# **Application Note EL9800**



Version 1.7 Date: 2018-07-11



LEGAL NOTICE

#### Trademarks

Beckhoff®, TwinCAT®, EtherCAT®, Safety over EtherCAT®, TwinSAFE® and XFC® are registered trademarks of and licensed by Beckhoff Automation GmbH. Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners. Patent Pending

The EtherCAT Technology is covered, including but not limited to the following German patent applications and patents: DE10304637, DE102004044764, DE102005009224, DE102007017835 with corresponding applications or registrations in various other countries.

Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development. For that reason the documentation is not in every case checked for consistency with performance data, standards or other characteristics. In the event that it contains technical or editorial errors, we retain the right to make alterations at any time and without warning. No claims for the modification of products that have already been supplied may be made on the basis of the data, diagrams and descriptions in this documentation. Copyright

© Beckhoff Automation GmbH 05/2009.

The reproduction, distribution and utilization of this document as well as the communication of its contents to others without express authorization are prohibited. Offenders will be held liable for the payment of damages. All rights reserved in the event of the grant of a patent, utility model or design.

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

# DOCUMENT HISTORY

# CONTENTS

1	GENERAL	NOTES	6
2	EL9800 rev	vision	7
3	TwinCAT (	EtherCAT Master)	8
4	Slave Conf	figuration	9
	4.1	Digital I/O Slave	9
	4.2	Standard Slave	10
5	Slave Stac	k Code Project	11
	5.1	Create Project	11
	5.2	Microchip XC16 Compiler specific setting	11
	5.3	Download binary	12
		5.3.1 Debugger	12
Appe	endix A		14
•••	Prev	vious Board Versions	14
	Slav	e Stack Code IDE Software	14
		MPLAB IDE 8	15
		Microchip XC16 compiler	16
		MPLAB ICD2 driver	16
	Slav	e Stack Code Project in MPLAB 8	16
		Create Project	16
		Project settings	19
		Download binary	22
Арре	endix B		28
	Sup	port and Service	28
	Beck	<pre>khoff's branch offices and representatives</pre>	28
	Beck	<pre>khoff Headquarters</pre>	28
	Beck	<pre>khoff Support</pre>	28
	Beck	<pre><hoff pre="" service<=""></hoff></pre>	28
	Ethe	erCAT Technology Group (ETG) Headquarters	28

# FIGURES

Figure 1: EL9800 6 EtherCAT Evaluation board	. 7
Figure 18: 16Bit Digital I/O Device Description	. 9
Figure 49: Create a new MPLAB X Project	11
Figure 50: MPLAB X Sample Project name and folder	11
Figure 51: Heap setting for Microchip XC16 compiler	12
Figure 52: Rebuild Project	12
Figure 53: MPLAB X Debug Main Project	13
Figure 1: EL9800_4A EtherCAT Evaluation board	14
Figure 2: EL9800_2 EtherCAT Evaluation board	14
Figure 20: Execute Installation file	15
Figure 21: Select setup type	15
Figure 22: Compiler installation	16
Figure 23: Driver installation	16
Figure 24: Create a new MPLAB 8 Project	17
Figure 25: Project Details	17
Figure 26: Activate Project Manager	17
Figure 27: Add source files	18
Figure 28: Select source files	18
Figure 29: Select tools	19
Figure 30: Microchip XC16 Toolsuite	19
Figure 31: Selection of the controller	20
Figure 32: Heap setting for Microchip XC16 compiler	20
Figure 33: Compiler Settings	21
Figure 34: PIC define	21
Figure 35: Rebuild Project	22
Figure 36: Build succeed output	22
Figure 37: Select "MPLAB ICD 2"	23
Figure 38: ICD 2 setup wizard	23
Figure 39: Communication interface selection	23
Figure 40: Power supply selection	24
Figure 41: Auto connect	24
Figure 42: Download OS	24
Figure 43: Connect to debugger	25
Figure 44: Download warning	25
Figure 45: Output window: connection successful	25
Figure 46: Program PIC memory	26
Figure 47: Output window: Programming successful	26
Figure 48: Run binary	27

# Table

Table 1: Used Software	6
Table 2: Used Hardware	6

# ABBREVIATIONS

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

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/me mberarea/stack_code.aspx
Compiler	Microchip XC16	1.35	https://www.microchip.com
IDE	MPLAB X	3.40	http://www.microchip.com
OS Ett. CAT	Microsoft Windows	10	
EtherCA I Master	TWINCAT	3.1 Build 4022.16	nttp://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. <u>https://www.beckhoff.com/english.asp?download/ethercat\_development\_products.htm?id=710031271</u> <u>00387</u>

# 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 (<u>https://www.ethercat.org/en/membership\_application.html</u>)

# 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"
- I. 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 <u>here</u>.

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]  $\rightarrow$  [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:

🗙 New Project				X
Steps	Select Project Name	e and Folder		
1. Choose Project 2. Select Device 3. Select Header 4. Select Tool	Project Name:	SscProject		
5. Select Plugin Board 6. Select Compiler	Project Location:	c:\working\SSC\	Br	owse
7. Select Project Name	Project Folder:	c:\working\SSC\SscProject.X		
	Overwrite existing	g project.		
	Also delete source	es.		
	📝 Set as main proje	ect		
	Use project locati	on as the project folder		
	1			
//////	Encoding: ISO-885	i9-1 <b>v</b>		
//////	9			
<b>RPLAB</b>	C			
		< <u>B</u> ack Next >	Finish Cancel	<u>H</u> elp

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.

- ⊙ General	Options for xc16-gc	: (v1.20)		
Conf: [default]	Option categories:	General	•	Rese
<ul> <li>Licensed bebugger</li> <li>Loading</li> </ul>	Heap size		1000	
Libraries	Min stack size		16	
Building     XC16 (Global Options)	Use Local Stack		(N/A)	
	Allow overlapped se	ections		
• xc16-gcc	Init data sections			
<ul> <li>xc16-id</li> <li>xc16-ar</li> </ul>	Pack data template			
	Create handles			
	Create default ISR			
	Remove unused sec	tions		
	Additional options:			
	Option Description	Generated Command Line		
	Set heap to size by Allocate a run-time memory. If not eno	tes. heap of size bytes for use by C ugh memory is available, an erri	programs. The heap is allocated from unused ar is reported.	data
Manage Configurations				

Figure 5: Heap setting for Microchip XC16 compiler

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

<u>R</u> un	Debug Tea <u>m</u> Tools	<u>W</u> indow <u>H</u> elp
	<u>R</u> un Main Project	F6
	<u>T</u> est Project	Alt+F6
r	<u>B</u> uild Main Project	
8	<u>C</u> lean and Build Main Batch Build Main Proj	Project ect
	Set Project Co <u>n</u> figura Set <u>M</u> ain Project	tion 🕨
	Run <u>F</u> ile	Shift+F6
	T <u>e</u> st File	Ctrl +F6
$\bigtriangledown$	Chec <u>k</u> File	Alt+F9
$\triangleleft$	<u>V</u> alidate File	Alt+Shift+F9
	Repeat Build/Run	
	<u>S</u> top Build/Run	

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"

Deb	ug Tea <u>m T</u> ools <u>W</u> indow <u>H</u> elp	
85	<u>D</u> ebug Main Project	,
	Debug File	Ctrl+Shift+F5
	Debug Test Fi <u>l</u> e	Ctrl+Shift+F6
	Discrete Debugger Operation	۱.
	Einish Debugger Session	Shift+F5
0	Pause	
	<u>C</u> ontinue	F5
ß	Step O <u>v</u> er	F8
\$	Step <u>I</u> nto	F7
	Step Instruction	
< <u> 1</u>	R <u>u</u> n to Cursor	F4
0	Reset	
-1	Set PC at Cursor	

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

 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
- 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]  $\rightarrow$  [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.

Project Name	
Slave Sample Code V410	
Project Directory	
C:\SSC_410\SPI\SRC	Browse

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.

Edit View	Project	Debugger Programm	er Tools	Configur
) 📽 🖬 🛛	Proje	ct Wizard		
	New.			
Slave Sam	Орег	h		
-	Close			- <b>F</b>
Slave :	Set A	ctive Project		•
- 🛄 Sou - 🦲 Hea	Quick	build (no .asm file)		
🛄 Obj	Packa	age in .zip		
🛄 Libr	Clear	1		
🛄 Oth	Build		F10	
	Rebu	ild	Ctrl+F1	5
	HI-TE	CH C Manual	F11	
	Build	Configuration		•
	Build	Options		•
	Save	Project		
	Save	Project As		
	Add F	Files to Project		
	Add I	New File to Project		
	Remo	ove File From Project		•
	Selec	t Language Toolsuite	1	_
	Set L	anguage Tool Locations.		
	Versi	on Control		

Figure 17: Add source files

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

Add Files to Project			×
Look in: 퉬 SRC	- 3 😰 🖾 -		
aoeappl.c	h] ecatsly.h	Spihw.c	
b aoeappl.h	ecatsoe.c	b spihw.h	
bootmode.c	h) ecatsoe.h		
bootmode.h	el9800.c		
cia402appl.c	h] el9800.h		
h cia402appl.h	emcy.c		
📄 coeappl.c	h] emcy.h		
🖞 coeappl.h	eoeappl.c		
h] ecat_def.h	h] eoeappl.h		
ecataoe.c	b] esc.h		
h] ecataoe.h	foeappl.c		
ecatappl.c	b) foeappl.h		
h] ecatappl.h	mailbox.c		
ecatcoe.c	h mailbox.h		
h] ecatcoe.h	mcihw.c		
ecateoe.c	h mcihw.h		
h ecateoe.h	objdef.c		
ecatfoe.c	h objdef.h		
h] ecatfoe.h	sdoserv.c		
ecatslv.c	b sdoserv.h		
•	III		
ile <u>n</u> ame:			Open
iles of type: All Source And He	ader Files (*.s;*.c;*.h;*.inc)	•	Cancel
lume ha			
Project Directory		•]	
Remember this setting			
Auto: Let MPLAB IDF quess			
C User: File(s) were created as	pecially for this project, use relative path		
Custem: File(s) were credited es	protect use absolute path		
<ul> <li>System, File(s) are external to</li> </ul>	project, use absolute path		

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**"

Select Device		
De <u>v</u> ice:	Device Eamily:	20
PIC24HJ128GP306	✓ ALL	•
	Microchip Tool Sup	port
Programmers		
PICSTART Plus	MPLAB REAL ICE	PICkit 1
PRO MATE II	MPLAB ICD 2	PICkit 2
MPLAB PM3	MPLAB ICD 3	PICkit 3
Language and Design Too	bls	
ASSEMBLER 1.30	COMPILER 1.30	VDI
Debuggers		
MPLAB SIM	MPLAB ICD 2	PICkit 2
MPLAB REAL ICE	MPLAB ICD 3	PICkit 3
MPLAB ICE 2000	MPLAB ICE 400	0 ICE/ICD Headers
No Module	🔴 No Module	No Header
	D <u>K</u> Cancel	Help

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").

Build Options For Project "slvProject.mcp	)"	? 💌
Directories Custom Build XC16 ASM XC1	Trace I6 C	XC16 ASM/C Suite XC16 LINK
Categories: General		•
Heap size: 4096 bytes	Output	understand sections
Min stack size: bytes	Don't i	nitialize data sections
Symbols	📃 Don't p	oack data template
Keep all	📃 Don't d	create handles
Strip debugging info	📃 Don't d	create default ISR
Strip all symbol info	Remov	ve unused sections
Output Filename Root (no leading dire	ectories, no ex	Restore Defaults
heap=4096 -Map="\$(BINDIR_)\$(TAR(	GETBASE).ma	ap"report-mem -o"\$(BI)
Use Alternate Settings Use Alternate Settings -o"\$[BINDIR_]\$[TARGETBASE].\$[TAR	GETSUFFIX)'	'-Map=''\$(BINDIR_)\$(T/
ОК	Cancel	Apply Help

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).

ectories Custom Build Trace Drive	r Compiler Linker Global
Define macros	
	Remove
Ī	Add
Undefine macros	
	Remove
	Add
Preprocess assembler	
Optimization settings	Messages
Speed	Warning level 0

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.** 

94		-
95	/*1	
96	\brief shall be set if the MCI of the ESC is connected	
97	*/	
98	#define MCI_HW 0	
99		
100	/*1	
101	\brief shall be set if the SPI of the ESC is connected	
102	*/	
103	#define SPI_HW 1	
1 104		
105	/*!	
106	(brief PIC18 Evaluation board (Hardware version up to EL9800_2)	
107	*/	
108	#define _PIC18 0	
109		
110	/*1	
111	iorief PIC24 Scaluation board (Hardware version till )Specific code for BL9800 up to Hardware vers	51
112	*/	
113	#define_PIC24 1	
114		
115	/*!	
116	\brief the software for the NIOSII-CPU in combination with the ESC IP-Core and the	
117	DBC2C20-Eva-Board is used (www.devboards.de fro more information to this Eva-Board)	
118	*/	
* 119	#define NIOSII_CPU 0	
i 120		
121	/*!	



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

le Edit View	Project	Debugger	Programmer	Tools	Configu
D 🚅 🖬	Proje	ct Wizard			
Slave Samı	New. Open Close Set A	  ctive Project			:
<u>ି</u>	Quick	build (no .asm	í file)		
	Packa Clear Build	age in .zip 1		F10	
9	Rebu	ild		Ctrl+F1	0
	HI-TE Build Build	CH C Manual Configuration Options		F11	ŀ
) ( ) ( ) ( ) ( ) ( ) ( ) ( ) ( ) ( ) (	Save Save Add f Add f Remo	Project Project As Files to Project New File to Pro New File From F	: Dject Project		•
	Selec Set L Versi	t Language To anguage Tool	oolsuite Locations		

Figure 25: Rebuild Project

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

advisory[1234] * Writes	to flas	sh must	target	opp	osite s	ide of	400	00h boundar
femory Summary:								
Program space	used 7	7960h (	31072)	of	8000h	bytes	(	94.8%)
Data space	used	53Fh (	1343)	of	600h	bytes	(	87.4%)
EEPROM space	used	Oh (	0)	of	100h	bytes	(	0.0%)
External data memory	None av	7ailable	•					x 2.2.10923-0694
ID Location space	used	Oh (	0)	of	8h	nibbles	(	0.0%)
Configuration bits	used	4h (	4)	of	7h	words	(	57.1%)
.oaded C:\SSC_410\SPI\SRC\SI	ave Samp •••	ole Code \	/410.cof.					
•1	Yan							

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

MPL	AB ICD 2 Setup Wizard 🛛 🛛 🔀
C	Enable auto-connection.
	Would you like MPLAB IDE to automatically connect to the MPLAB ICD 2 on start up?
	If enabled, MPLAB IDE conveniently connects to the MPLAB ICD 2 at project startup.
	If disabled, MPLAB IDE waits until the user initiates a connect operation before attempting to communicate with the MPLAB ICD 2.
	Note: The potential downside to enabling auto connection is that MPLAB IDE will always attempt to communicate with the MPLAB ICD 2 on project startup. If there is a device other than the MPLAB ICD 2 on the selected communications channel it might be damaged by this attempted communication.
	MPLAB IDE automatically connects to the MPLAB ICD 2
	<zutiuck weiter=""> Abbrechen Hilfe</zutiuck>

Figure 31: Auto connect

g. Check automatic download operating system

1	.AB ICD 2 Setup Wizard
I	Download Enable auto-download of operating systems.
	Would you like MPLAB IDE to automatically download (without prompting) MPLAB ICD 2 operating systems when needed?
	The MPLAB ICD 2 uses different operating systems for different target families.
	If enabled, MPLAB ICD 2 will automatically download the appropriate operating system whenever it detects that the operating system currently residing within the MPLAB ICD 2 is not the correct one for the selected target device, or whenever MPLAB IDE detects a newer version of the operating system on the users computer.
	If disabled, MPLAB ICD 2 will prompt the user for permission to download before actually downloading.
	$\checkmark$ MPLAB ICD 2 automatically downloads the required operating system
	77.viick Waiters Abbrechen Hilfe

Figure 32: Download OS

h. Connect to debugger

File Edit View Project I	Debugger	Programmer	Tools	Configure	Win
0 📽 🖬   🐰 🖿	Select T Clear Me	ool emory		۲ ۲	pug
HW BP 1 Used	Run			F9	
Stave Sample Code	Animate Halt			F5	
Source Files	Step Int	0		F/	
	Step Ov	er		18	
	Step Ou	t:			
coeappi.(	Reset			•	
ecataoe.	Breakpo Advance	<b>ints…</b> ed Breakpoints	à	F2	
ecateoe.	MPLAB I	CD 2 Setup Wi	zard		
ecatfoe.d	Program	6			
C ecetone	Read				
	Read EE	PROM			
eoeappl.(	Abort O	peration			
foeappl.d	Connect				
	Downloa	nd ICD2 Opera	ting Sys	tem	
	Settings		315 174		

Figure 33: Connect to debugger

i. Acknowledge the download warning with [OK]

ICDWarn0030: M If MPLAB IDE is Please be patie downloaded. (N	PLABICD2 is about just starting, it will ap nnt. MPLABIDE will lote: You may wish fut	to download a opear to "hang finish it's initia to select to igr ure.)	a new operating system '' at the splash screen lization after the OS is nore this warning in the
🔲 Don't display t	his warning again		
	ОК	Cano	el
	<b>—</b>		

Figure 34: Download warning

j. Output if connection succeeds

Output			
Build Version Control	Find in Files	MPLAB ICD 2	
Auto-connect not er Connecting to MPL/ Connected Setting Vdd source Target Device PIC1 Reading ICD Proc Running ICD Self Ta Passed MPLAB ICD 2 ready	abled - Not AB ICD 2 to target 8F452 found luct ID est / for next ope	connecting (T d, revision = cl eration	ry enabling auto-connect on the ICD2 settings pages.)

Figure 35: Output window: connection successful

k. Download the binary file

ile Edit View Project I	Debugger	Programmer	Tools	Configure	Win
D 📽 🖬   🐰 🐂	Select Tool Clear Memory			+ +	pug
HW BP 1 Used	Run			F9	
🗖 Stave Sample Code	Halt			F5	
	Step Int	:0		F7	
Source Files	Step Ov Step Ou	rer		F8	
coeappl.c	Reset			•	
ecataoe 	Breakpo Advance	ints ed Breakpoints		F2	7
ecateoe.	MPLAB 1	CD 2 Setup Wi	zard		
ecatfoe.	Program	1			ì
ecatsly.c	Read Read EE	PROM			
eoeappl.	Abort O	peration			
Foeappl.d mailbox.d	Connect Downloa	: ad ICD2 Opera	ting Sys	tem	
objappl.d⊂ 	Settings	la			

Figure 36: Program PIC memory

Output if programming succeeds

ulia	Version Control	Find in Files	MPLAB ICD 2
Deb	ug Executive		
Deb	ug Vector		
Veri	fy Succeeded		
Progr	amming Config	guration Bits	
Con	fig Memory		
Verify	ing configurati	on memory.	
veri	ry Succeeded		
Drov	ecting to debu	g executive	
1FIU	ar-2009 10:44:2	Deeueu 21	
09-Ma	ar-2009, 10:44:2	21 21	
15 14 15 16 19 19 19 19 19 19 19 19 19 19 19 19 19			

Figure 37: Output window: Programming successful

I. Select [Debugger]  $\rightarrow$  [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: <u>http://www.beckhoff.com</u>

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: 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: <u>www.ethercat.org</u>