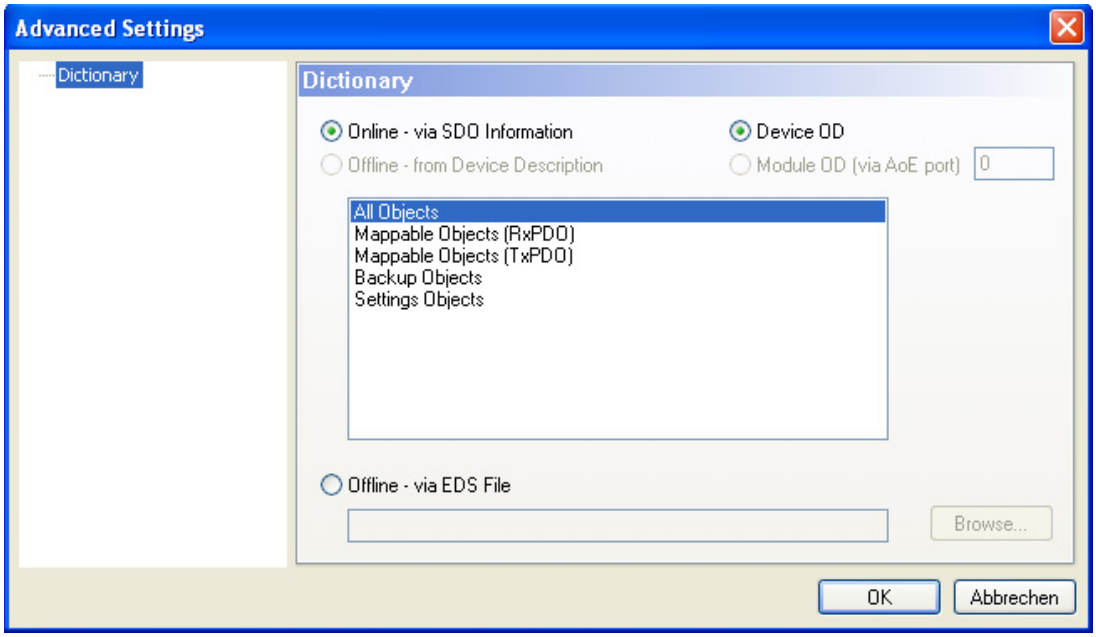
8 Encoder configuration

The following picture 41 shows an example of the SDO's and PDO's supported by the encoder. The objects are based on the CiA CANopen profile for encoders in Version 3.2.10 of 18 February 2011 and will be explained subsequently.

The display shown in picture 39 is obtained reading the SDO/PDO information of the encoder with the „Advanced..." button, see picture 40.



Picture 39



Picture 40

Object 1000_h : Device type (read-only)
Displays the following for the Turck encoders range
0x00010196 for Singleturn encoders or
0x00020196 for Multiturn encoders

Object 1001_h : Error register (read-only, mappable)
This is the Error Register of the encoder. If an error occurs, it will be displayed, in case of a temperature overshoot or undershoot, directly in bit 3 with the help of this register. Bit 0 (generic error) is also always set in this case. Therefore, the global error code is 0x09 in case of an unpermissible temperature over /undershoot. In addition, an EMERGENCY message is emitted with code 0x4200.

In case of a position error or a „Commissioning diagnostic" error, bit 0, „generic error", and Bit 5, „device profile specific error", are set in this register. Object 0x6503 indicates whether a position error or a „Commissioning diagnostic" error occurred. Refer to the description of Object 0x6503 in the following pages and to Reference [2].

Important!
Since reading the temperature is a not insignificant operation as concerns time, the temperature, in the case of the DC mode, is only read continuously from the ASIC if the temperatures belongs to the process data. In other words, if Object 0x2120 is mapped. When the DC mode is activated, but Object 0x2120 is not mapped, Object 0x2120 shows the correct temperature value immediately after switching on, but this value is not updated any more in the Operational status! Therefore, a possibly occurring temperature error will not be displayed in Object 1001.
In the case of the FreeRun mode, the temperature is updated with every bus cycle.

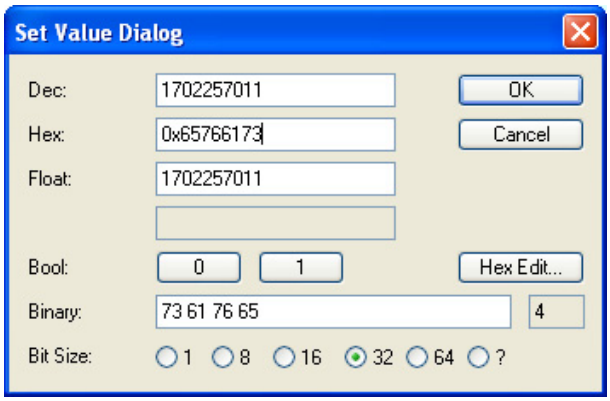
Object 1008_h : Device name (read-only)
Has the constant value „Turck Encoder".

Object 1009_h : Hardware version (read-only)
Has the constant value V01.03.

Object $100A_h$: Software version (read-only)
Has a constant value Va.b, a and b representing respectively the numerical values of the major and minor firmware version.

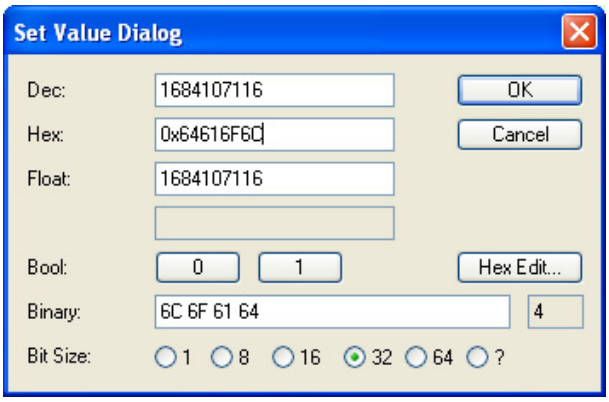
Object 1010_h : Store parameters (readWrite)
Sub-index 1 of this object allows the persistent storage of the following user parameters in the non-volatile memory of the encoder:
• 1A00
• 6000
• 6001
• 6002
• 6003
• 6031
• 6401
• 6402

Therefore, these values are available even after a reset, without requiring a new download from the control.
A double-click on the TwinCAT line „Save all parameters“ opens the following dialogue box. After the input of value 0x65766173, which represents the Hex signature of the word „save“ according to ISO 8859, the encoder stores the values.



Picture 41

Object 1011_h : Restore parameters (readWrite)
Sub-index 1 of this object allows activating all user parameters with the values stored in the non-volatile memory. Which parameters belong to the user parameters is described in Object 1010. A double-click on the line „Restore all parameters“ opens a dialogue box as shown in picture 42. After the input of value 0x64616F6C, which represents the Hex signature of the word „load“ according to ISO 8859, all user parameters are replaced with those stored in the non-volatile memory.
To take the values over in TwinCAT, the configuration must be activated.



Picture 42

Sub-index 2 is similar to sub-index 1: a double-click on the line „Restore factory default values“ activates the so-called Factory Values. In this case, the user parameters receive the values that have been determined at the time of the production of the encoder. Also in this case, the values are taken over in TwinCAT only after having activated the configuration.

Object 1018h : Identity (read-only)
Object 1018 has four indexes, which are all read-only:

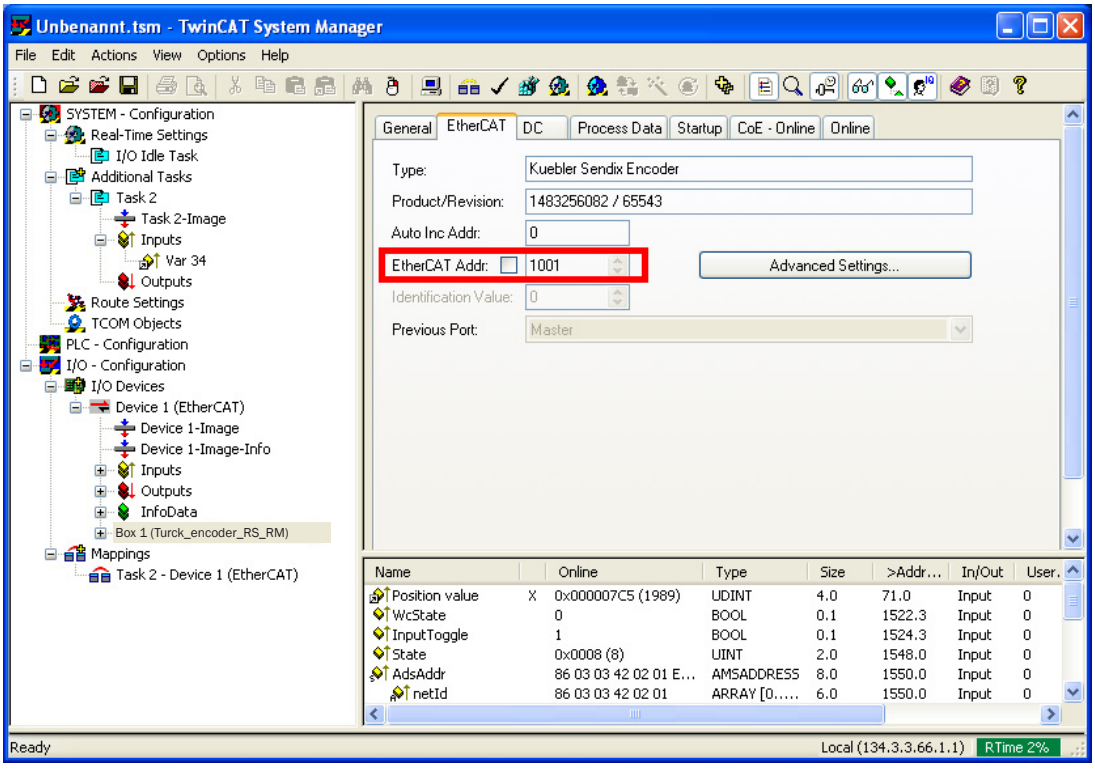
Vendor ID
This is the Vendor ID belonging to the Turck Group, which is registered with the EtherCAT Technology Group.

Product code
This value represents the Turck-specific order code of the encoder.

Revision
This value represents the revision of the EtherCAT encoder in general and is always 0x00010007.

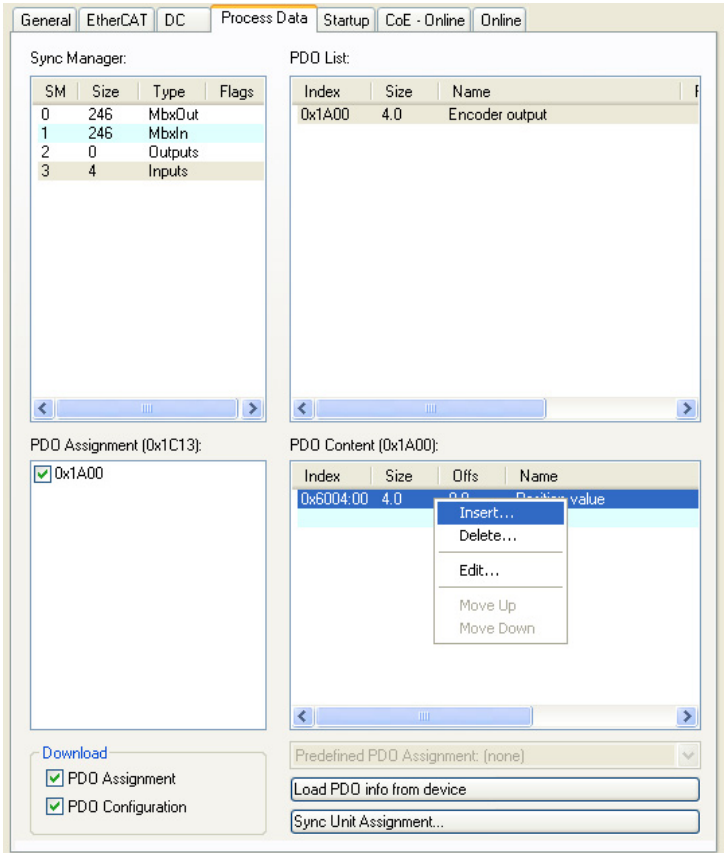
Serial Number
This value represents the unique serial number of the encoder. Each serial number applies to only one encoder.

Object 1100h : EtherCAT address (read-only)
Object 1100 indicates the EtherCAT address of the encoder. In the specific case of TwinCAT, this is the address that has been set in the dialogue window of picture 43.



Picture 43

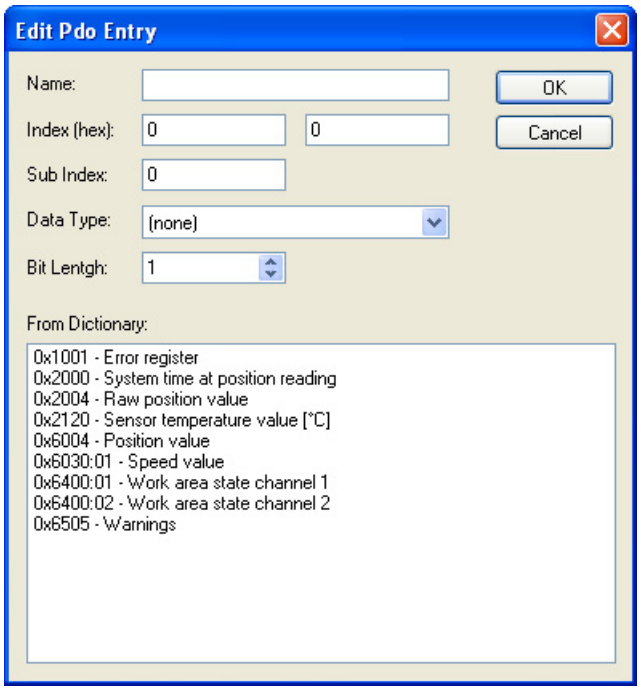
Object 1A00h : TxPDO 1 Normal PDO mapping (read-write)
This object allows selecting the data that is to be transmitted during runtime as process data. This data can be for example input in the „Process Data“ tab of the TwinCAT manager, as shown in picture 44.
To that purpose, click on the last item of the list with the right mouse button and select menu item „Insert...“. This opens a dialogue window, see picture 45, that allows selecting the single objects and placing them in the desired order.



Picture 44

Important!

For the DC mode, observe Annex „DC cycle times". The cycle time to be set increases with every additional process data byte. Independently of the operating mode of the encoder, DC or FreeRun, a maximum of seven objects may be inserted for mapping.



Picture 45

Object $1C00_h$: Sync Manager type (read-only)
Object 1C00 indicates the assignment of the type to the respective Sync Manager. According to [3], the type assignment for the Sync Manager is selected as follows:

Sync Manager 0: 1 Mailbox receive (Master to Slave)
Sync Manager 1: 2 Mailbox send (Slave to Master)
Sync Manager 2: 0 Disabled since the encoder has no output process data
Sync Manager 3: 4 Input process data (Slave to Master)

Object $1C12_h$: RxPDO assignment (read-only)
Since the encoder has no output process data, there is no assignment to a RxPDO object.

Object $1C13_h$: TxPDO assignment (read-only)
Assignment of the encoder process data takes place through Object 0x1A00.

Object $1C33_h$: SM 3 input parameter (read-only)
Object 1C33 has only Read-Only sub-indexes. This information is only for information purposes. It allows in particular:
• reading the synchronisation type in DC mode in sub-index 1. E.g. 2 represents „DC SYNC0 synchronized with AL Event".
• checking the cycle time in sub-index 2.
• reading the minimum cycle time in sub-index 5.

Object 2000_h:System time at position reading (read-only, mappable)
This object represents the system time at the moment when the position is generated in the encoder.

Object 2004_h : Raw position value (read-only, mappable)
This object allows evaluating the raw position data. The raw data depends only on the physical resolution of the sensor; it is independent of the scaling operations.

Object 2120_h : Sensor temperature value (read-only, mappable)
The sensor includes an ASIC whose component is a temperature sensor. It allows displaying the internal temperature of the encoder sensor. Object 0x2120 indicates the temperature in °C.

Important!
Since reading the temperature is a significant operation as concerns time, the temperature, in the case of the DC mode, is only read continuously from the ASIC and checked for exceeding the allowed range if the temperature belongs to the process data. In other words, if Object 0x2120 is mapped.
When the DC mode is activated, but Object 0x2120 is not mapped, Object 0x2120 shows the correct temperature value immediately after switching on, but this value is not updated any more in the Operational status, and it is not checked for exceeding the allowed range!
In the case of the FreeRun mode, the temperature is updated and checked for exceeding the allowed range with every bus cycle.

Object 2121_h : Temperature lower limit (read-write)
The sensor includes an ASIC whose component is a temperature sensor. It allows displaying the internal temperature of the encoder sensor. This object allows setting the lower temperature limit; an alarm is triggered if it is undershot. This alarm is signalled through Object 1001 (Error Register) and by a corresponding Emergency Message. The value is given in °C. The values allowed for this object are in the range of -45 °C to +90°C.

Important!
This value is set using the same ASIC interface that is used for reading the position. Therefore, an interruption of the bus cycle time of 250 ms will take place in DC mode. Ideally this value should be set in PreOperational mode.

Object 2122_h : Temperature upper limit (read-write)
The sensor includes an ASIC whose component is a temperature sensor. It allows displaying the internal temperature of the encoder sensor. This object allows setting the upper temperature limit; an alarm is triggered if it is overshot. This alarm is signalled through Object 1001 (Error Register) and by a corresponding Emergency Message. The values allowed for this object are in the range of -45 °C to +90°C.

Important!
This value is set using the same ASIC interface that is used for reading the position. Therefore, an interruption of the bus cycle time of 250 ms will take place in DC mode. Ideally this value should be set in PreOperational mode.

Object 2123_h : Temperature offset correction value (read-write)
This object allows offsetting the temperature sensor so that Object 2120 displays the value 64 (decimal) for a temperature of 0°C.

Important!
This value is set using the same ASIC interface that is used for reading the position. Therefore, an interruption of the bus cycle time of 250 ms will take place in DC mode.

Object 6000_h : Operating parameters (read-write)
In compliance with Reference [1], this object is defined as follows:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
nbr	su1	su0	-	-	-	-	-	-	-	-	-	-	sfc	cdc	cs

Important!
This value is set using the same ASIC interface that is used for reading the position. Therefore, an interruption of the bus cycle time of 250 ms will take place in DC mode.

Standard parameters:
cs = 0 = ˆ Code sequence = CW (clock wise). Reference view: on the encoder shaft side
cs = 1 = ˆ Code sequence = CCW (counter clock wise)

cdc = 0 = ˆ Commissioning diagnostic control disabled
cdc = 1 = ˆ Commissioning diagnostic control enabled

sfc = 0 = ˆScaling function control disabled
sfc = 1 = ˆ Scaling function control enabled

Manufacturer-specific parameters:
su0 und su1 = ˆ Speed Unit.

su1	su0	Speed unit
0	0	Rounds per minute (default)
0	1	Steps per 10 ms
1	0	Steps per 100 ms
1	1	Steps per second

nbr = 0 = ˆ Binary Ratio of TMR (Total Measuring Range) to MUR (Measuring Units per Revolution)
nbr = 1 = ˆ Non Binary Ratio of TMR to MUR

Relationship of TMR and MUR
TMR = Total Measuring Range
MUR = Measuring Units per Revolution

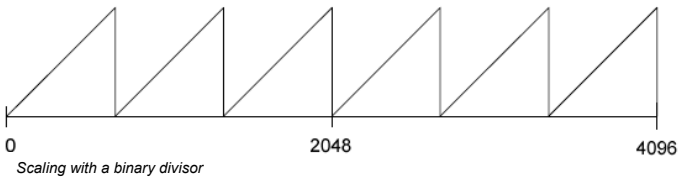
The single and multiturn units of the Turck encoders operate independently of each other. According to the resolution of the disc or to the interpolator, the bits used for the resolution can always be incremented or decremented only bit by bit. This also applies to the multiturn gear.
This fact affects the selectable values of MUR and TMR

The TMR value must be chosen so that either this value itself or a multiple of this value corresponds to the total measuring range and is a binary multiple of MUR.

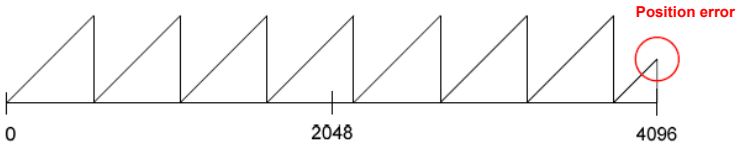
The total measuring range is defined as the product of the singleturn resolution g_ST (e.g. 65536 corresponding to 16 bits) and the multiturn resolution g_MT (e.g. 4096 corresponding to 12 bits).

If TMR is a binary multiple of MUR, the position value evolves without error at the end of the range, where an overflow from the maximum position to zero takes place, as shown in the picture below.

The second picture shows the case when TMR is not a binary multiple of MUR. An error appears at the end of the range.



Scaling with a binary divisor



A binary multiple is therefore defined as

TMR=MUR/2^k and in special cases **TMR=g_ST/2^k**
where k=0, 71, 72, …, 7`

If k is positive, TMR is a binary multiple of MUR

If k=0, TMR = MUR

If k is negative, MUR is a multiple of TMR. Figuratively speaking: the „saw teeth“ in the pictures above then exist k times within one revolution.

Therefore the following boundary conditions must be met for MUR und TMR:

- The MUR value is only accepted during the SDO download if it fulfils the following criterion:
  0 , MUR # g_ST and
  MUR must an addition be a multible of 2, thus: MUR = g_ST/2n
  where 0 # n # `

- The TMR value of an encoder without multiturn stage is only accepted during the SDO download if it fulfills the following criterion:
  0 , TMR # g_ST and
  TMR must be a multiple of 2, thus: TMR = MUR/2k and in special cases
  TMR = g_ST/2k
  where k = 0, 71, 72, …, 7`

- For an encoder with a multiturn stage, the following condition for the TMR value must be met so that it can be accepted for SDO download:
  0 , TMR # MUR*g_MT,
  in special cases 0 , TMR # g_ST*g_MT and
  TMR must be a multiple of 2, thus: TMR = MUR*g_MT/2k
  where k = 0, 71, 72, …, 7`

Important!
The check of the binary relationship of TMR with respect to MUR can be switched on and off with the bit „nbr“ in Object 6000. The default value of this bit is zero, which enables the check. If the bit has the value one, the check during SDO download is disabled, allowing to load any value for MUR and TMR.

Before bit „nbr“ is disabled, the values for MUR (Object 6001) and TMR (Object 6002) should be se so that they fulfill the „Binary“ criterion. Otherwise, they will be set automatically to default values.

Object 6001_h : Measuring units per revolution (MUR) (read-write)
This object indicates the number of distinguishable steps per revolution. The description of Object 6000 is to be observed. Depending on the bit „nbr“ in Object 6000, a check of the boundary conditions for MUR and TMR is performed. If this check is performed, in the case of an error, the value is rejected and a corresponding SDO Download Abort-message is sent.

Object 6002_h : Total measuring range (TMR) (read-write)
This object indicates the number of distinguishable steps up to the overflow from the maximum value to zero. The description of Object 6000 is to be observed also here. Depending on the bit „nbr“ in Object 6000, a check of the boundary conditions for MUR and TMR is performed. If this check is performed, in the case of an error, the value is rejected and a corresponding SDO Download Abort-message is sent.

Object 6003_h : Preset (read-write)
This object allows initializing the position value as it is displayed by Object 6004. This value is given in the scaled unit, which is also used to display Object 6004. The maximum value that can be input corresponds to the TMR value, and thus to the value of Object 6002.

Important!
This value is set using the same ASIC interface that is used for reading the position. Therefore, an interruption of the bus cycle time of 250 ms will take place in DC mode.

Object 6004_h : Position (read-only, mappable)
This object supplies, depending on the value of Object 0x6000, the scaled or unscaled position.
It must be noted that this value always loses its validity when objects are activated, which use the same interface to access to the position ASIC. These objects are marked as such and described in this list.

Object 6030_h : Speed (read-only, mappable)
This object supplies, depending on the value of Object 0x6000, the speed in the following units:
• Revolutions per minute (RPM)
• Steps per second
• Steps per 100ms
• Steps per 10ms

Object 6031_h : Speed gating time (read-write)
In compliance with Reference [1], this object defines the time interval to be considered for the calculation of the speed.

Subindex 0
Defines the maximum supported sub-index, which has in this case always the value three.

Subindex 1 (Time Value T)
Defines the base time for the gating time. The true value of the gating time results from the combination of sub-indexes 1, 2 and 3.

Subindex 2 (Multiplier Value M)
The value of sub-index 1 is multiplied by the value of this sub-index.

Subindex 3 (Divider Value D)
The value of sub-index 1 is divided by the value of this sub-index.

Therefore, the gating time G is the result of:
G = T * M / D

Caution!
Maximum permitted gating time is 2 seconds.
If the SDO download value exceeds this value, the encoder reacts with an SDO Download Abort message: „Value of parameter too high". This takes place regardless of the sub-index that eventually caused the overflow.

Object 6400_h : Work Area Status Register (read-only mappable)
Sub-indexes 1 and 2 give two values, with which the current position of the encoder can be evaluated with respect to predefined work areas. See also reference [1] on that subject.
Both work areas are set with the help of Objects 0x6401 and 0x6402.

Object 6401_h : Work area low limit (read-write)
Defines the lower limit value of both areas for the representation by Object 6400.

Object 6402_h : Work area high limit (read-write)
Defines the upper limit value of both areas for the representation by Object 6400.

Object 6501_h : Single turn resolution (read-only)
This Object indicates the singleturn resolution of an encoder. In the case of the Turck sensor, this is usually the value 65536, which corresponds to 16 bits.

Object 6502_h : Number of distinguishable resolutions (read-only)
In the case of a multiturn encoder, this is the multiturn resolution. The Turck sensor has a 12 bits multiturn stage and thus a value of 4096. If the multiturn stage is missing, this object displays the value 1.

Object 6503_h : Alarms (read-only)
This object informs about the occurrence of the following errors on the encoder:
• Position error: „pe“
• Commissioning diagnostic error: „cde“

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	cde	pe

Object 6504_h : Supported alarms (read-only)
This object indicates which error cases are to be signalled externally, and thus in Object 0x6503. These are the Position errors and the Commissioning diagnostic.

Object 6505_h : Warnings (read-only)
This object signals the following warnings of the encoder:
• Light control reserve reached: „lcr“
• Speed range exceeded: „sr“. This bit is set for 9000 RPM.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	-	sr	-	-	-	-	lcr	-

Object 6506_h : Supported warnings (read-only)
This object indicates which warnings are to be signalled externally, and thus in Object 0x6505. These are two warnings: Light control reserve reached and Speed range exceeded.

Object 6507_h : Profile and software version (read-only)
Similarly to Object 0x1A00, the software version is represented in the two upper nibbles. If the software version is V4.5, the value of this object is 0x04050302, where the two lower nibbles 0x0302 represent the CANopen profile version, that is to say 3.2.

Object 6509_h : Offset value (read-only)
The offset value is updated when switching on and later every time the Preset value (Object 0x6003) is activated.

9 Annex: DC cycle times

Number of transmitted bytes	Transmission time in µs
4 as Object 0x6004 with scaled position	55 or 107 depending on cleared or set NBR-Bit in object 0x6000.
5 as 0x6004 and 0x1001	63
10 as 0x6004 (Scaled position), 0x6505 (Warnings), 0x2000 (System time)	76 or 128 depending on cleared or set NBR-Bit in object 0x6000.

10 Annex:
11 Network controllers supported by Beckhoff‘s Software Ethernet Driver

Intel Fast Ethernet Controllers (Vendor ID: 0x8086)

Device ID	Description
0x1029	82559
0x1030	82559
0x1031	82801CAM
0x1032	82801CAM
0x1033	82801CAM
0x1034	82801CAM
0x1038	82801CAM
0x1039	82801CAM
0x103A	82801DB
0x103B	82801DB
0x103C	82801DB
0x103D	82801DB
0x103E	82801DB
0x1050	82801EB/ER
0x1051	82801EB/ER
0x1052	82801EB/ER
0x1053	82801EB/ER
0x1054	82801EB/ER
0x1055	82801EB/ER
0x1056	82801EB/ER
0x1057	82801EB/ER
0x1059	82551QM
0x1064	82801EB/ER
0x1067	Intel PRO/100
0x1068	82562
0x1069	Intel PRO/100
0x106A	Intel PRO/100
0x106B	Intel PRO/100
0x1094	Intel PRO/100
0x1209	8255xER/IT
0x1229	82557/8/9/0/1
0x1249	82559ER
0x1259	82801E
0x245D	82801E
0x27DC	Intel PRO/100

Intel Gigabit Ethernet Controllers (Vendor ID: 0x8086)

Device ID	Description
0x1000	82542
0x1001	82543GC
0x1004	82543GC
0x1008	82544EI
0x1009	82544EI
0x100C	82544EI
0x100D	82544GC
0x100E	82540EM
0x100F	82545EM
0x1010	82546EB
0x1011	82545EM
0x1012	82546EB
0x1013	82541EI
0x1014	82541ER
0x1015	82540EM
0x1016	82540EP
0x1017	82540EP
0x1018	82541EI
0x1019	82547EI
0x101A	82547EI
0x101D	82546EB
0x101E	82540EP
0x1026	82545GM
0x1027	82545GM
0x1028	82545GM
0x1049	82566MM
0x104A	82566DM
0x104B	82566DC
0x104C	82562V
0x104D	82566MC
0x104E	82571EB
0x104F	82571EB
0x1060	82571EB
0x1075	82547EI
0x1076	82541GI
0x1077	82547EI
0x1078	82541ER
0x1079	82546EB
0x107A	82546EB
0x107B	82546EB
0x107C	82541GI
0x107D	82572EI
0x107E	82572EI
0x107F	82572EI
0x108A	82546GB
0x108B	82573E
0x108C	82573E
0x1096	80003ES2LAN
0x1098	80003ES2LAN
0x1099	82546GB
0x109A	82573L
0x10A4	82571EB
0x10A7	82575

0x10A9	82575
0x10B5	82546GB
0x10B9	82572EI
0x10BA	80003ES2LAN
0x10BB	80003ES2LAN
0x10BC	82571EB
0x10C4	82562GT
0x10C5	82562G
0x10C9	82576
0x10D3	82574L

12 References

[1] CANopen device profile for encoders. CiA 406 Work Draft. Version 3.2.10

[2] CANopen Application Layer Communication Profile. CiA Draft Standard 301.

[3] EtherCAT Specification – Part 6. Application Layer protocol specification.

Turck Inc. sells its products through Authorized Distributors. These distributors provide our customers with technical support, service and local stock. Turck distributors are located nationwide – including all major metropolitan marketing areas.
For Application Assistance or for the location of your nearest Turck distributor, call:
1-800-544-7769

Specifications in this manual are subject to change without notice. Turck also reserves the right to make modifications and makes no guarantee of the accuracy of the information contained herein.
Literature and Media questions or concerns?
Contact Marketing Turck USA – tusa.marketing@turck.com

TURCK

28 subsidiaries and over
60 representations worldwide!

Printed in USA

MA1027 A 12/16

www.turck.com