Sherpa R-IN32M3-EC EtherCAT device communication stack for Renesas Electronics Corporation's R-IN32M3 series industrial Ethernet controller

Technical reference

Sherpa LLC
http://sherpa-tech.jp

Version R01_06
2016-05-05

R-IN32M3 is registered trademark of Renesas Electronics Corporation.
Simple Device Application Interface SDAI is trademark of Softing Industrial Automation GmbH
Windows is registered trademark of Microsoft Corporation
Simatic S7-1200, S7-300, S7-400, S7-1500 are registered trademark of Siemens AG
TwinCAT® is registered trademark of Beckhoff Automation GmbH & Co. KG
IAR KickStart Kit™ is registered trademark of IAR Systems
CC-Link and CC-Link/IE are registered trademarks of CC-Link Partner Association
PROFINET is registered trademark of PROFIBUS & PROFINET International (PI)
EtherCAT is registered trademark of EtherCAT Technology Group (ETG)
EtherNet/IP™ and Common Industrial Protocol (CIP™) are registered trademarks of ODVA Inc.
Modbus TCP is registered trademark of Modbus Organization, Inc.
POWERLINK is registered trademark of Ethernet POWERLINK Standardization Group (EPSG)
Wireshark is registered trademark of Wireshark Foundation

© Copyright 2015 Sherpa LLC

No part of these instructions may be reproduced or processed, copied or distributed in any form whatsoever without prior written permission by Sherpa LLC. Any violations will lead to compensation claims.
All rights are reserved, particularly with regard to patent issue or TM (trade mark) registration.
All other product or service names are the property of their respective holders.
The producer reserves the right to make changes to the scope of supply as well as to technical data, even without prior notice. Careful attention was given to the quality and functional integrity in designing, manufacturing and testing the system. However, no liability can be assumed for potential errors that might exist or for their effects. In particular, Sherpa LLC cannot assume liability in terms of suitability of the system for a particular application. Should you find errors, please inform your distributor of the nature of the errors and the circumstances under which they occur. Sherpa LLC will be responsive to all reasonable ideas and will follow up on them, taking measures to improve the product, if necessary.

Table of Contents

1. Overview
2. Delivery overview
3. Overview of Sherpa LLC's EtherCAT slave communication stack licensing
4. Evaluation and development environment
5. Support Scope
6. R-IN32M3-EC and Sherpa EtherCAT slave
7. Delivery description
8. EtherCAT Stack documentation
9. Simple Device Application Interface
10. Sample Application
11. TwinCAT® program and Sherpa EtherCAT slave application
12. Sherpa EtherCAT application's external interface in IAR System board
13. Licensing, product development and additional services
14. EtherCAT slave stack functionality

1. Overview

This document is the technical reference for Sherpa LLC's EtherCAT slave communication stack and its corresponding Sherpa LLC's EtherCAT slave evaluation kit. This EtherCAT device communication stack has been optimized for the Renesas Electronics Corporation R-IN32M3 industrial network LSI and is the result of the Renesas Electronic's EtherCAT slave stack ported into the R-IN32M environment. Additionally, the user interface for the EtherCAT slave stack is wrapped under the Softing Industrial Automation's Simple Device Application Interface (SDAI). This document covers the description of the communication stack and its access library, setup of sample application on evaluation board, description of sample application, description of PLC program, scope of support, licensing, additional services and EtherCAT device stack specification.

2. Delivery overview

The Sherpa LLC's EtherCAT slave evaluation kit consists of a downloadable image which contains this technical document as well as the following data:

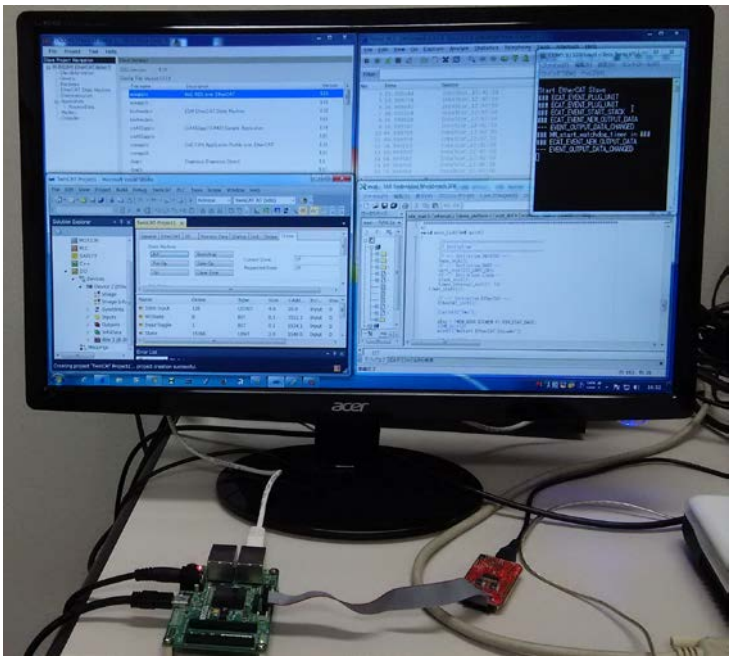
- Sample EtherCAT slave application in sources optimized for evaluation board described later in this document.
- Evaluation EtherCAT slave stack in binary format, with the full EtherCAT slave functionality but limited to 90 minutes of continued operation. By restarting the sample application, the PROFINET device stack can work normally for 90 minutes.
- ESI (EtherCAT Slave Information) file for the Sherpa LLC's EtherCAT slave evaluation kit sample application and the binary configuration for the ESC (EtherCAT Slave Controller) core in R-IN32M3.
- Sample programmable logic controller (PLC) program for TwinCAT® application.
- Additional documentation for detailed access library description, application description and EtherCAT stack description from Softing Industrial Automation GmbH.

3. Overview of Sherpa LLC's EtherCAT slave communication stack licensing

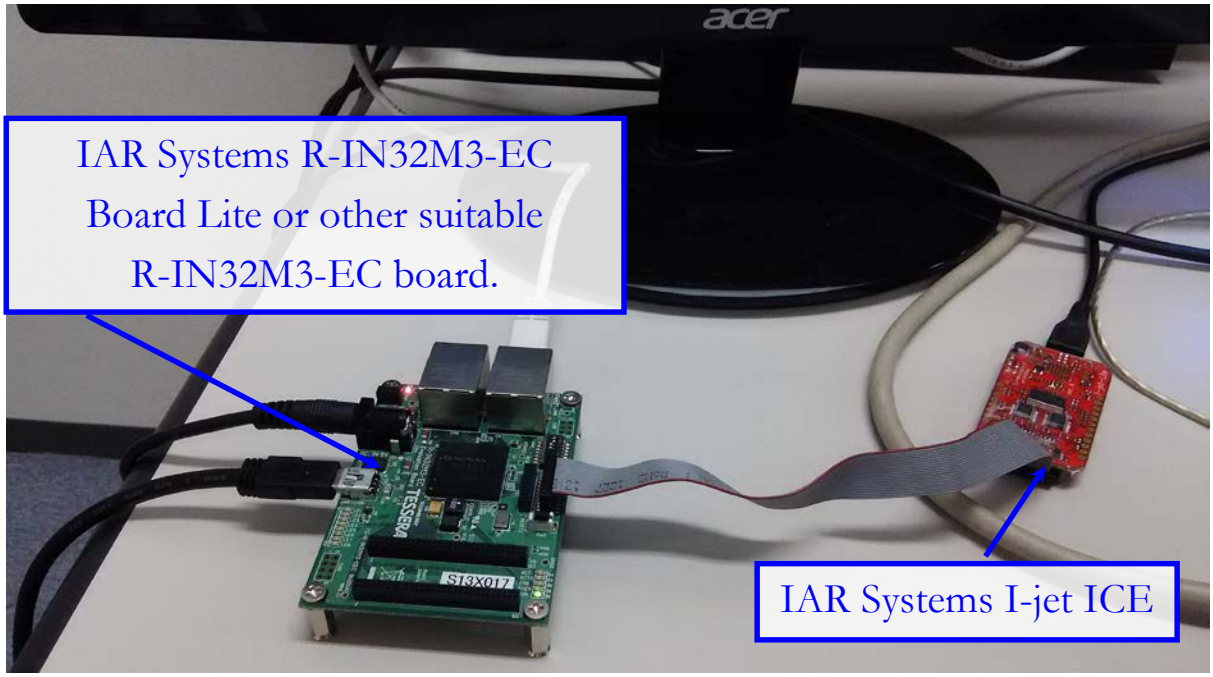
The EtherCAT slave communication stack provided as part of the Sherpa LLC's EtherCAT slave evaluation kit is an evaluation product. Its use is strictly restricted for evaluation in laboratory or display environment. This product is not licensed for use in actual industrial devices and the sale of this evaluation EtherCAT device communication stack is strictly prohibited. In order to use this communication stack in commercial products the device manufacturer must sign a contract with the owner of the intellectual property of this communication stack, Sherpa LLC. For licensing conditions please see clause "Licensing, product development and additional services" at the end of this document.

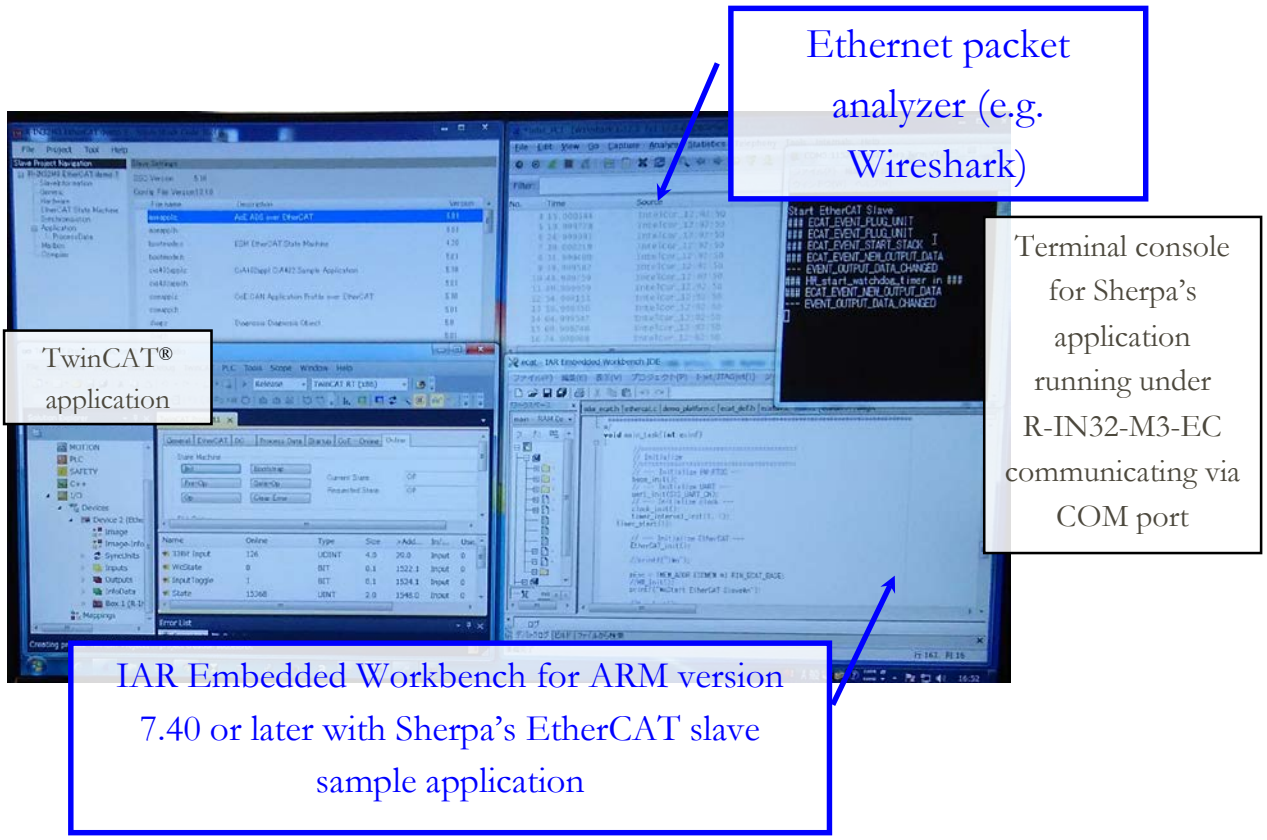
4. Evaluation and development environment

In order to successfully use the Sherpa LLC's EtherCAT slave evaluation kit in any meaningful way the below minimum setup is required.



Windows workstation with TwinCAT® software, IAR System Workbench tool, terminal for Sherpa's EtherCAT slave application and Ethernet packet analyzer.





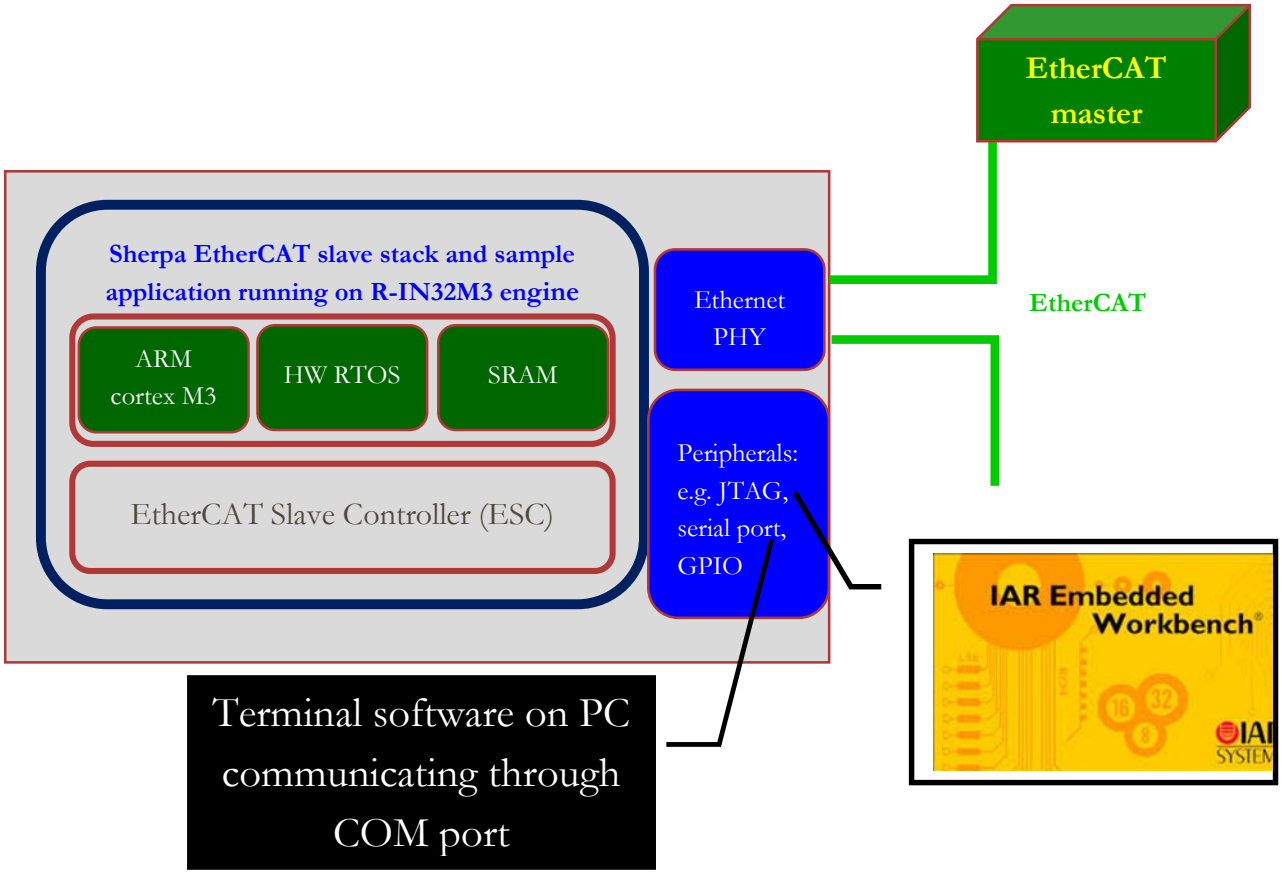
5. Support Scope

The Sherpa LLC's EtherCAT slave evaluation kit has been thoroughly tested and confirmed to work in environment described in the above sections. Should this application be used in "any" kind of different environment Sherpa LLC will regard any inquiry on the use of this EtherCAT slave kit as technical assistance beyond the scope of support for this evaluation application. In this context, "different environment" definition and not covered technical assistance includes, but is not limited to, the below circumstances:

- Any modification of the sources of this sample application
- Use of a compiler other than IAR Systems Embedded Workbench 7.40 or later.
  Note: Sherpa LLC product is optimized for the IAR Systems compiler. Use of any other compiler is not warranted and may require development efforts to be requested to Sherpa LLC
- Use of an EtherCAT master other than the TwinCAT® as described in this document.
- Any workshop that the end-user may require with regards to EtherCAT technology, use of IAR Systems Embedded Workbench tool, use of TwinCAT® tools or use of EtherCAT master configuration tools from other vendors, use of Wireshark software, etc..

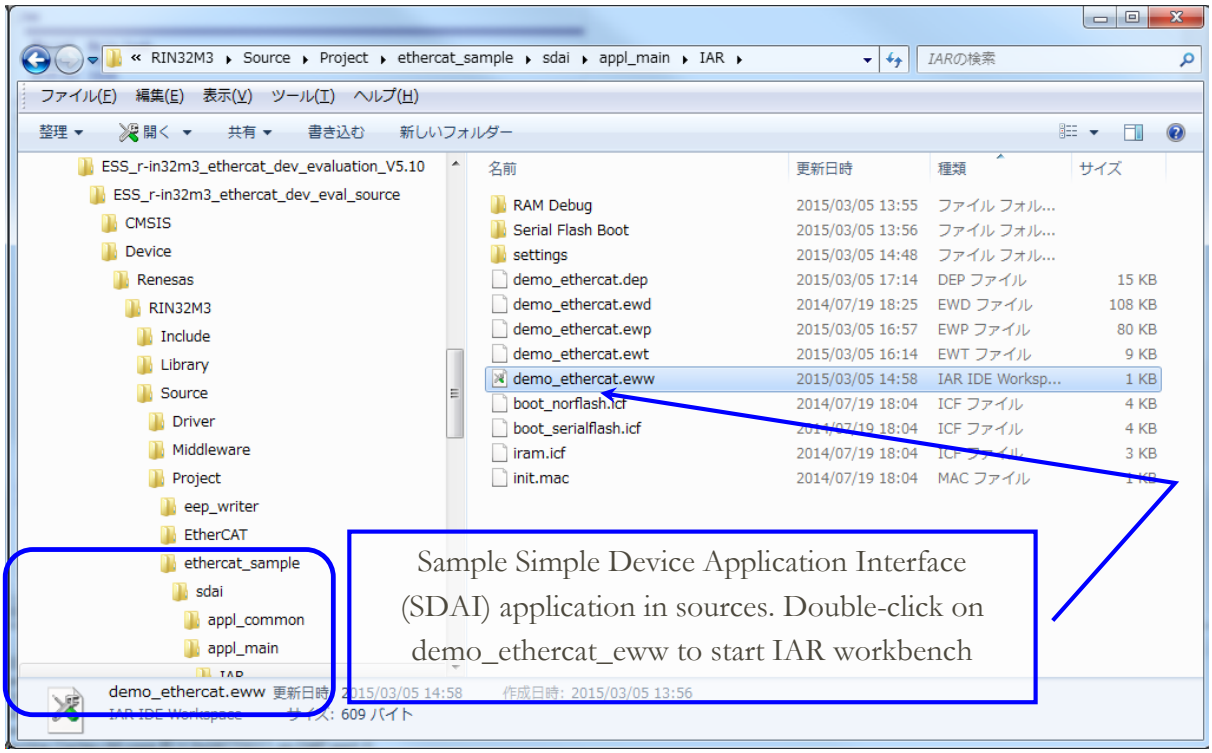
6. R-IN32M3-EC and Sherpa EtherCAT slave

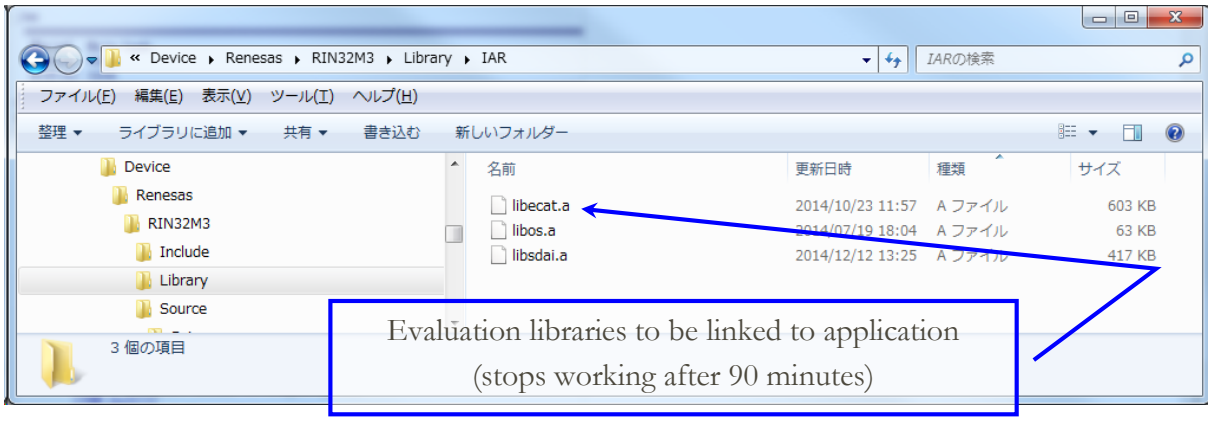
The Sherpa LLC's EtherCAT slave evaluation kit is optimized for the R-IN32M3-EC and is described as a "simplified" high level block diagram as per below illustration:



7. Delivery description

This section lists the main files that conform the Sherpa LLC's EtherCAT slave evaluation kit with comments on sections relevant to Sherpa delivery:





8. EtherCAT Stack documentation

The EtherCAT slave stack for R-IN32M3-EC has been developed by Sherpa LLC by porting the Softing EtherCAT slave communication stack into R-IN32M3-EC architecture. The EtherCAT functionality of this delivery conforms to the Softing product. Detailed explanations are provided in the Softing documentation which is part of the delivery.

NOTE: The Sherpa EtherCAT slave communication stack for R-IN32M3-EC is licensed and supported by Sherpa LLC. The Softing documentation provided in this delivery is published here with the consent of Softing Industrial Automation GmbH. All support inquiries for the Sherpa LLC's EtherCAT slave evaluation kit should be addressed to Sherpa LLC.

Softing and Sherpa LLC continue working together in the constant evolution and improvement of the EtherCAT slave communication stack. Improvements on the Softing stack will be made available on the Sherpa LLC's EtherCAT slave evaluation kit within a reasonable time frame.

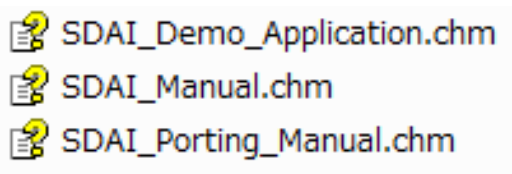
9. Simple Device Application Interface

The application programming interface of the Sherpa communication stack is based on Softing's Simple Device Application Interface (SDAI). Detailed explanations are provided in the Softing documentation which is part of the delivery.

10. Sample Application

The sample application of Sherpa LLC's EtherCAT slave evaluation kit is based on Softing's sample application. Detailed explanations are provided in the Softing documentation which is part of the delivery.

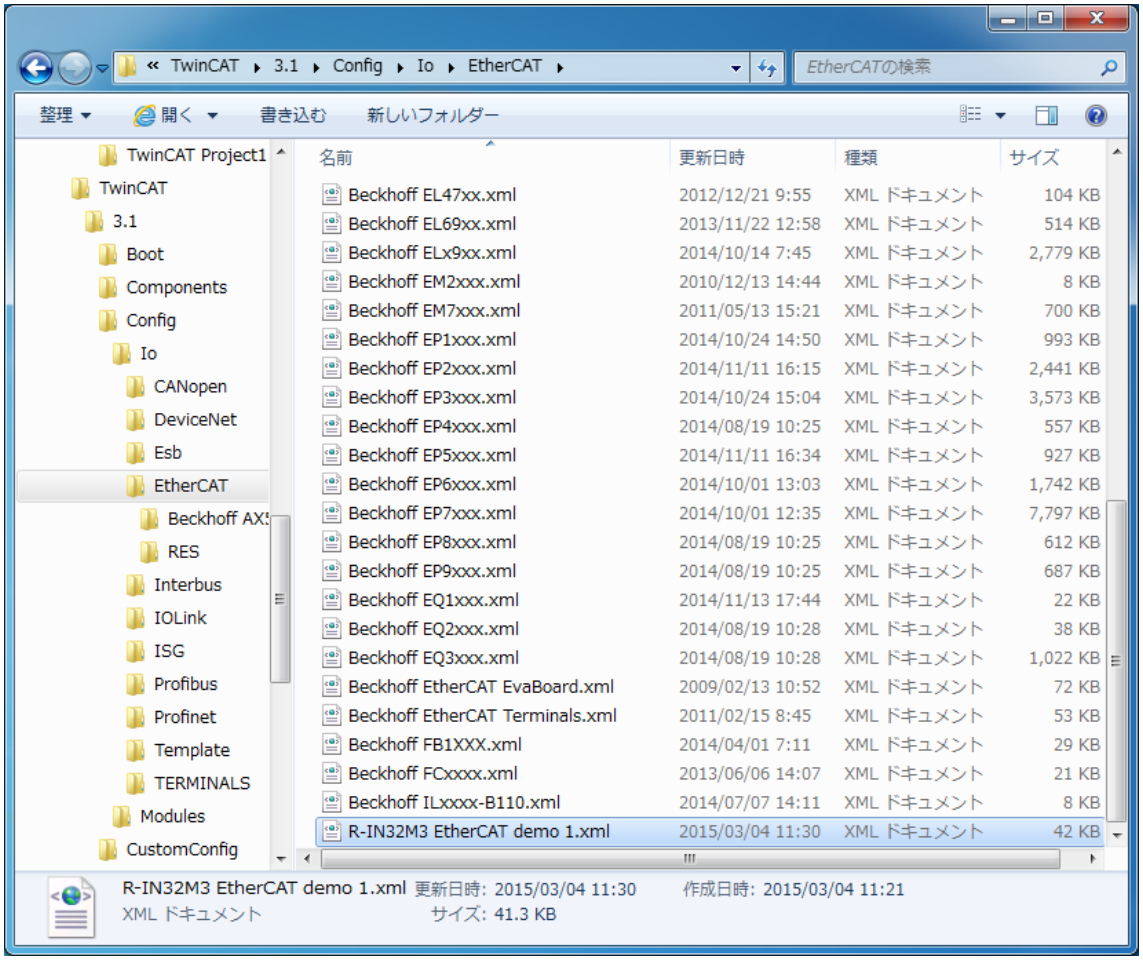
The Softing documentation provided in this delivery is shown below:



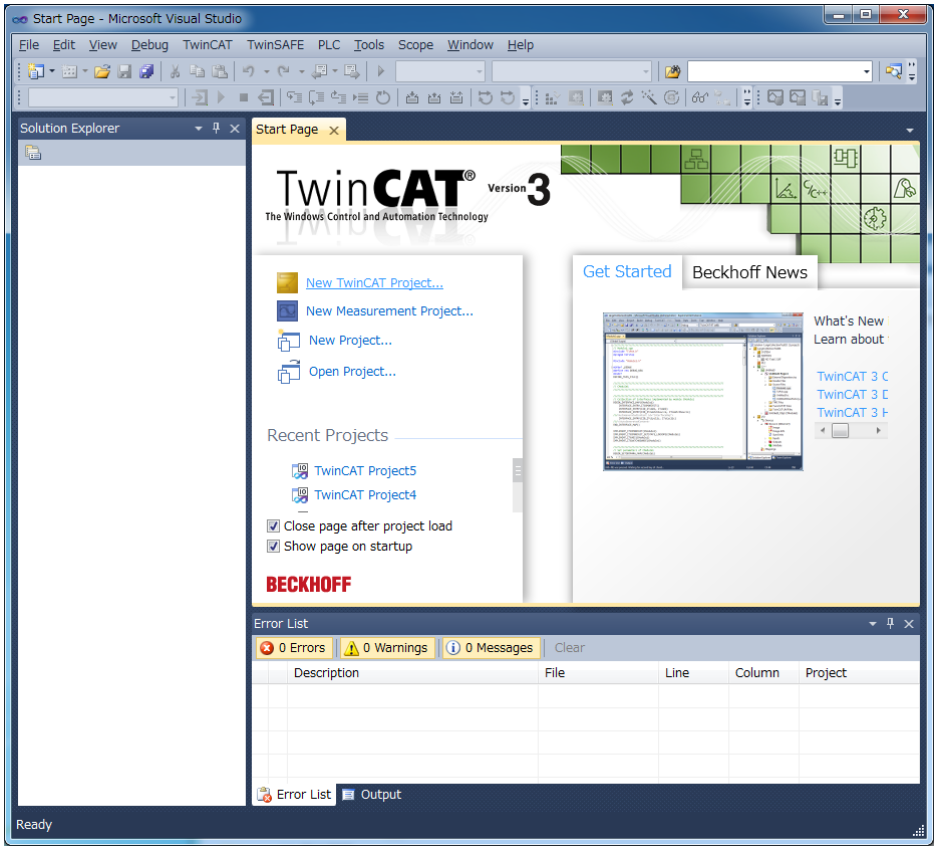
11. TwinCAT® program and Sherpa EtherCAT slave application

The setup of the TwinCAT® application can be a complex process. The below procedure may be helpful for first-time TwinCAT® users. If the reader does not have TwinCAT® license, a 7-day trial license is available from the vendor of TwinCAT®. The software must be properly installed on a Windows PC with suitable hardware. Normally TwinCAT® will not work with commercial USB-Ethernet LAN adapters, and some commercial LAN chipsets are not supported by TwinCAT®. Normally TwinCAT® works best with Intel® LAN controllers.

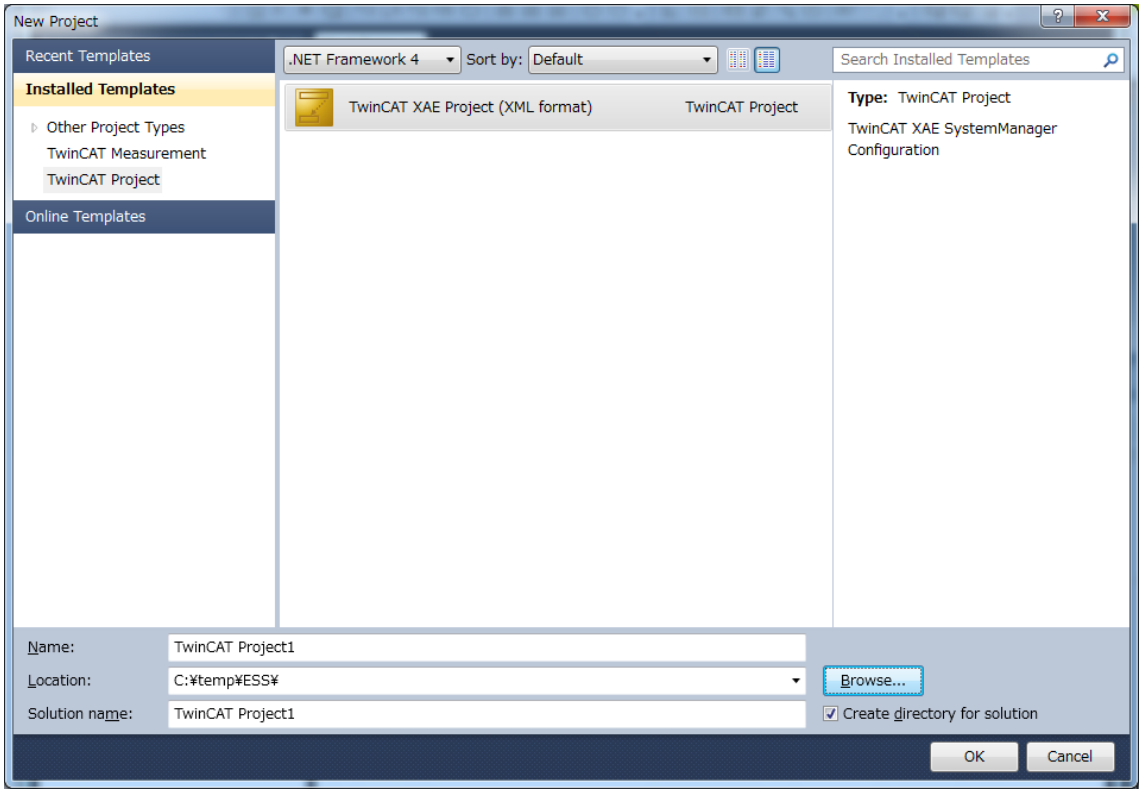
After installing TwinCAT® copy the provided R-IN32M3 EtherCAT demo 1.xml ESI (EtherCAT Slave Information) file to the directory in path C:\TwinCAT\3.1\Config\Io\EtherCAT, in order to add it to TwinCAT®'s EtherCAT devices library.



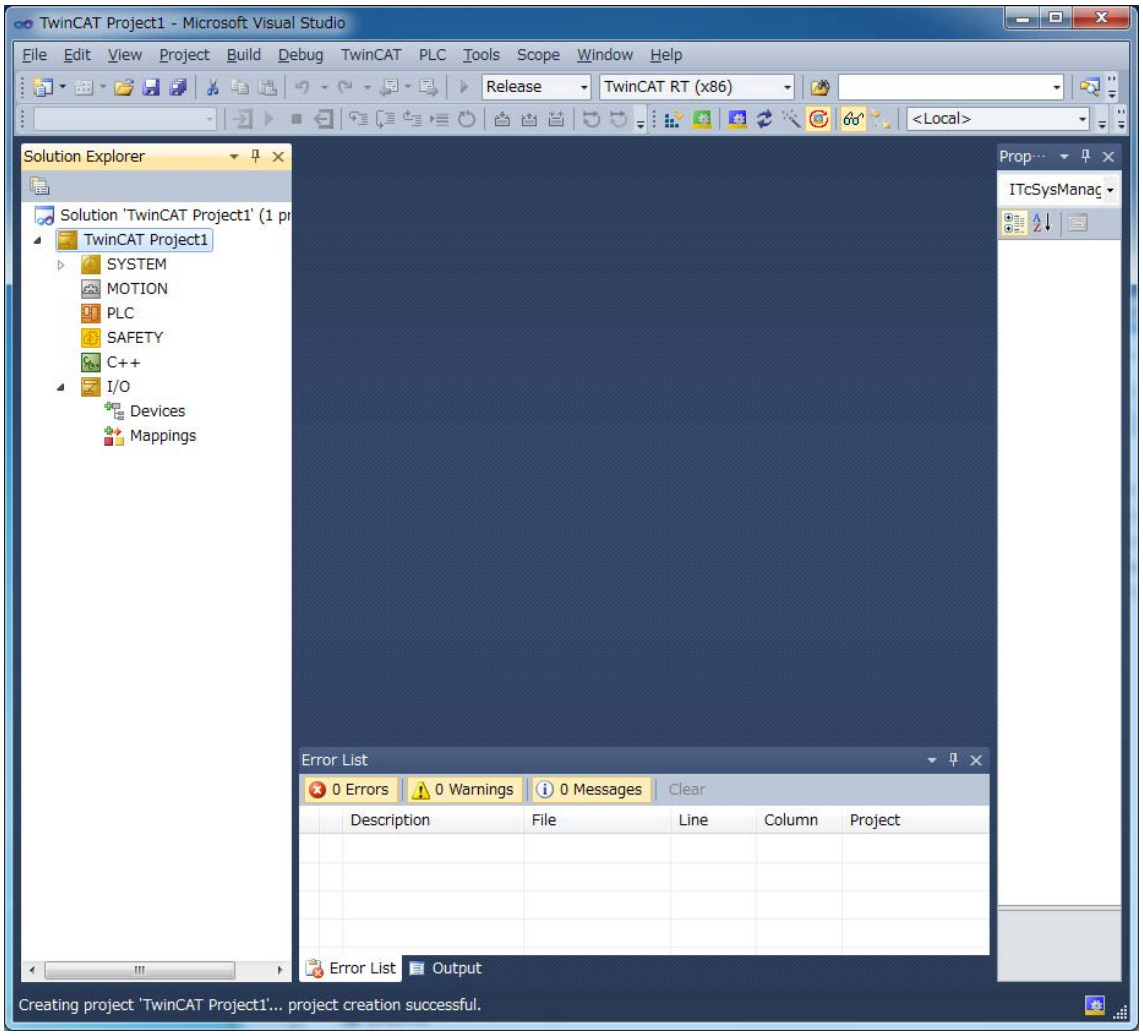
Start TwinCAT® and click New Project to create a new project.



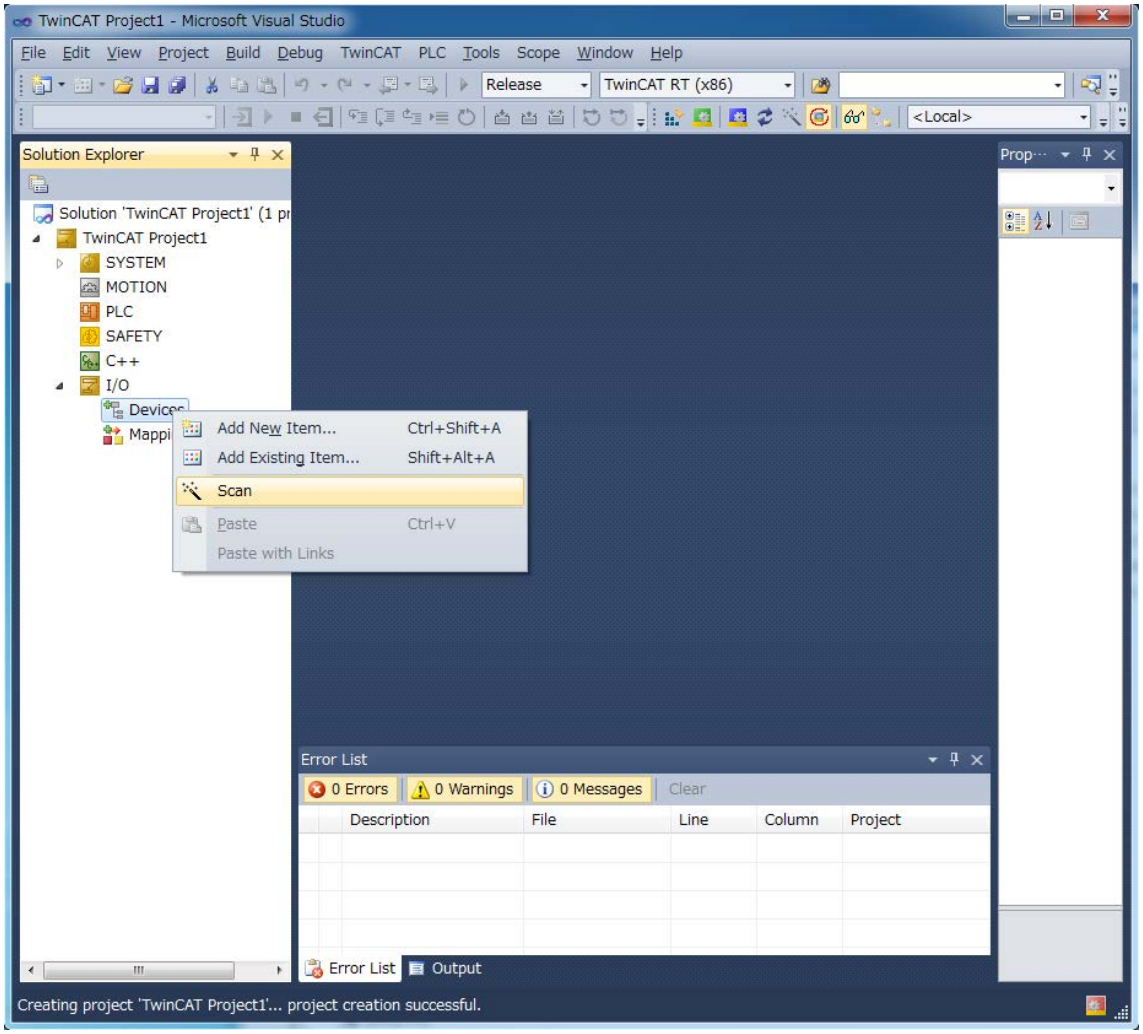
After selecting the save directory and writing the project's name the below screen is seen.



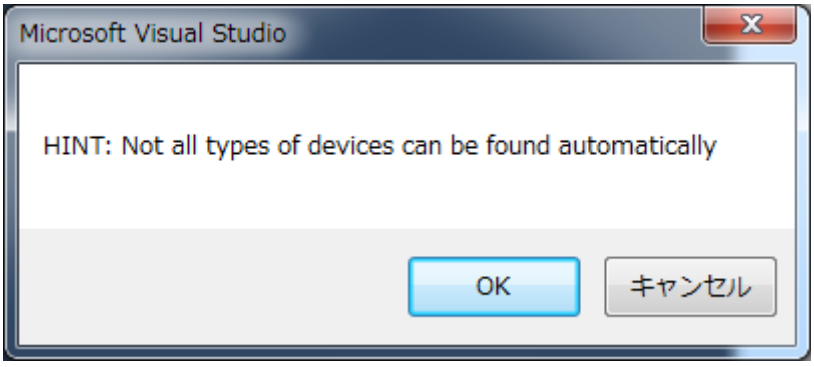
After clicking OK the TwinCAT® project is created and saved.



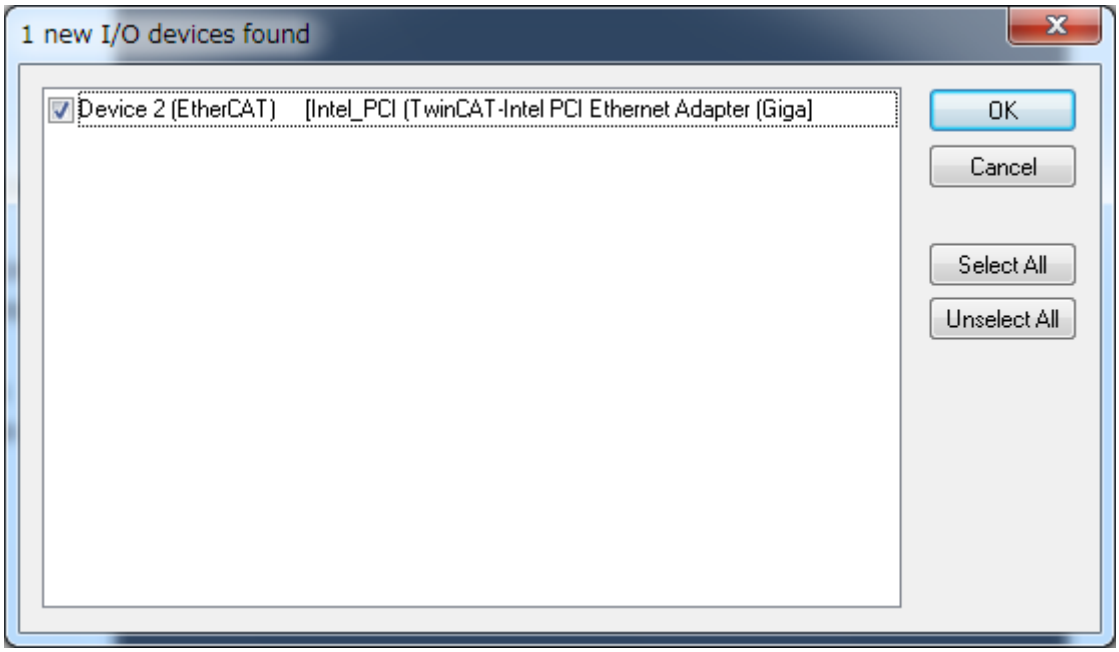
Connect the R-IN32M3-EC evaluation board, after having started the sample application, to the computer running TwinCAT®. Right-click on “Devices” and select “Scan”.



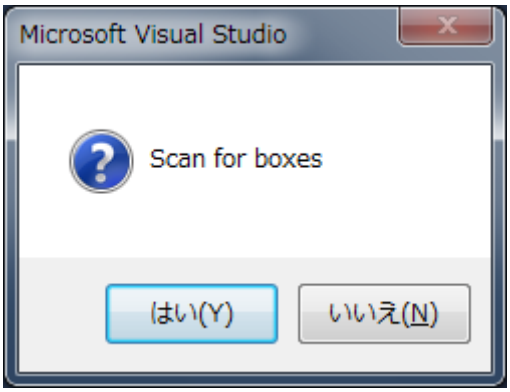
Click OK on screen below:



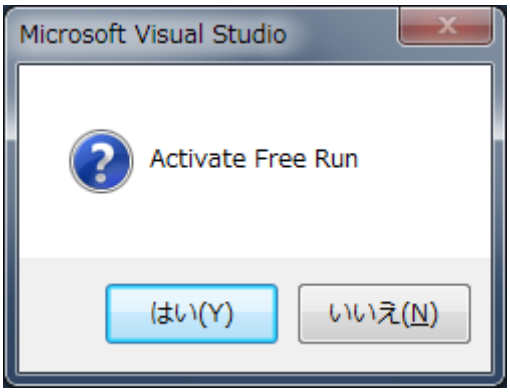
Select the TwinCAT® compatible LAN card and click OK in screen below:



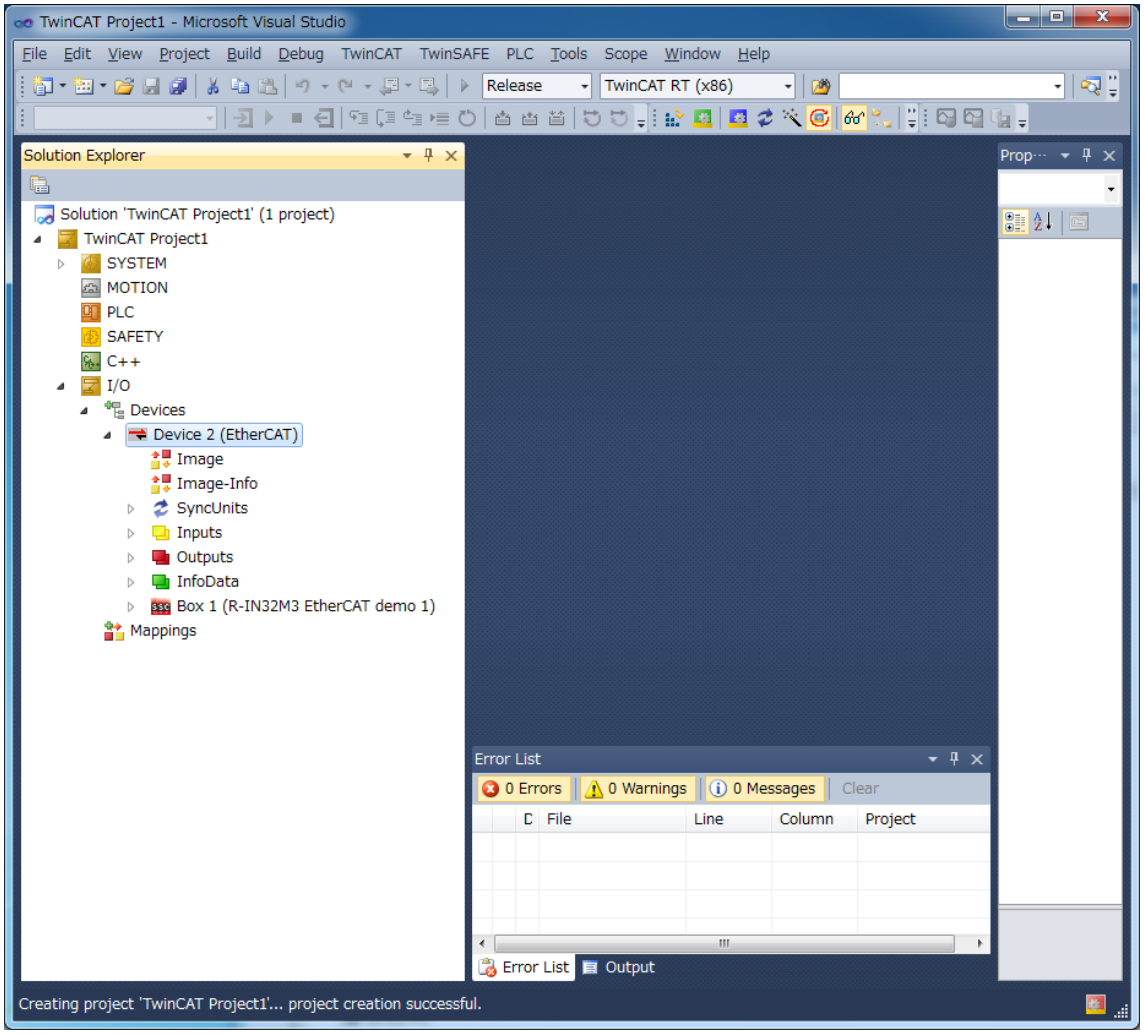
Next click “Yes” (Y) on screen below:



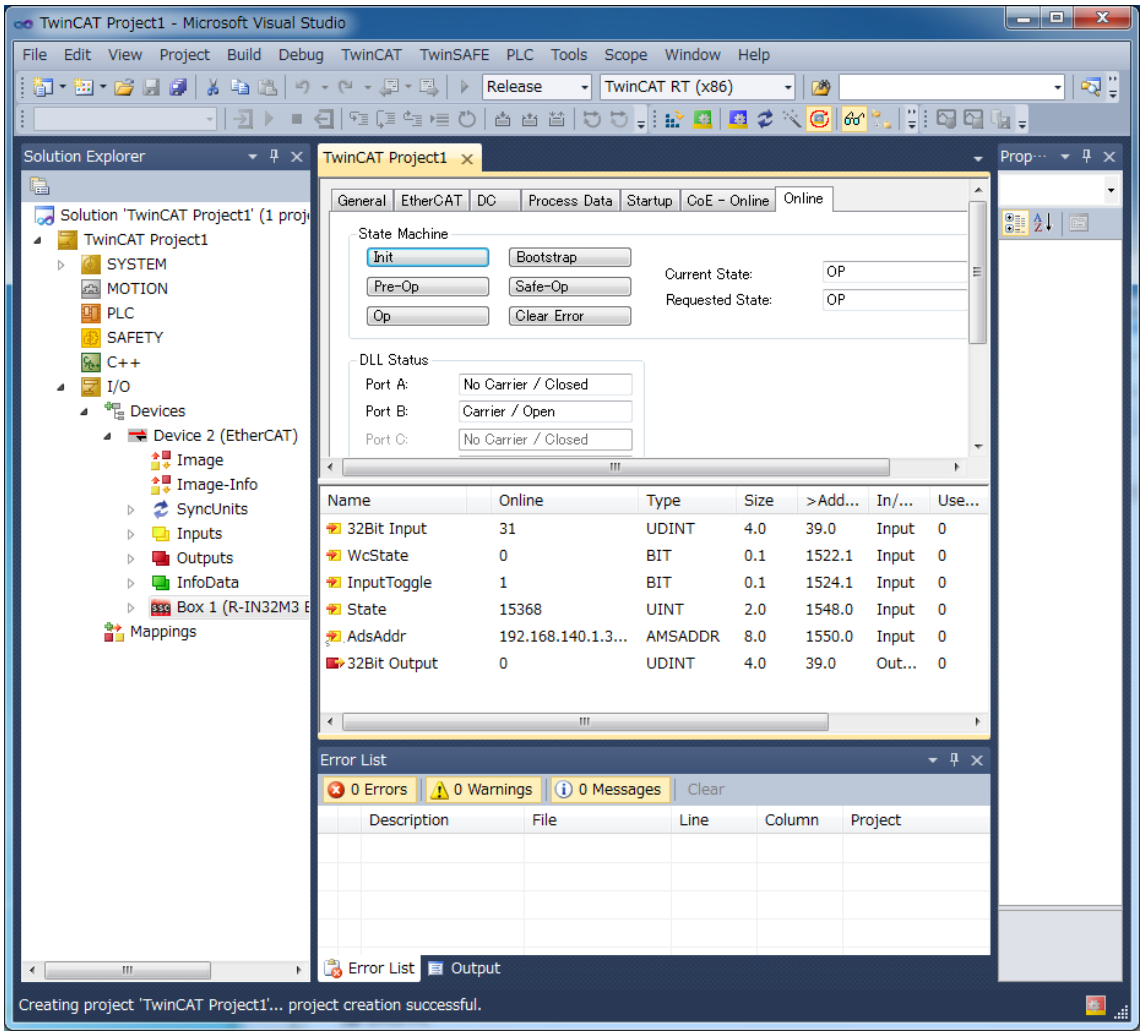
In order to change to execution mode click “Yes” (Y) on screen below:



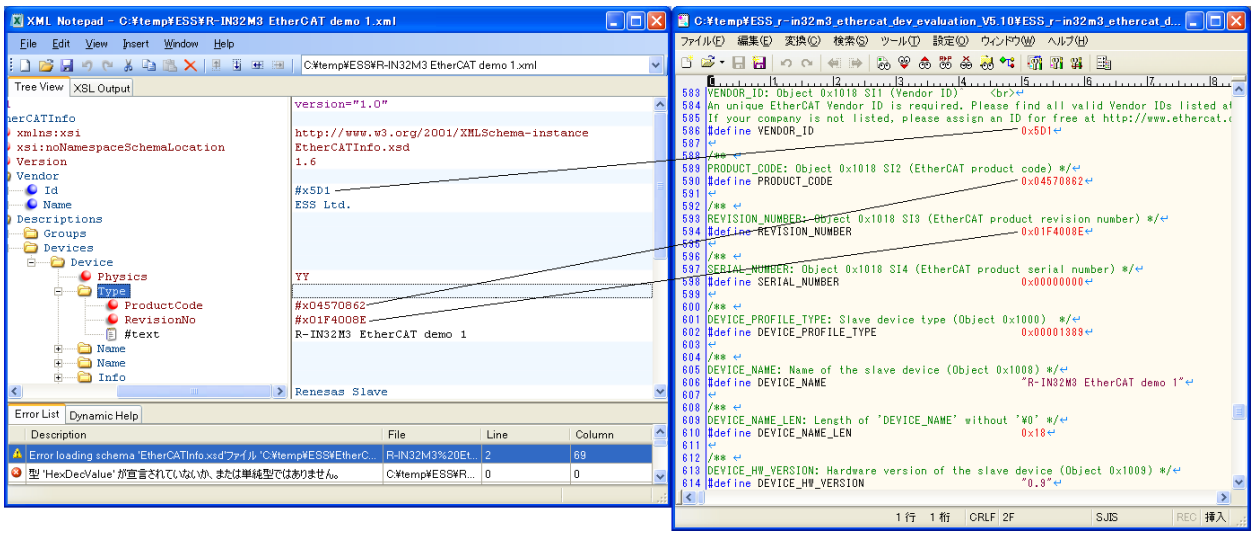
If the sample application in the R-IN32M3-EC card is working properly with the IAR Workbench and it is connected with a LAN cable to the computer running TwinCAT® the below screen showing detection of the ESS sample application should appear:



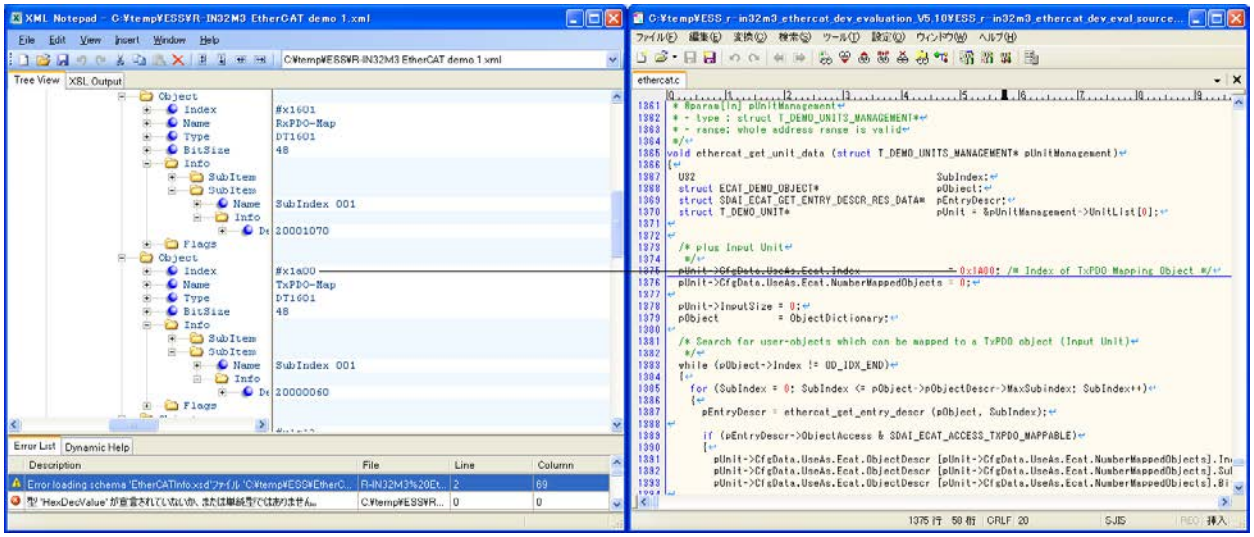
In order to confirm whether the EtherCAT slave application is working, double-click the “Box 1” for the EtherCAT slave to display its property page. In the “Online” tab the “Current State” should indicate “OP” (operational), which indicates that the application is working properly.

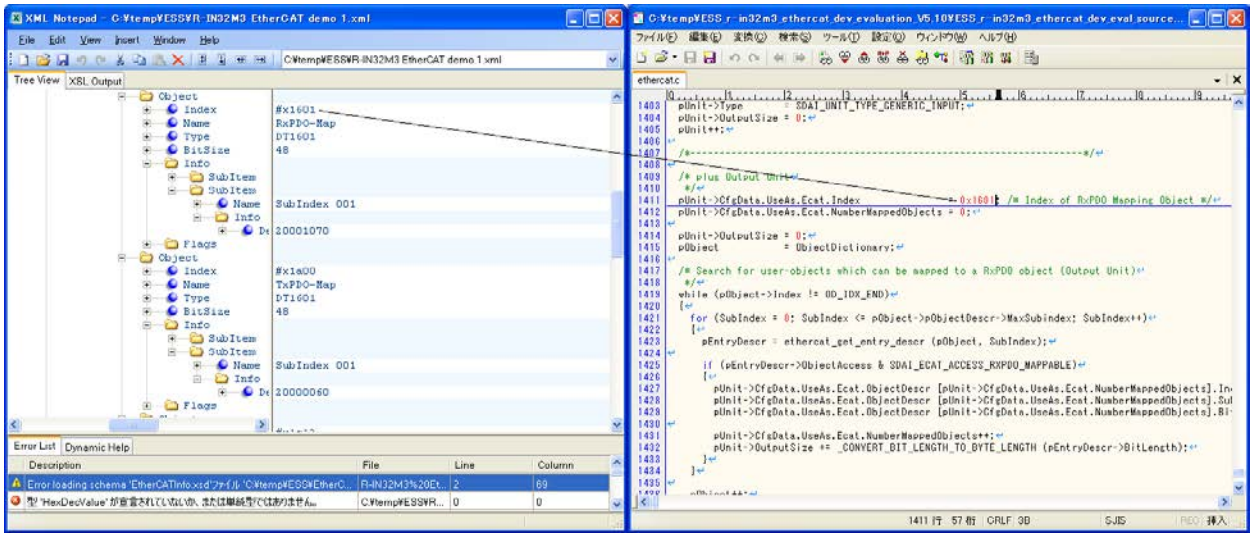


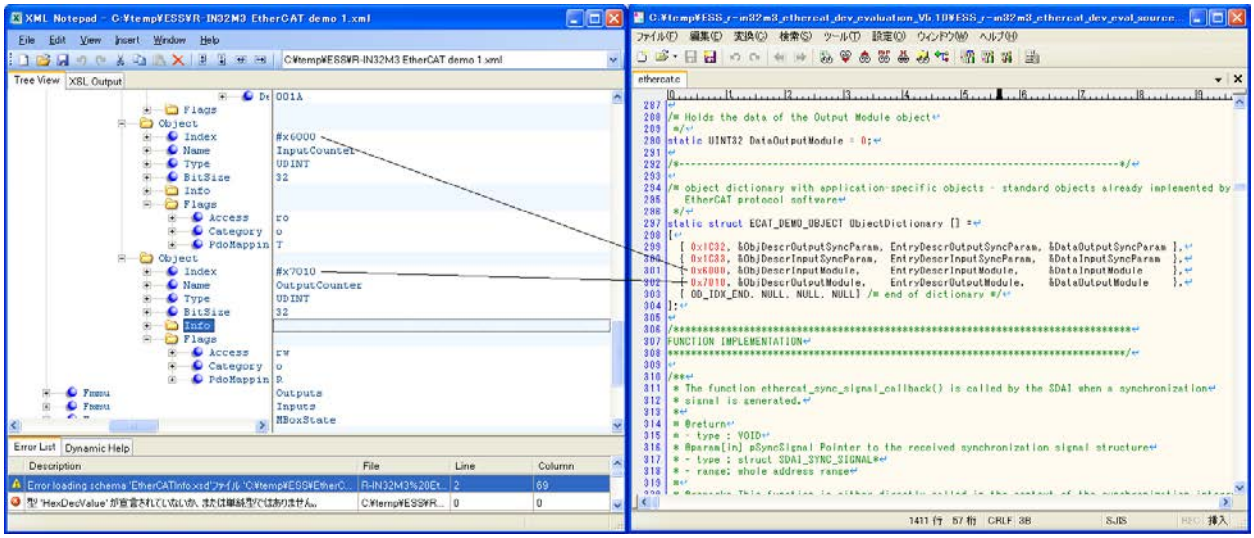
The below screen shows correspondence between ID information as defined in the ESI file and the corresponding ID information as programmed in the sample application.

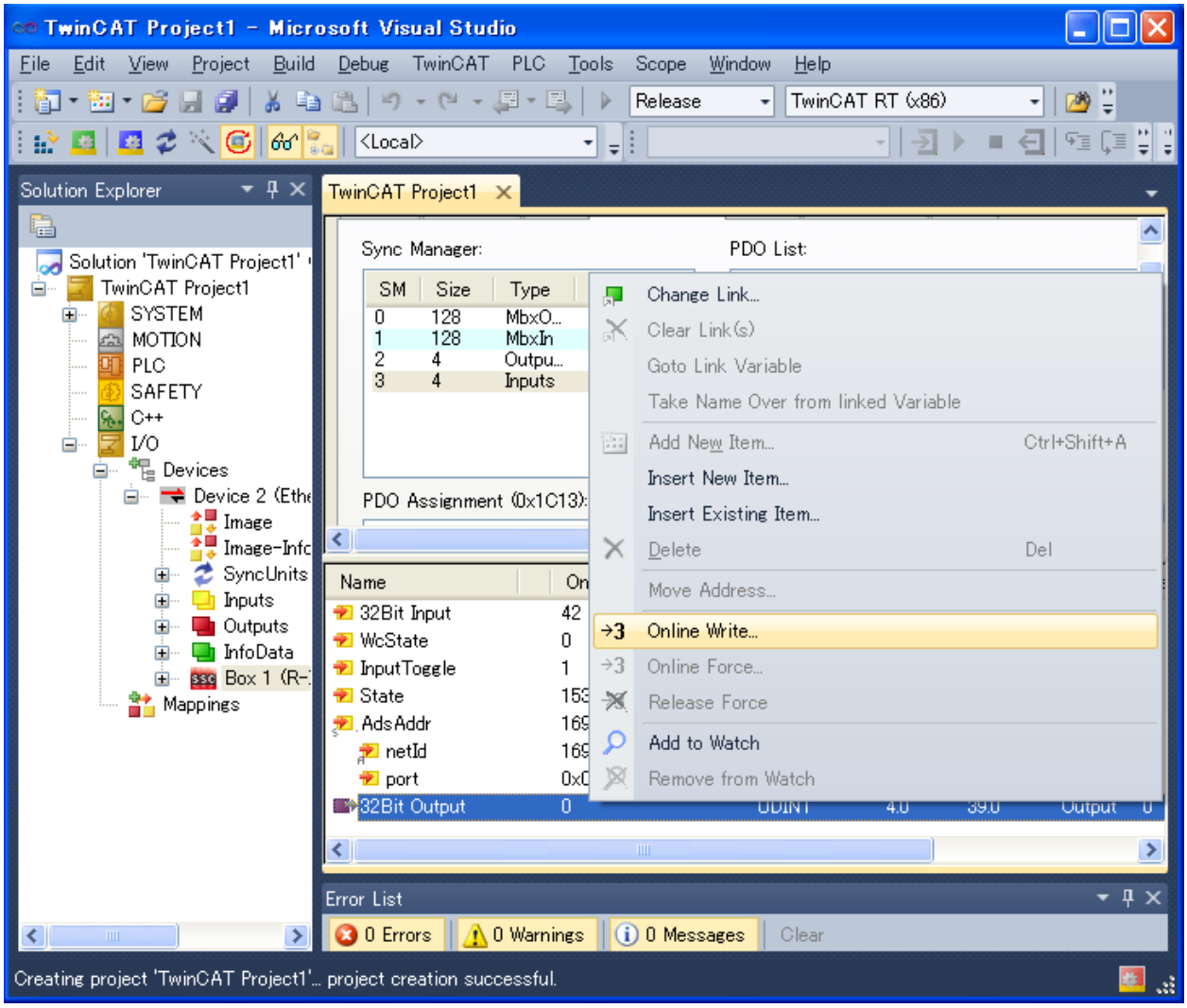


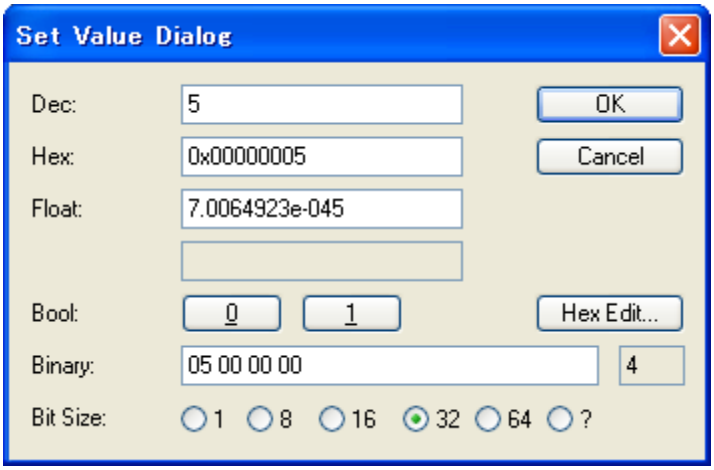
The IO configuration as defined in the EtherCAT master has to correspond to the actual modules as configured by the sample application running on R-IN32M3-EC. The below screenshot shows how the SDAI application configures the object dictionary entries from the ESI file.

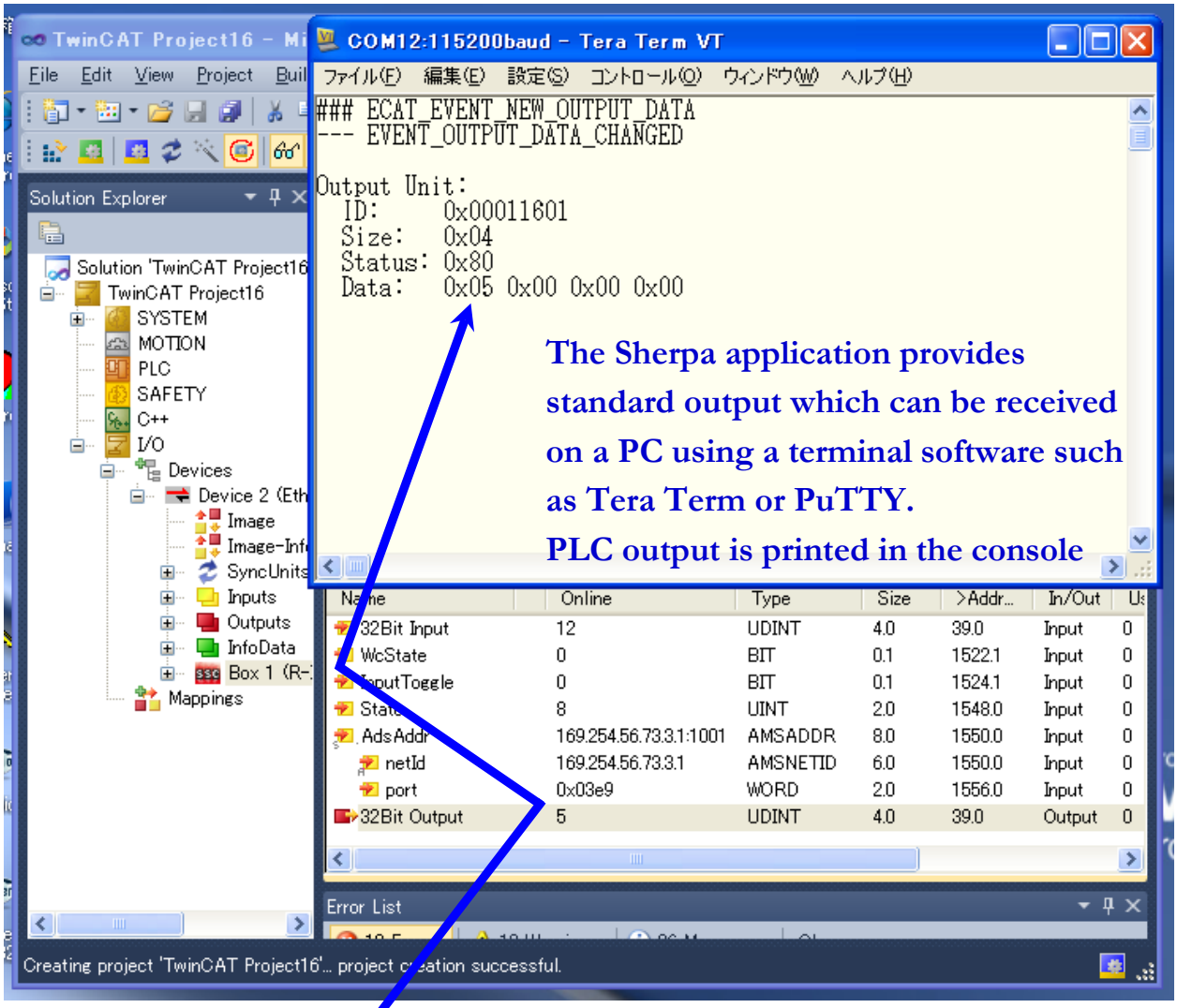


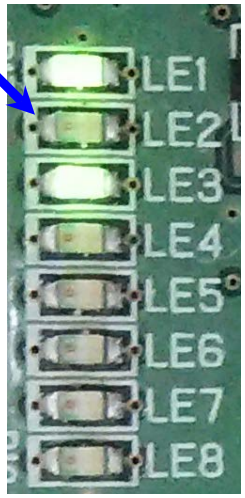

XML Notepad - C:¥temp¥ESS¥R-IN32M3 EtherCAT demo 1.xml
File Edit View Insert Window Help
Tree View XML Output
Object
Index #x1601
Name RxPDO-Map
Type DT1601
BitSize 48
Info
SubItem
SubItem
Name SubIndex 001
Info
Flags
Object
Index #x1a00
Name TxPDO-Map
Type DT1601
BitSize 48
Info
SubItem
SubItem
Name SubIndex 001
Info
Flags
Error List Dynamic Help
Description File Line Column
Error loading schema 'EtherCATInfo.xsd'ファイル 'C:¥temp¥ESS¥EtherC... R-IN32M3%20Et... 2 69
型 'HexDecValue' が宣言されていないか、または単純型ではありません。 C:¥temp¥ESS¥R... 0 0
C:¥temp¥ESS_r_in32m3_ethercat_dev_evaluation_V5.10¥ESS_r_in32m3_ethercat_dev_eval_source...
ファイル(F) 編集(E) 変換(C) 検索(S) ツール(T) 設定(O) ウィンドウ(W) ヘルプ(H)
ethercat.c
pUnit->Type = SDAI_UNIT_TYPE_GENERIC_INPUT;
pUnit->OutputSize = 0;
pUnit++;
/* plus Output Unit
*/
pUnit->CfgData.UseAs.Ecat.Index = 0x1601; /* Index of RxPDO Mapping Object */
pUnit->CfgData.UseAs.Ecat.NumberMappedObjects = 0;
pUnit->OutputSize = 0;
pObject = ObjectDictionary;
/* Search for user-objects which can be mapped to a RxPDO object (Output Unit)
*/
while (pObject->Index != OD_IDX_END)
for (SubIndex = 0; SubIndex <= pObject->pObjectDescr->MaxSubindex; SubIndex++)
pEntryDescr = ethercat_get_entry_descr (pObject, SubIndex);
if (pEntryDescr->ObjectAccess & SDAI_ECAT_ACCESS_RXPDO_MAPPABLE)
pUnit->CfgData.UseAs.Ecat.ObjectDescr [pUnit->CfgData.UseAs.Ecat.NumberMappedObjects].In
pUnit->CfgData.UseAs.Ecat.ObjectDescr [pUnit->CfgData.UseAs.Ecat.NumberMappedObjects].Su
pUnit->CfgData.UseAs.Ecat.ObjectDescr [pUnit->CfgData.UseAs.Ecat.NumberMappedObjects].Bi
pUnit->CfgData.UseAs.Ecat.NumberMappedObjects++;
pUnit->OutputSize += _CONVERT_BIT_LENGTH_TO_BYTE_LENGTH (pEntryDescr->BitLength);
1411 行 57 桁 CRLF 3B SJIS


XML Notepad - C:¥temp¥ESS¥R-IN32M3 EtherCAT demo 1.xml
File Edit View Insert Window Help
Tree View XML Output
001A
Flags
Object
Index #x6000
Name InputCounter
Type UDINT
BitSize 32
Info
Flags
Access ro
Category o
PdoMappin T
Object
Index #x7010
Name OutputCounter
Type UDINT
BitSize 32
Info
Flags
Access rw
Category o
PdoMappin R
Fmmu Outputs
Fmmu Inputs
MBoxState
Error List Dynamic Help
Description File Line Column
Error loading schema 'EtherCATInfo.xsd'ファイル 'C:¥temp¥ESS¥EtherC... R-IN32M3%20Et... 2 69
型 'HexDecValue' が宣言されていないか、または単純型ではありません。 C:¥temp¥ESS¥R... 0 0
C:¥temp¥ESS_r_in32m3_ethercat_dev_evaluation_V5.10¥ESS_r_in32m3_ethercat_dev_eval_source...
ファイル(F) 編集(E) 変換(C) 検索(S) ツール(T) 設定(O) ウィンドウ(W) ヘルプ(H)
ethercat.c
/* Holds the data of the Output Module object
*/
static UINT32 DataOutputModule = 0;
/* object dictionary with application-specific objects - standard objects already implemented by
EtherCAT protocol software
*/
static struct ECAT_DEMO_OBJECT ObjectDictionary [] =
[0x1C32, &ObjDescrOutputSyncParam, EntryDescrOutputSyncParam, &DataOutputSyncParam],
[0x1C33, &ObjDescrInputSyncParam, EntryDescrInputSyncParam, &DataInputSyncParam],
0x6000, &ObjDescrInputModule, EntryDescrInputModule, &DataInputModule],
0x7010, &ObjDescrOutputModule, EntryDescrOutputModule, &DataOutputModule],
[OD_IDX_END, NULL, NULL, NULL] /* end of dictionary */
/*****
FUNCTION IMPLEMENTATION
*****/
/**
* The function ethercat_sync_signal_callback() is called by the SDAI when a synchronization
* signal is generated.
*
* @return
* - type : VOID
* @param[in] pSyncSignal Pointer to the received synchronization signal structure
* - type : struct SDAI_SYNC_SIGNAL*
* - range: whole address range
*
1411 行 57 桁 CRLF 3B SJIS


TwinCAT Project1 - Microsoft Visual Studio
File Edit View Project Build Debug TwinCAT PLC Tools Scope Window Help
Release
TwinCAT RT (x86)
<Local>
Solution Explorer
TwinCAT Project1
Solution 'TwinCAT Project1'
TwinCAT Project1
SYSTEM
MOTION
PLC
SAFETY
C++
I/O
Devices
Image
Image-Info
SyncUnits
Inputs
Outputs
InfoData
Mappings
Sync Manager:
PDO List:
SM Size Type
0 128 MbxO...
1 128 MbxIn
2 4 Outpu...
3 4 Inputs
PDO Assignment (0x1C13):
Change Link...
Clear Link(s)
Goto Link Variable
Take Name Over from linked Variable
Add New Item... Ctrl+Shift+A
Insert New Item...
Insert Existing Item...
Delete Del
Move Address...
Online Write...
Online Force...
Release Force
Add to Watch
Remove from Watch
Name
32Bit Input 42
WcState 0
InputToggle 1
State
AdsAddr
netId
port
32Bit Output 0
Error List
0 Errors
0 Warnings
0 Messages
Clear
Creating project 'TwinCAT Project1'... project creation successful.

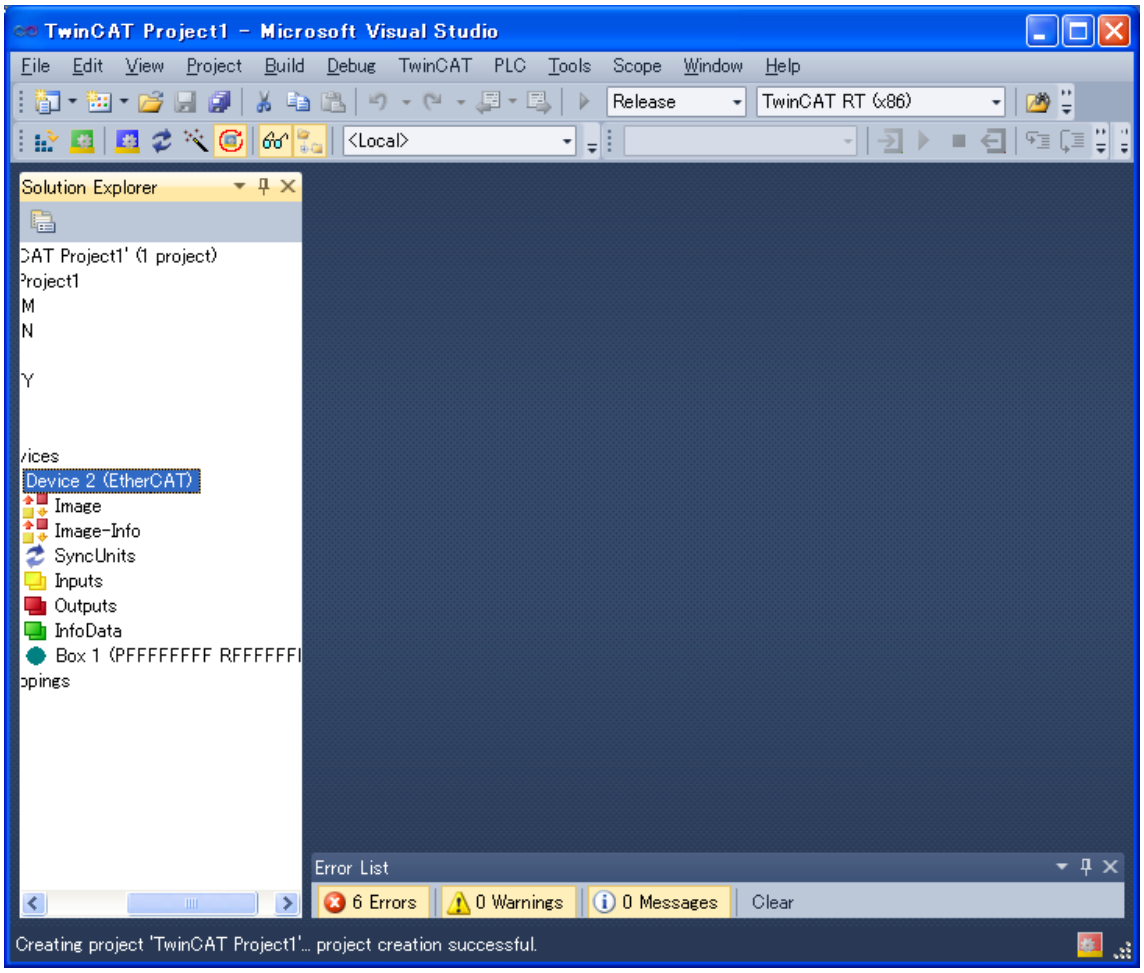

Set Value Dialog
Dec: 5
Hex: 0x00000005
Float: 7.0064923e-045
Bool: 0 1
Binary: 05 00 00 00
4
Bit Size: 1 8 16 32 64 ?
OK
Cancel
Hex Edit...


COM12:115200baud - Tera Term VT
ファイル(F) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) ヘルプ(H)
ECAT_EVENT_NEW_OUTPUT_DATA
--- EVENT_OUTPUT_DATA_CHANGED
Output Unit:
ID: 0x00011601
Size: 0x04
Status: 0x80
Data: 0x05 0x00 0x00 0x00
The Sherpa application provides standard output which can be received on a PC using a terminal software such as Tera Term or PuTTY.
PLC output is printed in the console
Solution Explorer
Solution 'TwinCAT Project16'
TwinCAT Project16
SYSTEM
MOTION
PLC
SAFETY
C++
I/O
Devices
Image
SyncUnits
Inputs
Outputs
InfoData
Mappings
Name
Online
Type
Size
>Addr...
In/Out
32Bit Input
12
UDINT
4.0
39.0
Input
WcState
0
BIT
0.1
1522.1
Input
netId
169.254.56.73.3.1
AMSNETID
6.0
1550.0
Input
port
0x03e9
WORD
2.0
1556.0
Input
32Bit Output
5
UDINT
4.0
39.0
Output
Error List
Creating project 'TwinCAT Project16'... project creation successful.

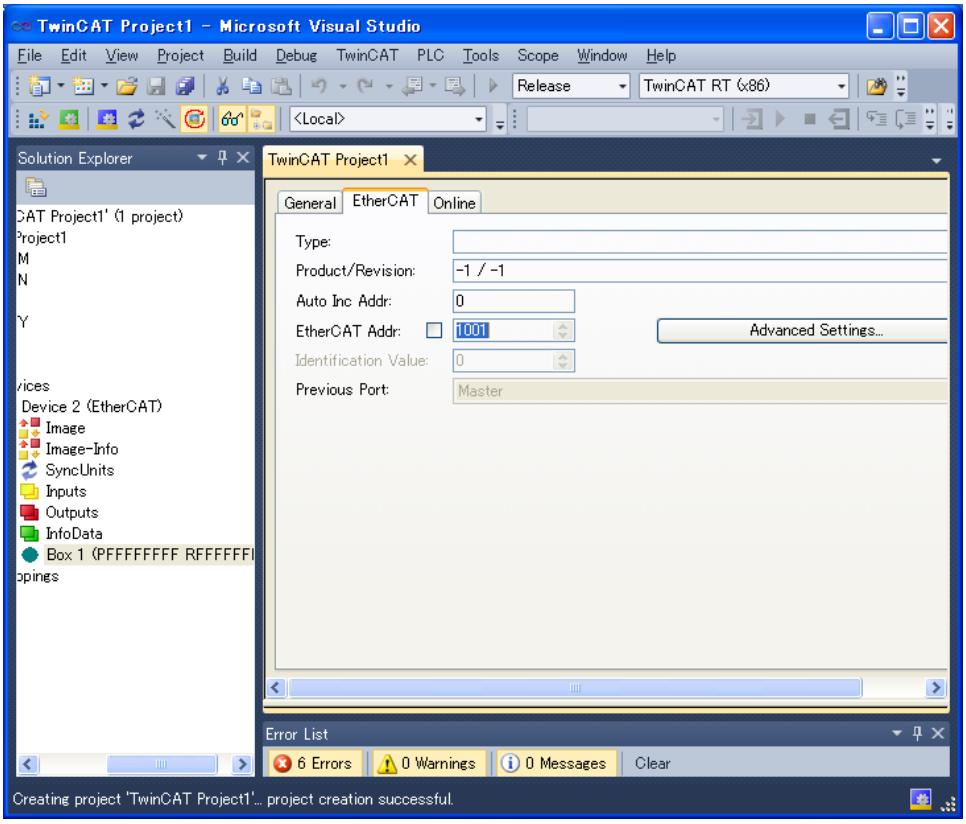

LE1
LE2
LE3
LE4
LE5
LE6
LE7
LE8

Programming the EEPROM for the ESC (EtherCAT Slave Controller) for the first time:

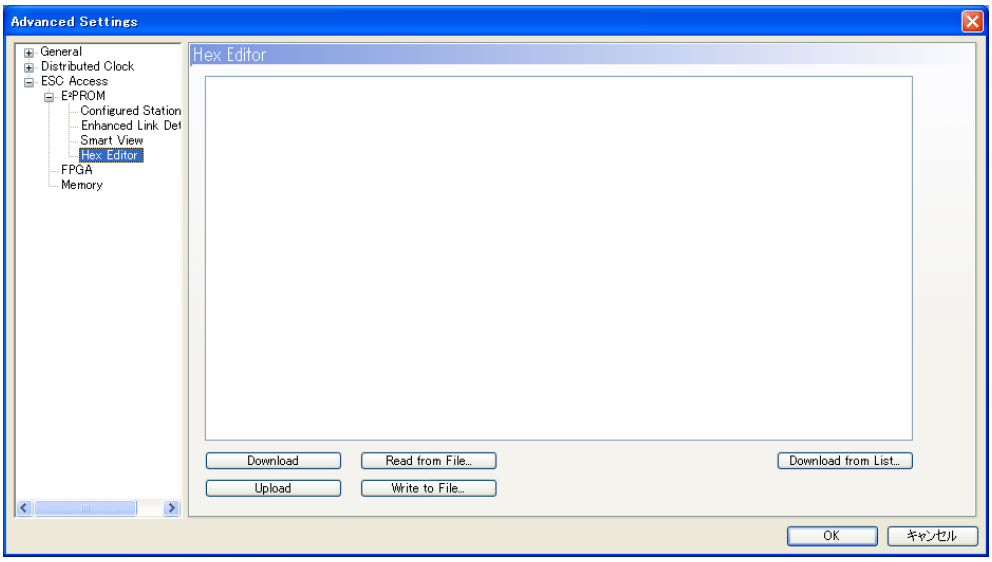
The ESC implemented in the R-IN32M3-EC needs to have the configuration for the Sherpa sample application programmed in its EEPROM which is provided in binary EtherCAT Terminal Configuration format. This needs to be done only once, and as long as the configuration of the ESC does not change, this configuration remains in EEPROM even after power cycle. There are different tools available to write the EEPROM. The steps below explain the procedure using TwinCAT®. In the TwinCAT® project select the box for the sample application.



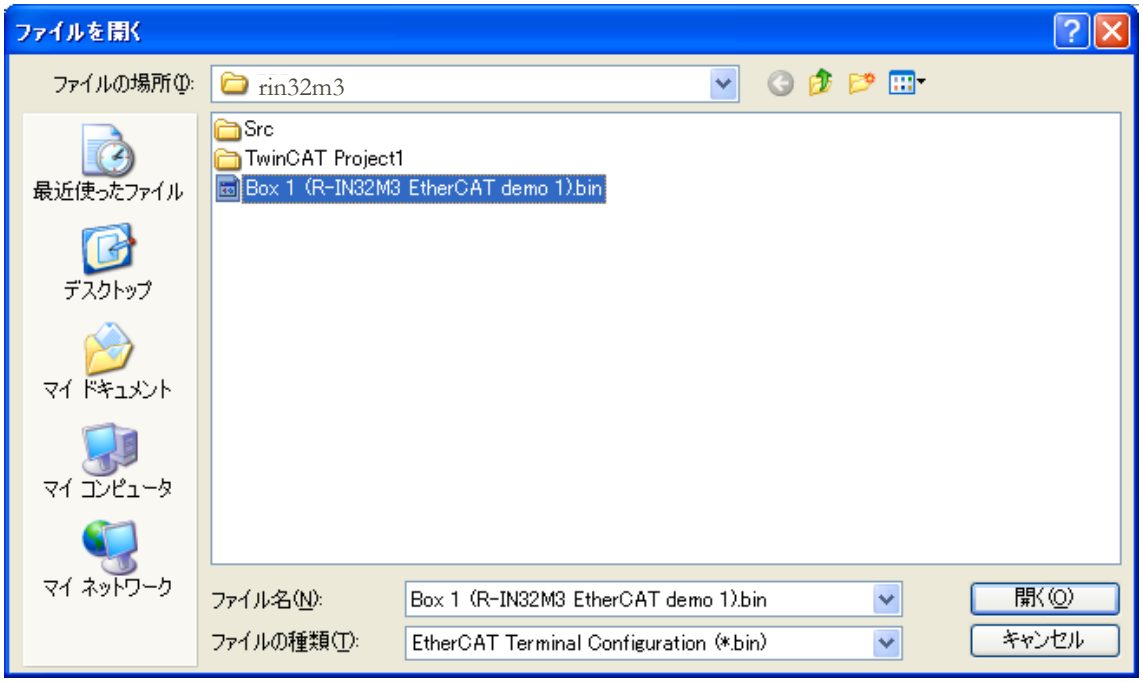
As shown below, for Box 1 select the “Advanced Setting” tab.

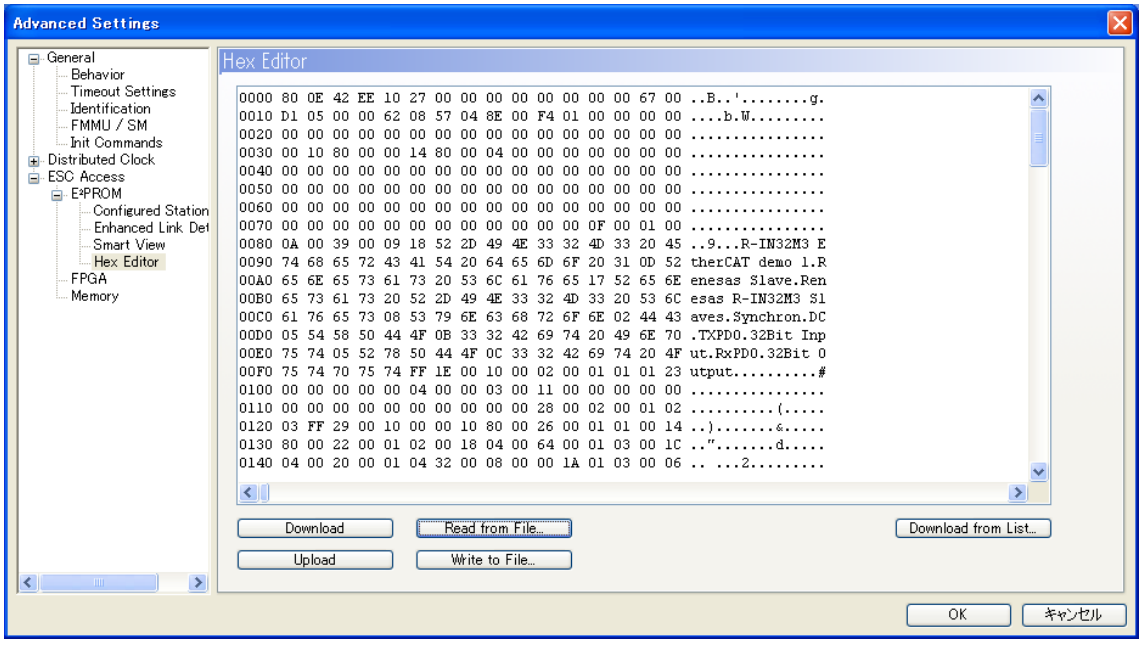


Select ESC Access→Hex Editor button and then click “Read from file” button:



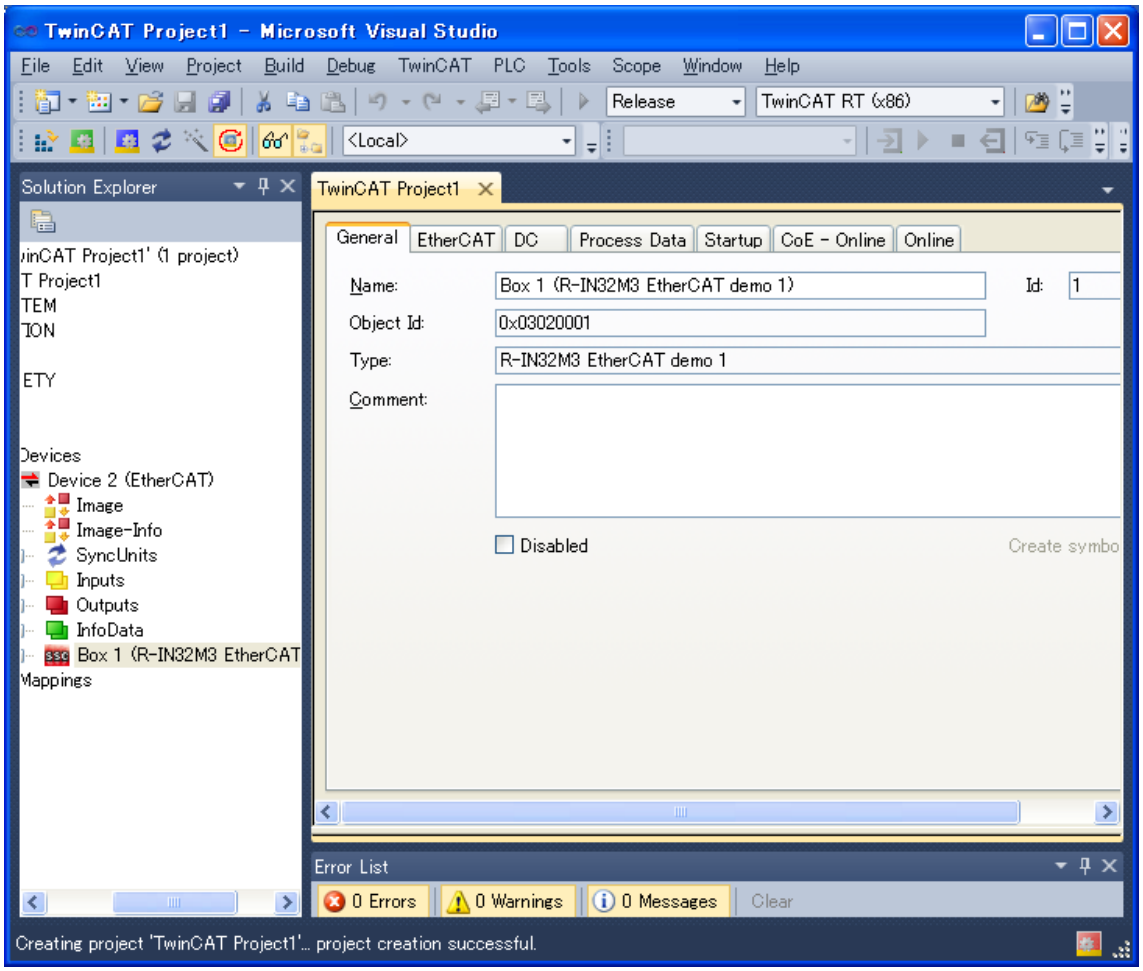
Select the file “Box 1(R-IN32M3 EtherCAT demo 1).bin” provided with this Sherpa LLC’s EtherCAT slave evaluation kit





After clicking the “Download” button above, the EEPROM will be programmed.

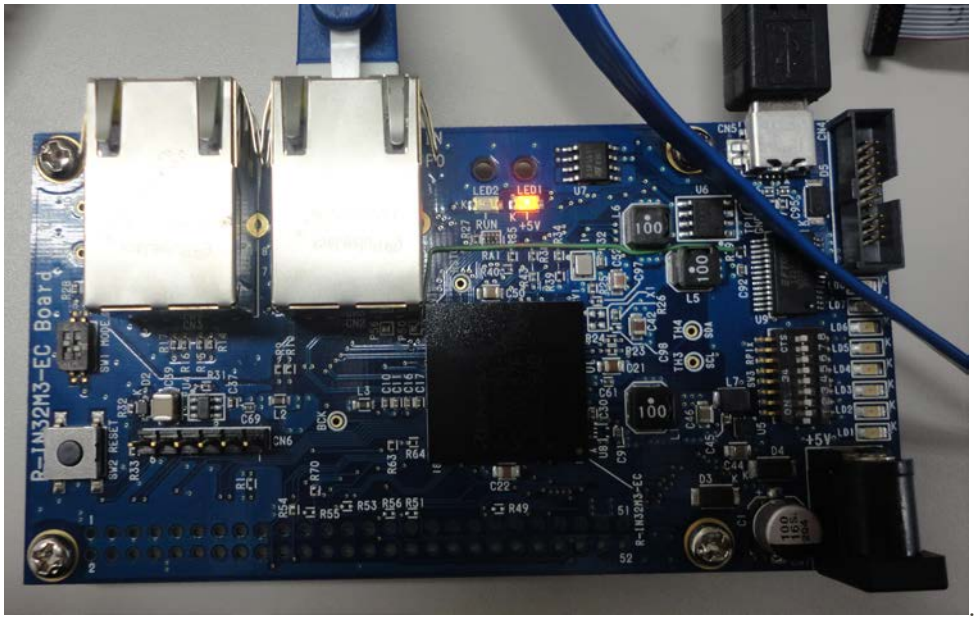
When TwinCAT® is restarted and a scan is performed the below detection screen is shown:



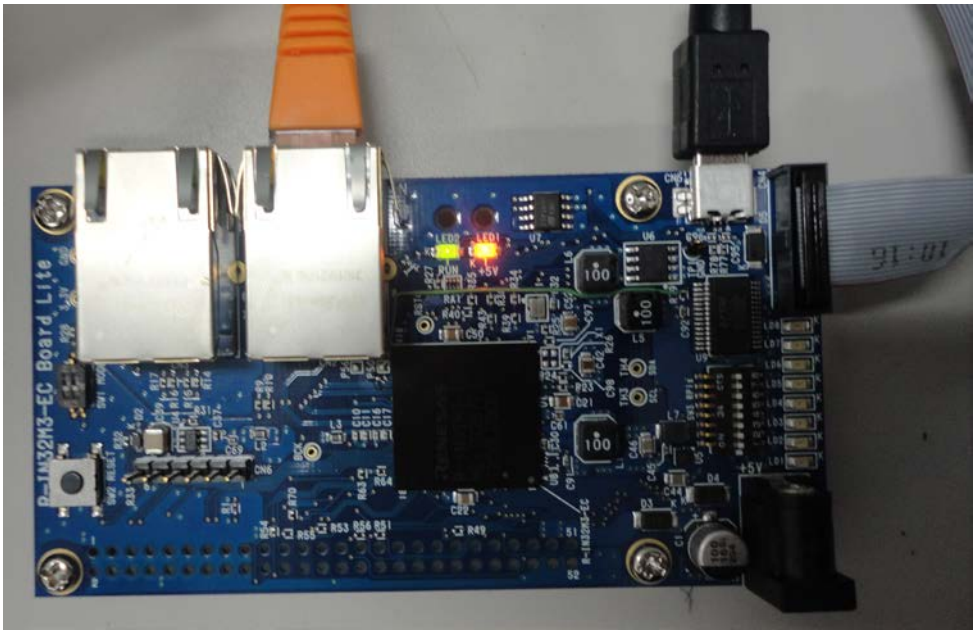
12. Sherpa EtherCAT application's external interface in IAR System board

This section provides a description for the Sherpa sample EtherCAT device applications external interface on the IAR System evaluation board. The external interface consist of light emitting diode for output and status representation and DIP switch for input to the EtherCAT master of Sherpa application operation.

IAR board after power on prior to sample application going into RUN mode:
LED1 status is solid amber (orange)



IAR board after sample application goes into run mode
and TwinCAT® status is Operational: LED2 status is solid green.



13. Licensing, product development and additional services

The Sherpa LLC's EtherCAT slave evaluation kit allows industrial device manufacturers to develop devices that conform to the EtherCAT standard in a very short time and with minimum involvement in the communication protocol management, which is done by the Sherpa library. The use of this library in production requires a licensing contract between the device manufacturer and Sherpa LLC. When this agreement is reached Sherpa will provide release library customized to the vendor's specific board. Customization services can include access library porting to external application processor when R-IN32M3 is used as a communication co-processor. Additionally, consulting services for measurement application development can be considered as part of consulting services package.

For information about licensing and consulting services, please contact Sherpa LLC at:

Sherpa LLC
Office #8, 3rd floor, Kase Building
88 3-19-11 Shin-Yokohama, Kohoku-ku
TEL 050-5532-6257
r-in32-stack@sherpa-tech.jp

14. EtherCAT slave stack functionality

Functionality	Full Slave Device (Data Link Layer) with two Ethernet Ports	
Performance Characteristics	FMMUs	3 (configurable)
	Sync Managers	4 (configurable)
	EtherCAT State Machine	Supported
	Distributed Clocks	Supported
	Generation of Synchronous Outputs	(SYNC0/1)
	CAN Application Protocol Over EtherCAT(CoE)	Supported
	Object Dictionary with Standard CoE Objects	• Device Type • Manufacturer Device Name • Manufacturer Hardware Version • Manufacturer Software Version • Identity • RxPDO Mapping (up to 32 instances) • TxPDO Mapping (up to 32 instances) • Sync Manager Communication Type • Sync Manager 2 PDO Assignment • Sync Manager 3 PDO Assignment
	SDO Upload/Download, SDO Information Services	Supported
	Complete SDO Access	Supported
	Ethernet Over EtherCAT (EoE)	Under development