Application Note EL9800

Version 1.7
Date: 2018-07-11

BECKHOFF

LEGAL NOTICE

DOCUMENT HISTORY

Version	Comment
0.1	Documentation based on EL9800_2 (PIC18)
0.2	Add EtherCAT piggyback notes Additional Instructions for PIC24 Evaluation board (EL9800_4A) Note for EL9800 / FB1111-014x setup
1.0	Add Section II Chapter 2 “Slave Sample Code”
1.1	Update “create project” instructions Update file references Section II is moved to Application Note ET9300 Add chapter “Testing Slave Sample Code”
1.2	Update document structure (guideline to configure slaves) Update file references
1.3	Editorial changes due to new naming in SSC 5.10 Update MPLAB 8 screenshot
1.4	Update compiler reference (changed from C30 to XC16) Update screenshots Update EL9800 related EEPROM update settings
1.5	Add MPLAB X and EL9800_6 related information
1.6	editorial changes
1.7	Move obsolete (related to old Evaluation boards) Information to the appendix section, reference ETG dowuments how to setup the EtherCAT master and program the EEPROM

CONTENTS

FIGURES

Table

ABBREVIATIONS

DC	Distributed Clocks
EEPROM	Electrically Erasable Programmable Read Only Memory
I/O	Input/Output
ICD	In Circuit Debugger
IDE	Integrated Development Environment
NIC	Network Interface Card
OS	Operating System
PDI	Process Data Interface
PIC	Peripheral Interface Controller
PICC	PIC Compiler
PDO	Process Data Object
RT	Real Time
SDO	Service Data Object
SII	Slave Information Interface
SM	Sync Manager
SPI	Serial Peripheral Interface
SSC	Slave Stack Code

1 GENERAL NOTES

This document is a guideline to start working with the EL9800 EtherCAT Evaluation board. Workshops and Trainings referred to the Slave Development and Slave Stack Code are listed in the event section on ETG website (http://www.ethercat.org).
The software and hardware used to create this manual are listed in Table 1 and Table 2.

Table 1: Used Software

Software	Name	Version	Comment
Code	Beckhoff EtherCAT Slave Code	5.12	http://www.ethercat.org/memberarea/stack_code.aspx
Compiler	Microchip XC16	1.35	https://www.microchip.com
IDE	MPLAB X	3.40	http://www.microchip.com
OS EtherCAT Master	Microsoft Windows TwinCAT	10 3.1 Build 4022.16	http://www.beckhoff.com

Table 2: Used Hardware

Hardware	Name	Version	Comment
Evauation board	EL9800	6 (PIC24)	http://www.beckhoff.com
EtherCAT piggyback controller	FB1111-0142	-	http://www.beckhoff.com

2 EL9800 revision

The actual revision is **EL9800_6** (Figure 1: EL9800_6 EtherCAT Evaluation board). The type of the board is also printed in the lower right corner.

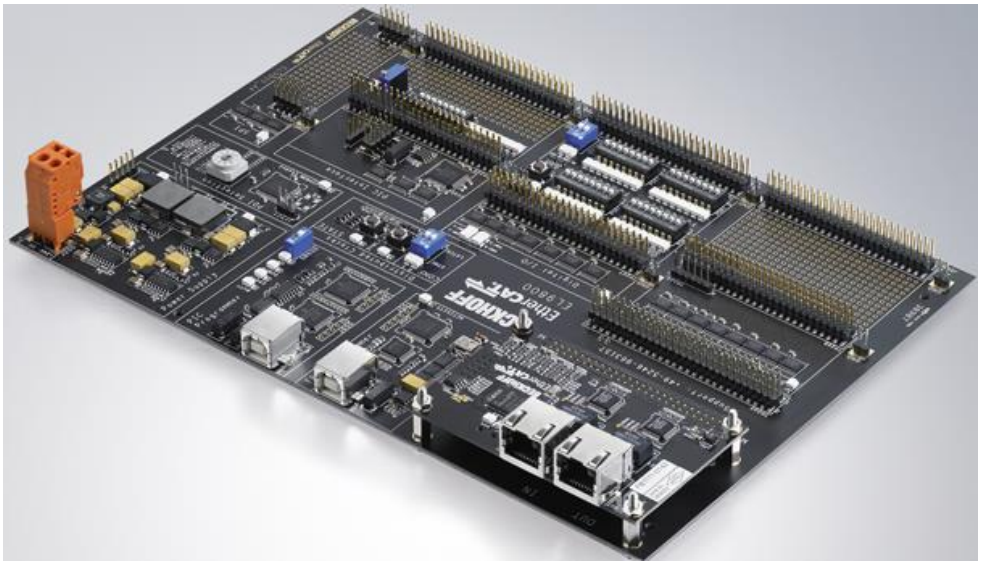

Figure 1: EL9800_6 EtherCAT Evaluation board

A detailed description and pinout of the board is available in the EL9800 datasheet.
https://www.beckhoff.com/english.asp?download/ethercat_development_products.htm?id=71003127100387

3 TwinCAT (EtherCAT Master)

How to install and operate with TwinCAT is described in the document “Set-Up a Network Configuration” (download: https://www.ethercat.org/memberarea/download/HowTo_SetUpNetworkConfiguration.pdf)

Note: Login to the ETG member area is required (https://www.ethercat.org/en/membership_application.html)

4 Slave Configuration

This chapter describes how to create an EtherCAT slave configuration using TwinCAT and the EL9800 Evaluation board.

4.1 Digital I/O Slave

The ET1100 and ET1200 provide the possibility to handle up to 32 (ET1100) digital signals without a connected local uController (and slave software). These slaves are called simple devices. This chapter describes how to configure such an EtherCAT slave.
NOTE: Only the piggyback board FB1111-0142 shall be used in this configuration.

a. Power off the EL9800 Evaluation board

b. Set PDI selector to position 0

c. Power on the EL9800 Evaluation board

d. Create a TwinCAT Project

e. Scan the network (chapter **Error! Reference source not found.**)

f. Write the Device Description for 16 Bit Digital I/O to the EEPROM (Figure 2: 16Bit Digital I/O Device Description). How to write the EEPROM is described in the EtherCAT Knowledgebase www.ethercat.org/KB).

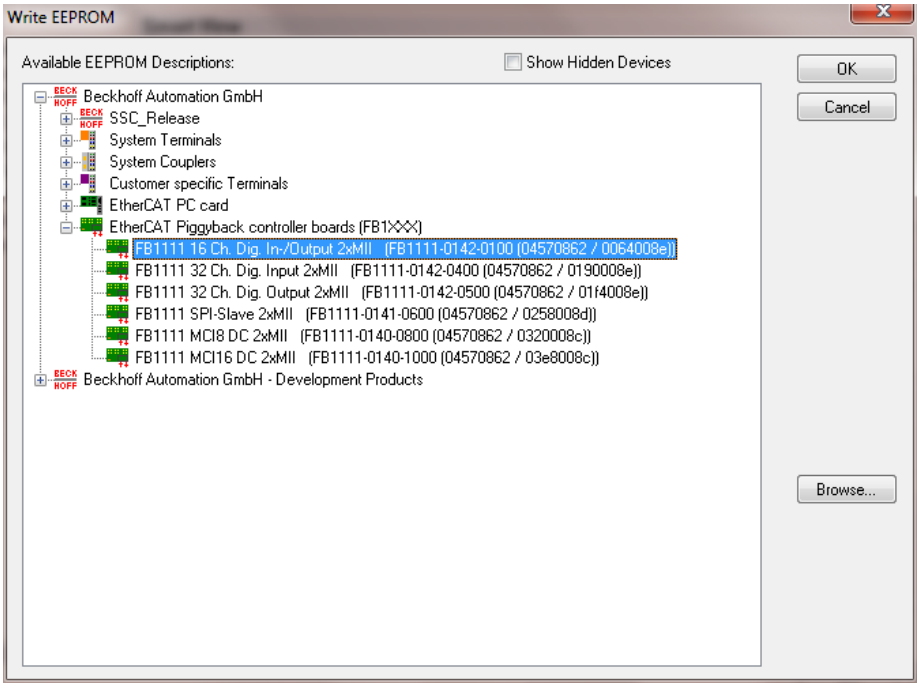

Figure 2: 16Bit Digital I/O Device Description

g. After writing and verification was successful close the “Advanced Settings” Dialog

h. Power off the EL9800 evaluation board

i. Set PDI selector to position 4

j. Power on the EL9800 evaluation board

k. Rescan for EtherCAT slaves (right mouse click on “Device xx (EtherCAT)”-> “Scan Boxes”

l. If a “Configuration changed” dialog is shown click “Copy all” and acknowledge this dialog with “OK”.

m. Activate “Free Run” (“Free Run” can also be activated by pressing Ctrl+F5)

n. Process data communication is now running.
Outputs can be set by right mouse click on the variable and selecting “Online write”.

4.2 Standard Slave

The standard EtherCAT slave includes an uController connected to the ESC which handles the EtherCAT related software stack. In this example the Slave Stack Code is used as the EtherCAT slave software.
The Slave Stack Code is free of charge a can be downloaded here.

How to create a slave project is described in the following chapter.

5 Slave Stack Code Project

5.1 Create Project

a. Create a working folder (e.g. “c:\working\SSC\src”) and copy the SSC source file to that folder. The source file are created with the SSC Tool (see Application Note ET9300) or located in the SSC download zip archive.
b. Open the MPLAB X and click [File] → [New Project] in the menu bar.

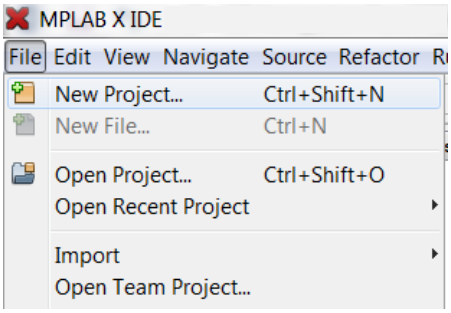

Figure 3: Create a new MPLAB X Project

c. Wizard steps
   a. Choose Project: “Standalone Project”
   b. Device Type: PIC24HJ128GP306
   c. Select Tool: Other Tools -> Licensed debugger-> “EL9800 PICKit OnBoard Programmer”
   d. Select Compiler: XC16
   e. Select Project Name and Folder:

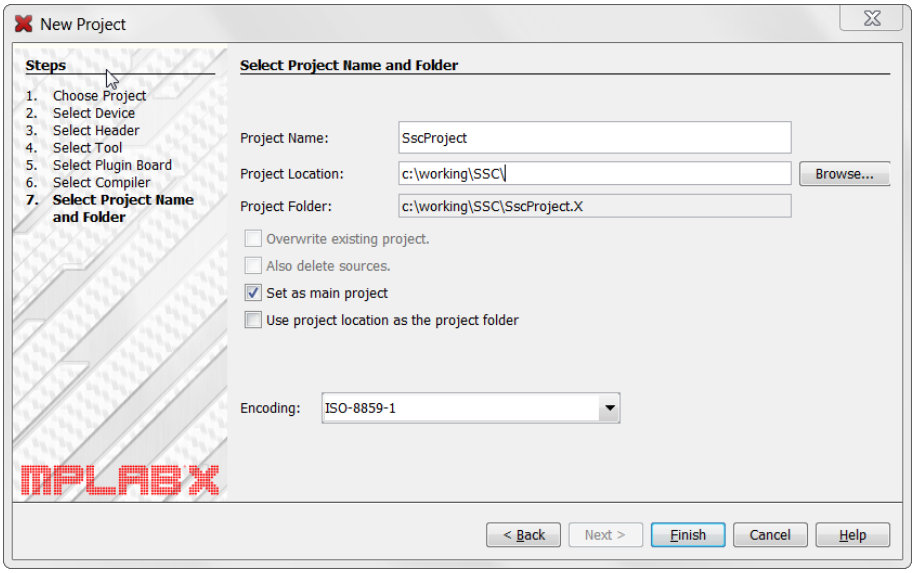

Figure 4: MPLAB X Sample Project name and folder

d. Open the context menu of the “Header Files” node, select “Add Existing Item …” and add all .h files
e. Open the context menu of the “Source Files” node, select “Add Existing Item …” and add all .c files

5.2 Microchip XC16 Compiler specific setting

f. Define a head size (e.g. 1000 bytes)
Open the Project context menu -> Properties -> xc16-ld.

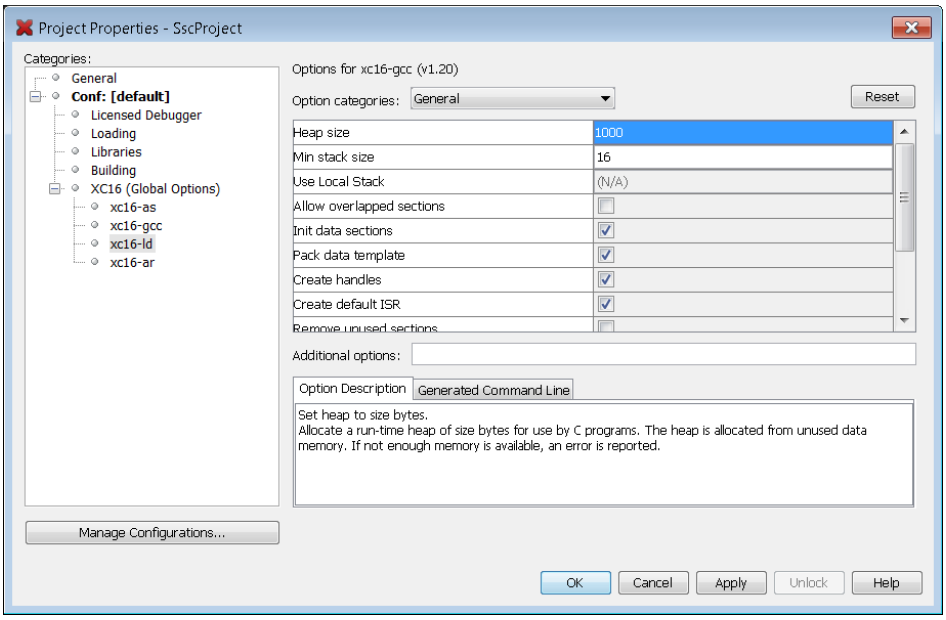

Figure 5: Heap setting for Microchip XC16 compiler

a. To compile the SSC select [Run] → [Build Main Project] in menu bar.

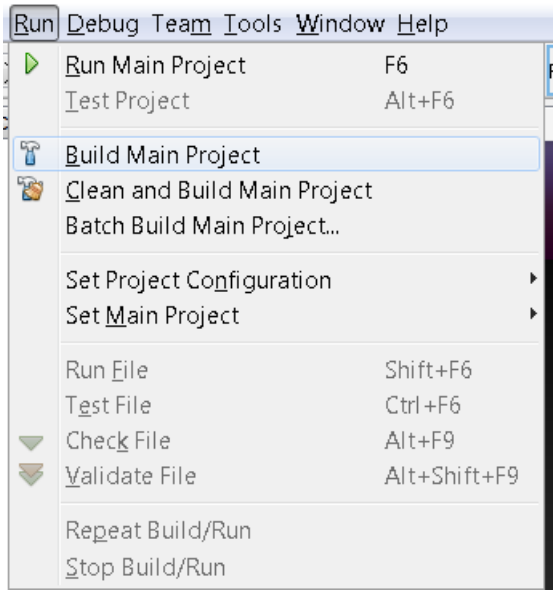

Figure 6: Rebuild Project

5.3 Download binary

5.3.1 Debugger

The EL9800_6 supports two PIC debugger interfaces. The first one is fixed connected to onboard PICKit debugger (communication channel 3) and the second one is connected to the “open” interface on J1005 (communication channel 2). The In-Circuit Debugger register need to be configured depending on the desired interface.

The register is set in *el9800hw.c*. (Selectable by define “EXT_DEBUGER_INTERFACE”)

- fixed connected debugger: **_FICD(ICS_PGD3 & JTAGEN_OFF);**
- “open” interface: **_FICD(ICS_PGD2 & JTAGEN_OFF);**

The following instructions refer to the fixed connected onboard PICKit debugger.

g. Enable the on board debugger interface. Set dipswitch SW600.
h. Select Debug -> “Debug Main Project”

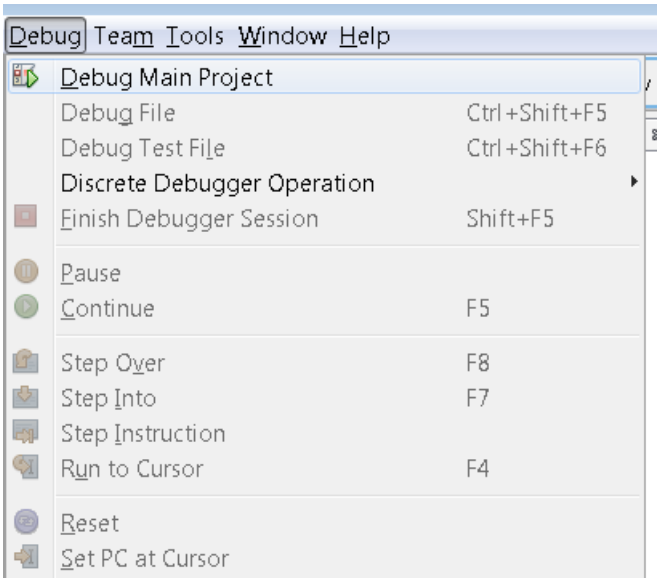

Figure 7: MPLAB X Debug Main Project

Appendix A

Previous Board Versions

EL9800_4A is similar to the revision EL9800_6 except of the PIC Programmer. This board can't be used with MPLAB X (except a standalone programmer connected to J1005 is used).

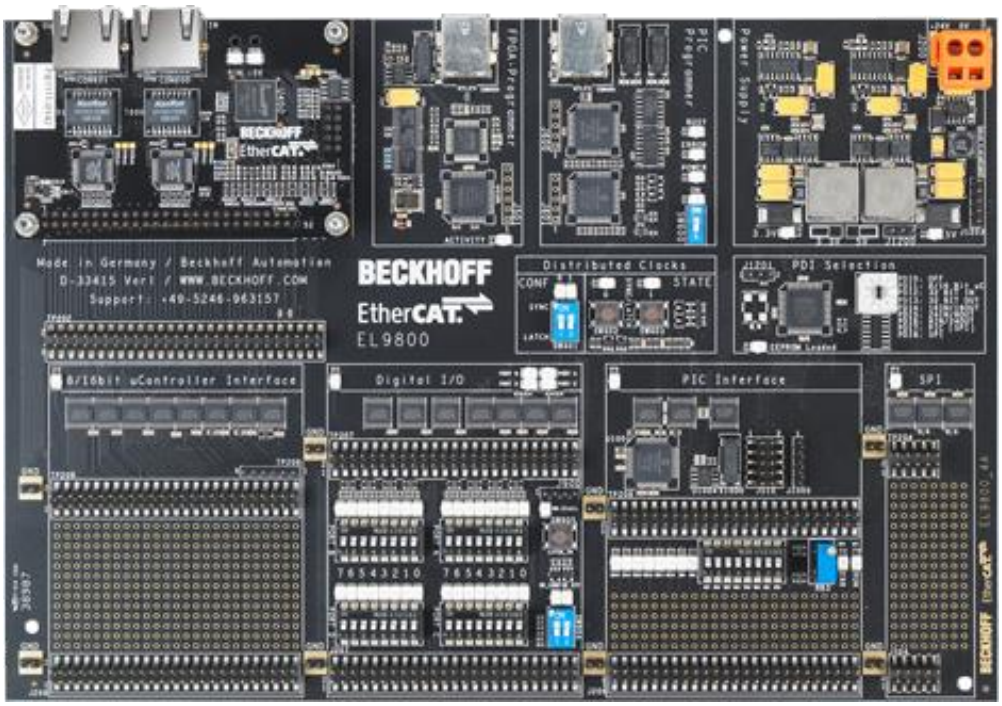

Figure 8: EL9800_4A EtherCAT Evaluation board

A detailed description and pinout of the board is available in the EL9800 datasheet.

The former revision of the board (**EL9800_2**) (**Error! Reference source not found.**) is no longer available.

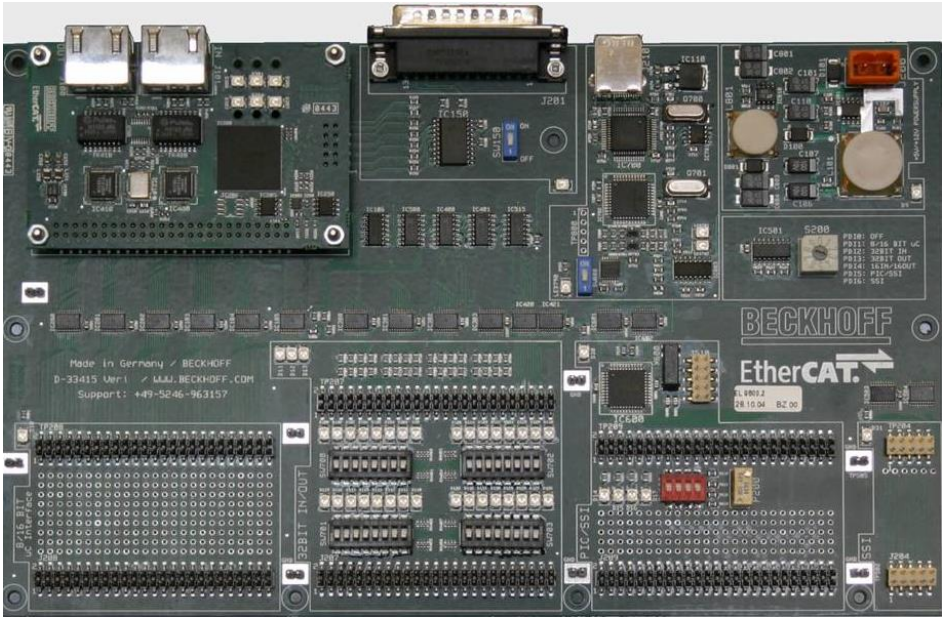

Figure 9: EL9800_2 EtherCAT Evaluation board

Slave Stack Code IDE Software

This chapter contains the list of the required PIC development software.

EL9800_2:

1. MPLAB 8
2. HI-TECH PICC-18 STD compiler
3. MPLAB ICD2 Debugger driver

EL9800_4A:

1. MPLAB 8
2. Microchip XC16 compiler
3. MPLAB ICD2 Debugger driver

MPLAB IDE 8

Download the latest MPLAB IDE 8.x from www.microchip.com

a. Run the Setup

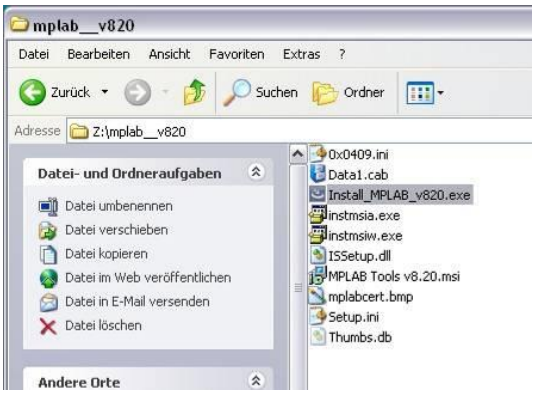

Figure 10: Execute Installation file

b. Select the “Complete” setup type
(To handle the SSC not all components from the MPLAB IDE are required but in the first move it’s recommended to install the complete package.)

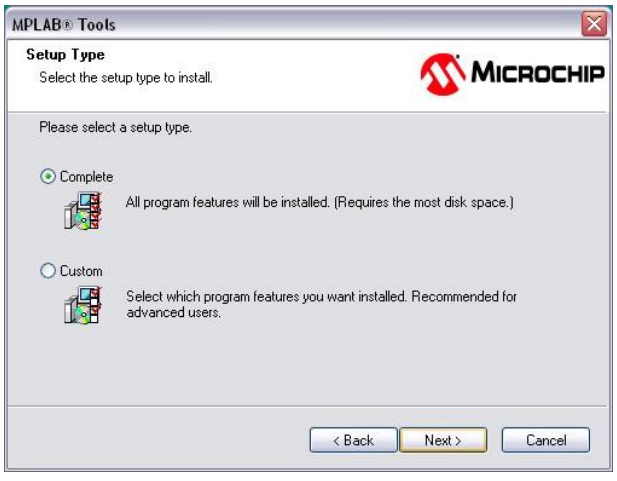

Figure 11: Select setup type

Don’t install the “HCPIC18-pro-960PL5” compiler

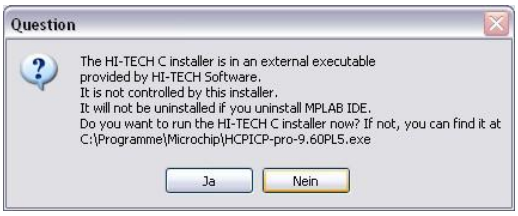

Figure 12: Compiler installation

Microchip XC16 compiler

This compiler is required for the PIC24 mounted on the **EL9800_4A** EtherCAT Evaluation board.

a. Download the Microchip XC16 free compiler from the Microchip homepage
b. Execute the setup (the installation routine is self-explanatory)

MPLAB ICD2 driver

a. Connect the USB-cable with the evaluation board and the PC
b. Set SW600 (>= EL9800.4) or SW800 (<= EL9800.2) to enable the onboard debugger interface
c. Power on the evaluation board
d. A new device is detected on the PC
e. Select “Automatic” installation

Figure 13: Driver installation

f. If the driver was not successfully installed please reinstall the driver.
Driver location: “c:\Program Files\Microchip\MPLAB IDE\ICD2\Drivers\”

Slave Stack Code Project in MPLAB 8

This chapter introduces how to create, download and run a local slave application using the Slave Stack Code and the MPLAB 8 IDE. The PIC Programmer/Debugger instructions are referring to the MPLAB ICD2 programmer (EL9800_4A).

Create Project

a. Create working folder (e.g. “c:\SSC_410\SPI\SRC”) and copy the SSC source file to that folder.
The source file are created with the SSC Tool (see Application Note ET9300) or located in the SSC download zip archive.
b. Open the MPLAB 8 and click [Project] → [New…] in the menu bar.

Figure 14: Create a new MPLAB 8 Project

c. Enter a name and the path where you want to store the project file and all other created files. Then press the [OK] button.

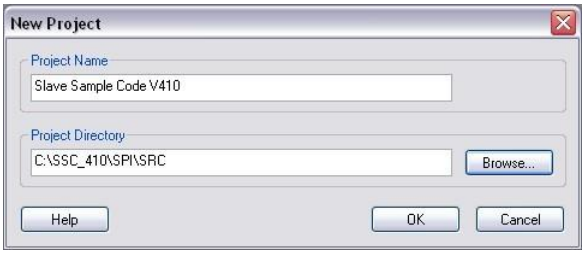

Figure 15: Project Details

d. If the project manager is not already displayed in MPLAB choose [View] → [Project] in menu bar.

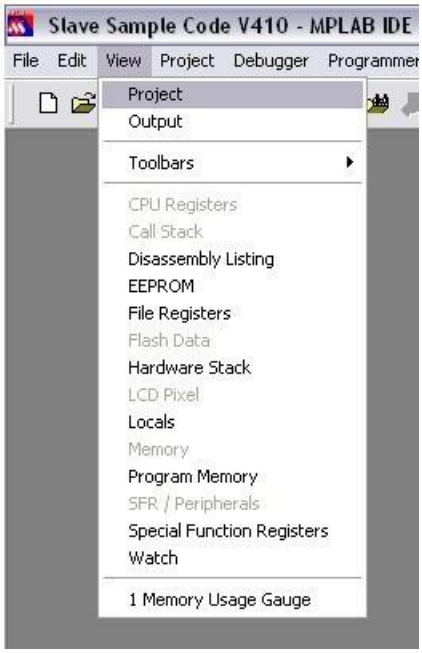

Figure 16: Activate Project Manager

e. To add the SSC source files select [Project] → [Add Files to Project…] in menu bar.

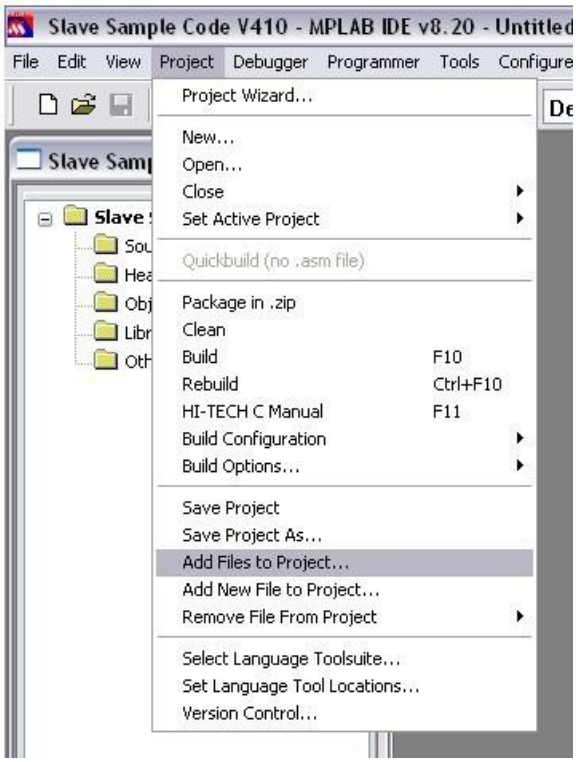

Figure 17: Add source files

f. Select **all *.c and *.h files**. Then press [Open] button.

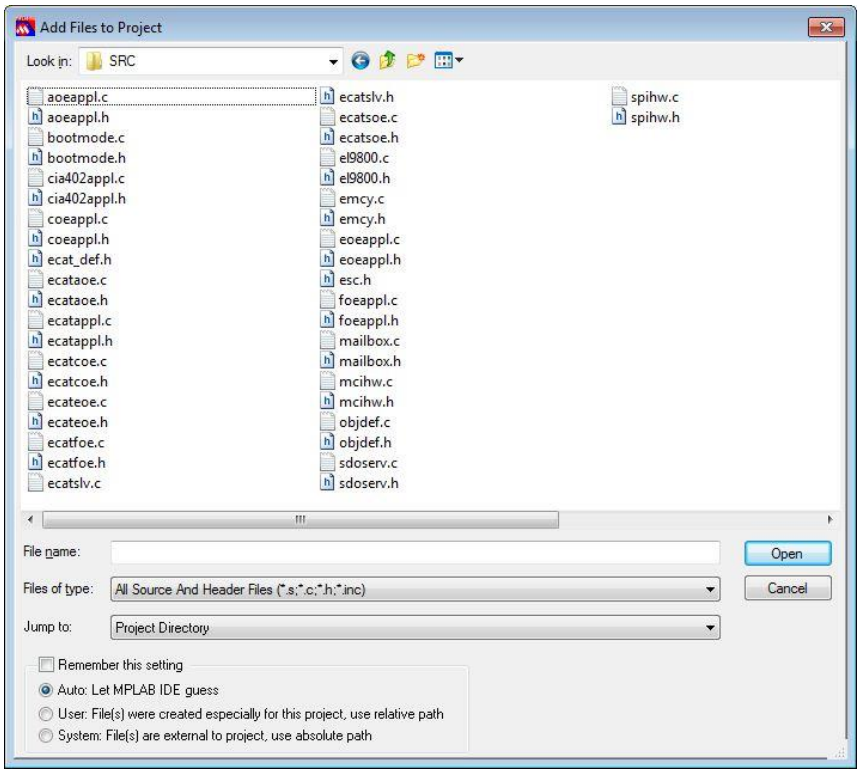

Figure 18: Select source files

Project settings

a. To select the compiler, linker and assembler choose [Project] → [Select Language Toolsuite…] in menu bar.

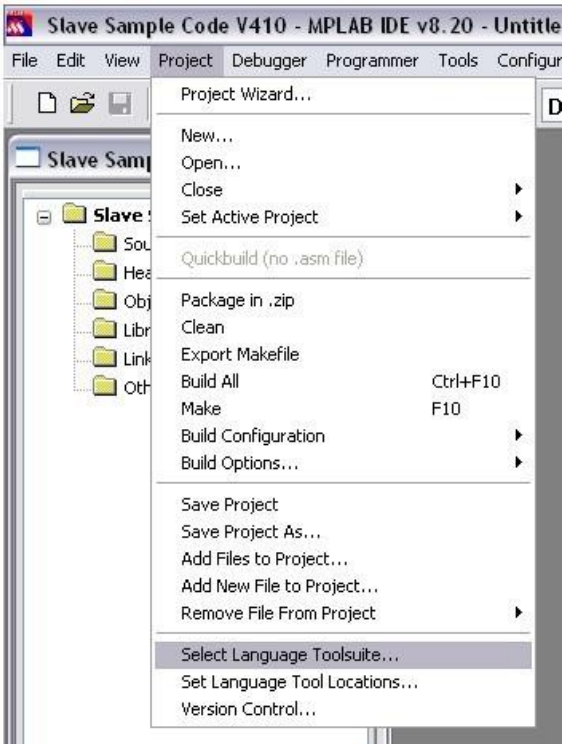

Figure 19: Select tools

b. Choose the corresponding Toolsuite
   - <= EL9800_2: “HI-TECH PICC-18 Toolsuite”.
   - >= EL98004A: “Microchip XC16 Toolsuite”

c. If necessary correct the path in the location field.

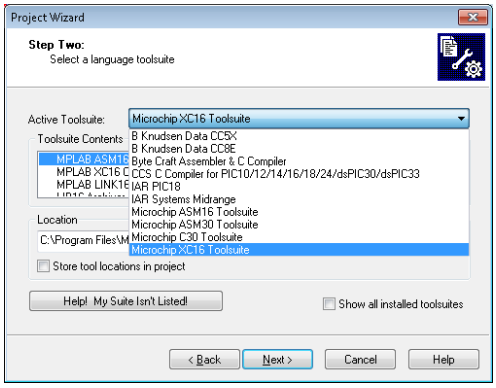

Figure 20: Microchip XC16 Toolsuite

a. Select the corresponding PIC for the EtherCAT slave platform. Therefore click [Configure] → [Select Device…].
   For evaluation boards up to and including version EL9800_2 select “**PIC18F452**”.
   For evaluation boards from version EL9800_4A select “**PIC24HJ128GP306**”

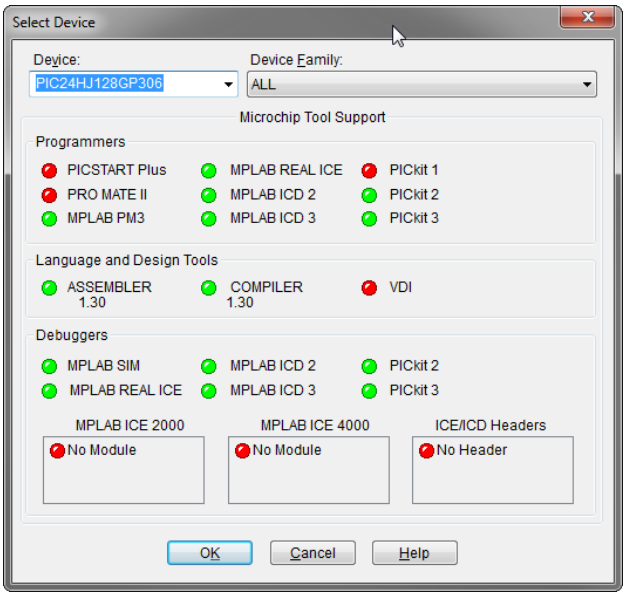

Figure 21: Selection of the controller

Compiler/PIC specific settings

- **Microchip XC16** compiler (PIC24):
  Define a head size (e.g. 4096 bytes) (Project->"Build Options"->Project-> tab "MPLAB LINK30").

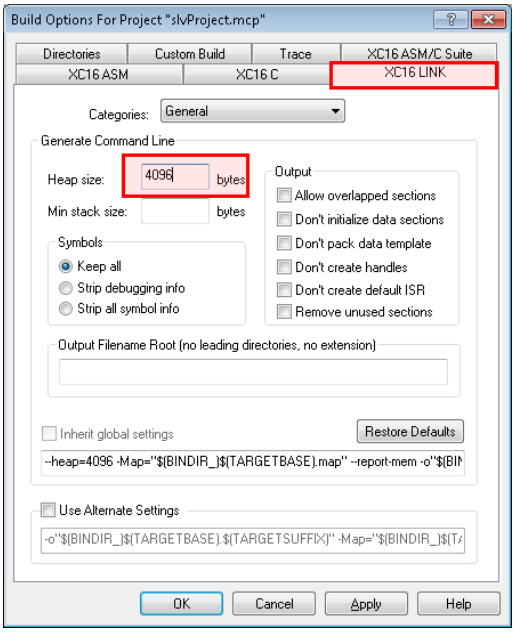

Figure 22: Heap setting for Microchip XC16 compiler

- **HI-TECH PICC-18 STD** compiler (PIC18):
  Increment the “identifier length” > 60 and disable code optimization ([Project] → [Build Options] → [Project] → [Compiler] tab).

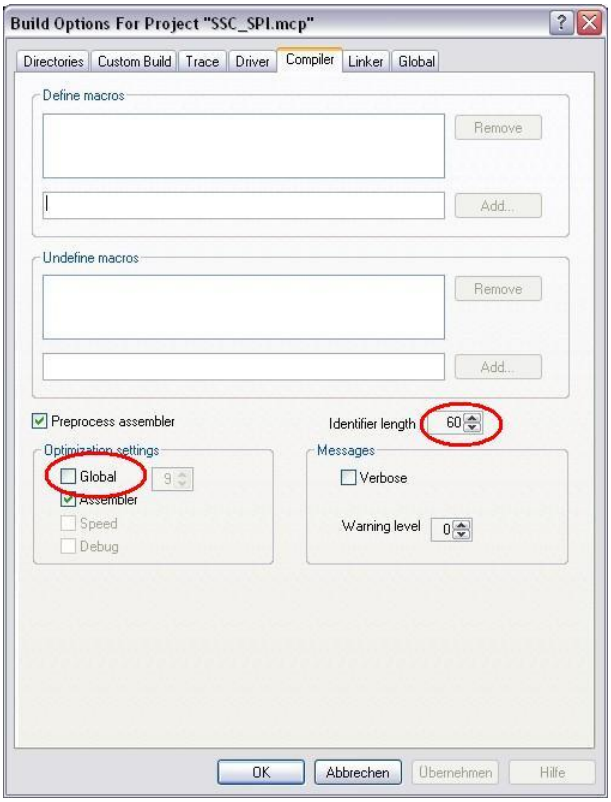

Figure 23: Compiler Settings

b. Confirm that the correct “PIC define” is set. These defines are located in “ecat_def.h”.

For EL9800 hardware up to version 2 select _**PIC18.**
It is not possible to get full feature setup for the PIC18 in addition to other reasons caused by limited program memory. So it is necessary to check which features are required for the desired EtherCAT slave. The corresponding defines are located in “ecat_def.h”.
For EL9800 hardware from version 4 select _**PIC24.**

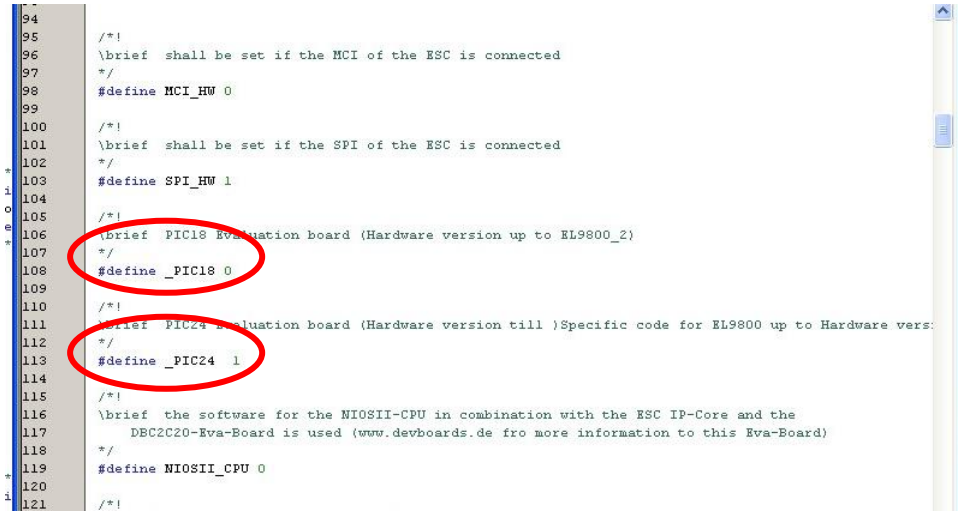

Figure 24: PIC define

c. To compile the SSC select [Project] → [Rebuild] in menu bar.
(Since MPLAB version 8.46 “Rebuild” is renamed to “Build All”)

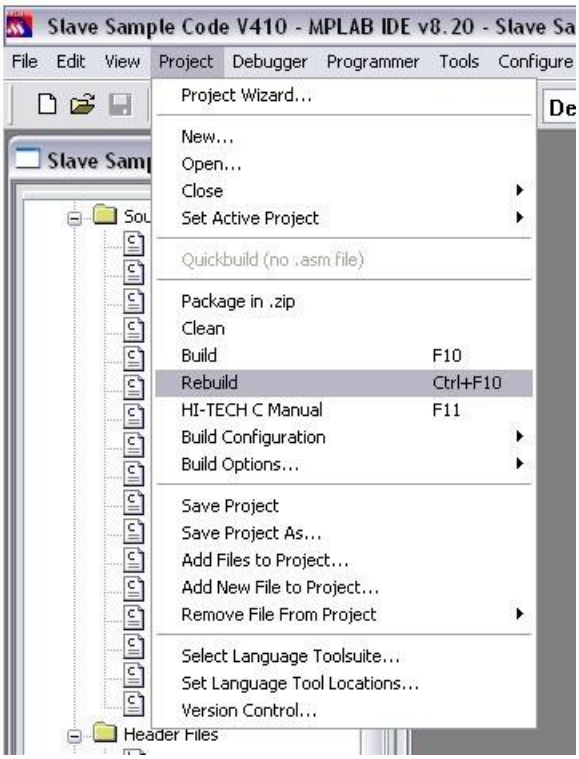

Figure 25: Rebuild Project

An output window with further information appears. The *.hex and *.cof (for debugging) files are created in the project folder.

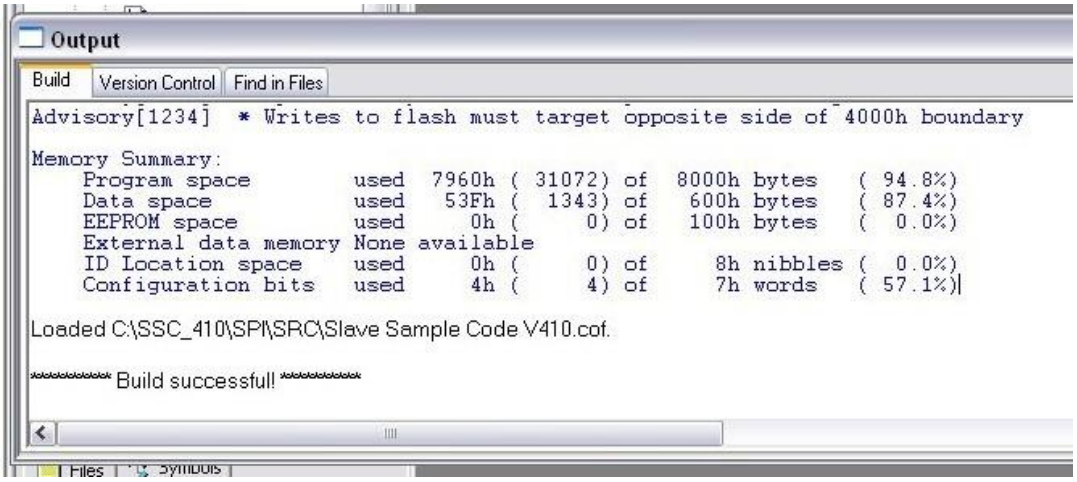

Figure 26: Build succeed output

Download binary

Debugger

The Evaluation Kit from hardware version 4 supports two PIC debugger interfaces.The first one is fixed connected to onboard ICD 2 debugger (communication channel 3) and the second one is connected to the “open” interface on J1005 (communication channel 2). The In-Circuit Debugger register need to be configured depending on the desired interface.

The register is set in *el9800hw.c*. (Selectable by define “EXT_DEBUGER_INTERFACE”)

- fixed connected debugger: **_FICD(ICS_PGD3 & JTAGEN_OFF);**
- “open” interface: **_FICD(ICS_PGD2 & JTAGEN_OFF);**

The following instructions refer to the fixed connected onboard ICD 2 debugger.

a. Enable the on board debugger interface. Set dipswitch SW600 (>= EL9800.4) or SW800 (<=EL9800.2) to “on”
b. Select MPLAB ICD2 Debugger (All EL9800 EtherCAT development boards contain a MPLAB ICD 2 onboard debugger)

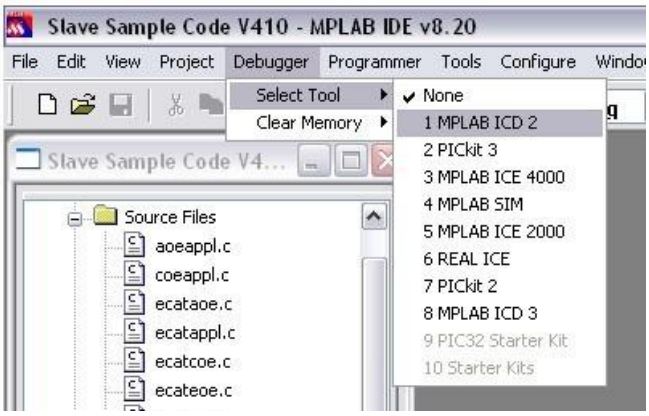

Figure 27: Select "MPLAB ICD 2"

c. The MPLAB ICD2 setup wizard start up

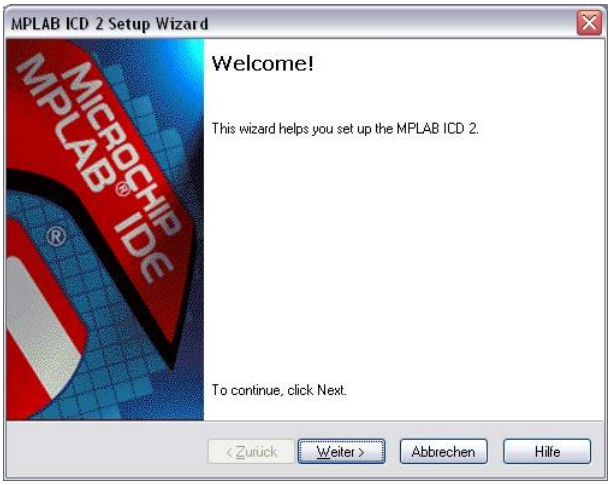

Figure 28: ICD 2 setup wizard

d. Select “USB”

Figure 29: Communication interface selection

e. Select „Target has own power supply“

Figure 30: Power supply selection

f. Don't check automatic connection

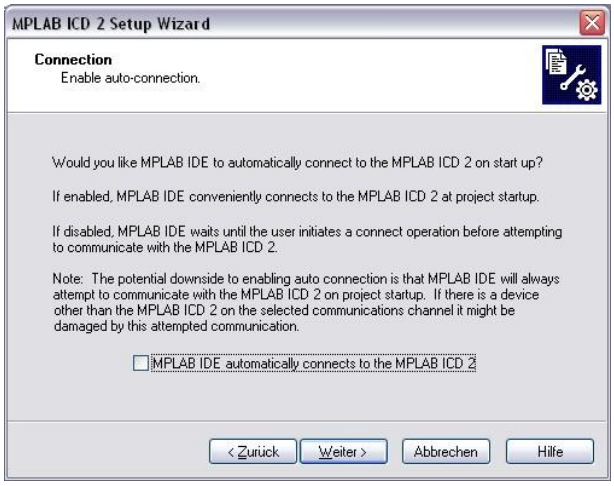

Figure 31: Auto connect

g. Check automatic download operating system

Figure 32: Download OS

h. Connect to debugger

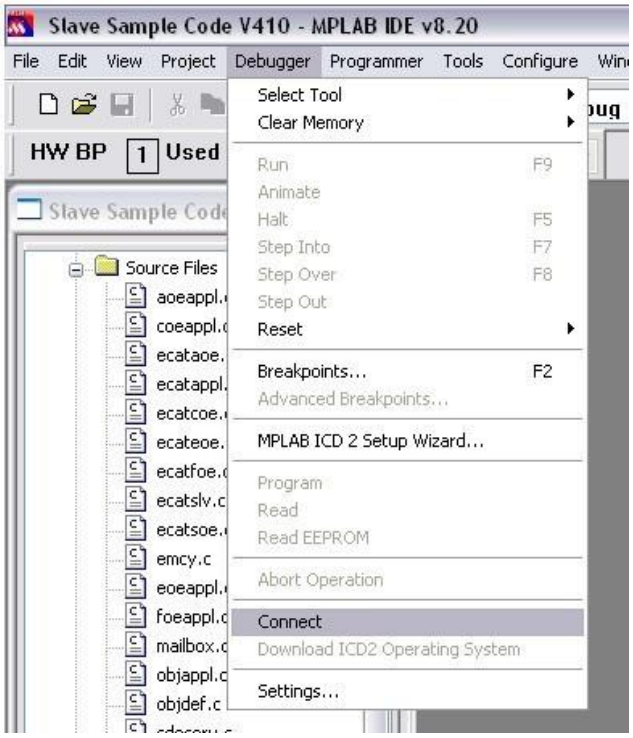

Figure 33: Connect to debugger

i. Acknowledge the download warning with [OK]

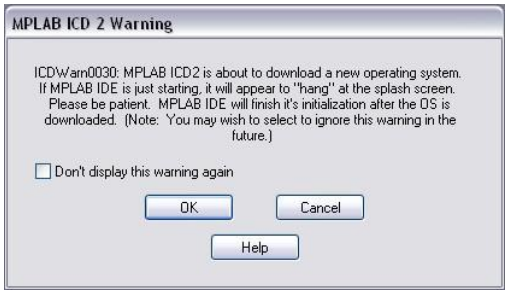

Figure 34: Download warning

j. Output if connection succeeds

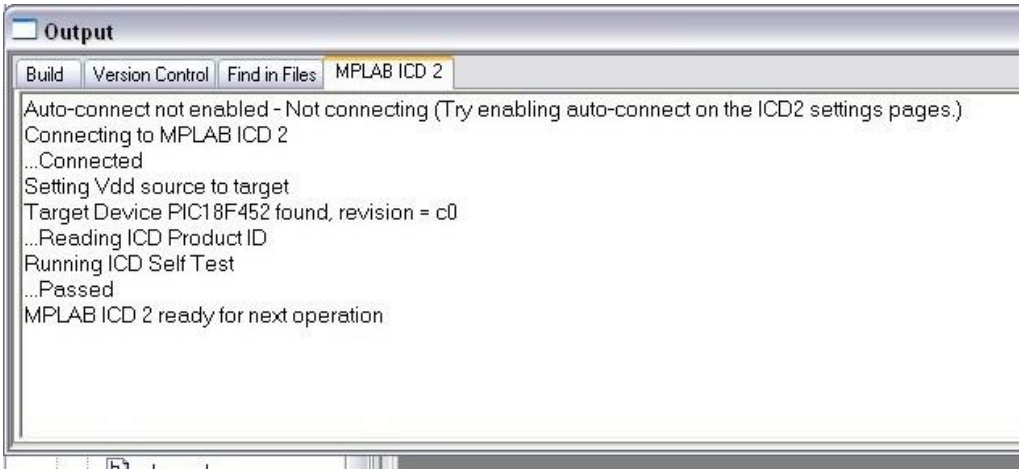

Figure 35: Output window: connection successful

k. Download the binary file

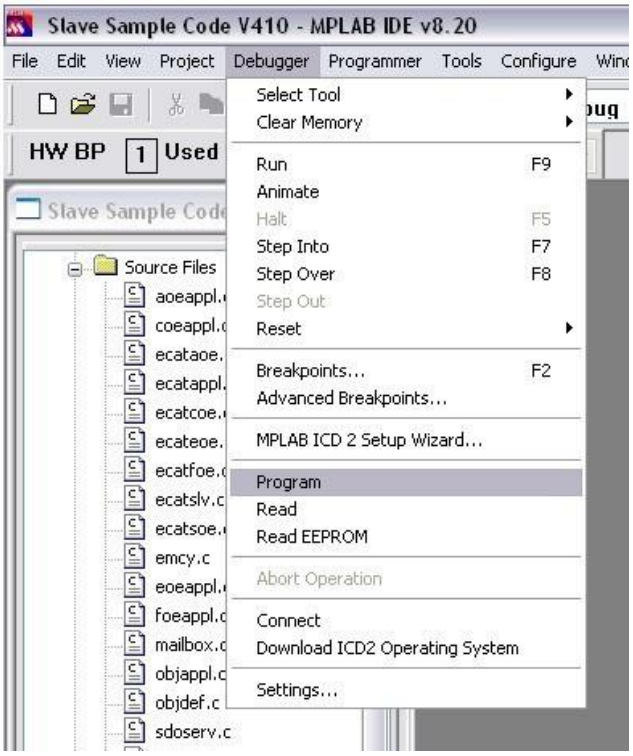

Figure 36: Program PIC memory

Output if programming succeeds

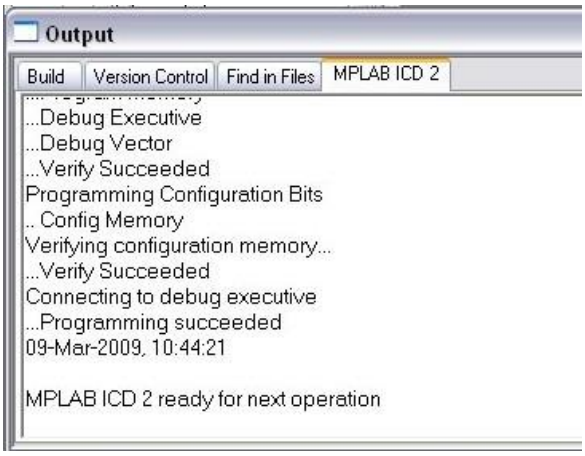

Figure 37: Output window: Programming successful

I. Select [Debugger] → [Run]

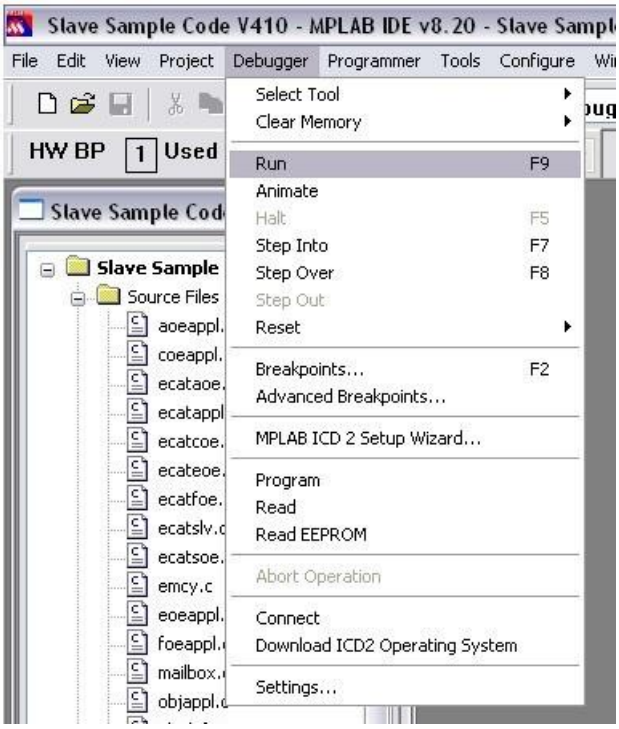

Figure 38: Run binary

Appendix B

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
Eiserstr. 5
33415 Verl
Germany
phone: + 49 (0) 5246/963-0
fax: + 49 (0) 5246/963-198
e-mail: info@beckhoff.com
web: www.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:
world-wide 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

EtherCAT Technology Group (ETG) Headquarters

Phone: +49 (911) 540 5620
Fax: +49 (911) 540 5629
Email: info@ethercat.org
Internet: www.ethercat.org