Beckhoff Automation



TwinCAT 3 PLC Training



Module 3 – First Project



Introduction and Quick Start

Modules 1 – 4

Programming Basics

Modules 5 – 13

Diagnostics and Hardware

Modules 14 – 16

Introduction



- 1. General PLC Architecture
- 2. TwinCAT 3.1 Installation
- 3. First Project
- 4. Setting the Cycle Time

- After installation of TwinCAT 3.1 is complete and the computer has rebooted, the Cog is added to your System Tray.
- This provides a convenient shortcut menu to interact with TwinCAT, and to start up TcXAEShell, which is the development environment used to develop TwinCAT applications.
- Click on the Cog and select TwinCAT XAE (TcXaeShell).







3.1 Introduction to TcXAEShell



After a few seconds, TcXaeShell will open to the start page, allowing easy access to the last few projects.

On the right hand side, select New TwinCAT Project.





The New Project Wizard will open

- 1. Select TwinCAT Project in the collection on the left-hand side.
- 2. Select TwinCAT XAE Project (XML Format) on the right-hand side.
- 3. Give your project the name TC3_Project_1.
- 4. Tick the Create directory for solution checkbox.
- 5. Click on OK.



BECKHOFF

The different parts of the project are accessed in the Solution Explorer window on the left.

Compile and runtime errors are displayed in the Error List at the bottom.

Properties of selected items and the Toolbox are available in the window on the right.





At this point, the Solution is just an empty framework, and doesn't contain a PLC project.

Add a PLC Project by rightclicking on PLC in the solution explorer and selecting Add New Item...





This opens up the Add New Item wizard

- 1. Select Plc Templates in the collection on the left-hand side.
- 2. Select Standard PLC Project on the right-hand side.
- 3. Give the PLC project the name PLC_Training.
- 4. Click the Add button.



Your Solution should now contain a PLC Project framework.

The various items in the tree can be expanded by clicking on the arrow to the left of the collapsed item.

The PLC Project contains everything we need to run PLC code, including an empty Program called Main.

TC3_Project_1 - TcXaeShell Build Debug TwinCAT View Project 8 G - O | 🎁 - 🎦 - 😩 💾 📲 命 り、 2 🎯 🔯 🍡 🔏 TC3_Project_1 Solution Explorer G O 🟠 🛗 -~ Search Solution Explorer (Ctrl+ Solution 'TC3_Project_1' (1 project) TC3_Project_1 SYSTEM MOTION DD PLC PLC_Training 🔺 🚔 PLC_Training Project External Types References DUTs 🧰 GVLs POUs MAIN (PRG) VISUs 🔓 PicTask (PicTask) PLC_Training Instance SAFETY C++ ANALYTICS 1/0

BECKHOFF

Double-click MAIN (PRG) to reveal the coding interface in the middle window.

The coding interface is separated into two halves:

- The top half is for declaring variables.
- The bottom half is for editing program logic.

Before we write any logic, we are going to declare some Global Variables that represent some digital inputs.



le la



Add to Source Cont



Global Variables are declared and stored in a Global Variable List, or GVL.

To add a GVL:

- Right-click on the GVLs folder.
- Select Add \rightarrow Global Variable List.

TC3_Project_1 - TcXaeSh	ell												
File Edit View Project	Bui	ld Dek	bug T	winCAT	Twin	SAF	E I	PLC	Team	Scope	Tools	Window	Help
🛛 🖸 - 🗇 👘 🖓 - 🍅 - 🏠	🗎 🖌	* X	r 🖞	9-	C	Rel	ease		• Twin	CAT RT (x	54)	- 🕨	Attach 👻
Build 4024.7 (Loaded) 🔹	÷ 🗄	è 🖪	💶 🥩 🛛	< 🖸	💽 🖫	9	s 1	гС3_Р	roject_1		• <loo< td=""><td>al></td><td>-</td></loo<>	al>	-
Solution Explorer			→ Į	x	MAIN +	• ×	(
G O 🔓 🛗 - To - A	1	_=		- 1	1		PROG	RAM	MAIN				
Search Solution Explorer (Ctrl+	;)			ρ-	3		END_	VAR					
Solution 'TC3_Project_1' (1 project) Solution 'TC3_Project_1 G SYSTEM MOTION PLC PLC PLC PLC_Training PLC_Training PLC_Training SOLUTION SOLUTIO					4								
Reference	ces			- 1	1								
DUTs													
GVLs	А	dd					•	₿	POU				
🗐 M/	E	Export to ZIP Import from ZIP Import PLCopenXML						₿	POU for	implicit ch	necks		
🚞 VISUs	In							¢ ₁ ¢	DUT				
PICTas PIC Train	ln 📰							2	Global V	ariable List			
SAFETY C	Жc	Cut		Ctrl+X			é	Reference	ed Task				
<u>к</u> С++	D C	ору			Ctrl+C			B	Visualiza	tion			
ANALYTICS	× D	C Delete			Del				Visualiza	tion Mana	ger		
▶ <mark>≱</mark> I/O	X R	ename						GlobalTe	xtList				
	Properties				Alt+Enter			A	Recipe N	lanager			
-									Image P	ool			
				- 1				⊶	Interface	·			
								Pø	Paramet	er List			
									Text List.				
								品	Class Dia	igram			
								* a	Existing	tem		Shift+/	Alt+A
								*-	New Fol	der			
									Existing	Folder Cor	ntent		



When the Add Global Variable List wizard appears:

- Set the name to Global_IO.
- Click Open.

Add Global Variable List						
Create a new global variable list						
Global_IO						
Open Cancel						



The new Global Variable List is automatically opened after being created.

If you want to access it in the future, simply double-click on the Global_IO node in the GVLs folder in the Solution Explorer.



Add two new Input variables, taking care to copy the syntax exactly.

These two variables will be accessible from anywhere in our PLC code.

The {attribute 'qualified_only'} at the top means that the variable names must be prefixed with the GVL name Global_IO whenever we reference them.







Go back to your MAIN program by double-clicking the node in the Solution Explorer, or by selecting the open tab at the top of the Editing Window.





Add the code to implement an AND gate shown below into the bottom window within the MAIN program.

Solution Explorer	• 4 ×	Global_IO*	
Solution Explorer Search Solution Explorer (Ctrl+;) Solution 'TC3_Project_1' (1 project) C3_Project_1 Solution 'TC3_Project_1' (1 project) C3_Project_1 SYSTEM SYSTEM SYSTEM PLC PLC_Training PLC_Training Project External Types Solution DUTs	× ∓ × - م	Global_IO* 1 2 3 4	<pre>MAIN* * X PROGRAM MAIN VAR END_VAR IF (Global_IO.bSwitch1 AND Global_IO.bSwitch2) THEN Global_IO.bLED1 := TRUE;</pre>
 ▲ GVLs Global_IO ▲ POUs ▲ MAIN (PRG) ▲ VISUs ▶ PIcTask (PIcTask) ④ PLC_Training Instance SAFETY SAFETY C++ ANALYTICS ▶ Z I/O 		■ 3 4 5 6 7	ELSE Global_IO.bLED1 := FALSE; END_IF

Let's compile the code to check for errors and get it ready to be executed.

- Select Build from the menu across the top of the TcXAEShell window.
- Then select Build Solution.

An easy keyboard shortcut to use in future whenever you want to build or compile your code is Ctrl+Shift+B.





If there aren't any errors in your code, you should see "Build succeeded" in the bottom left-hand corner of Visual Studio, as well as "0 Errors" in the Error List.

If you do have errors, they will appear in the error list, and double-clicking on the error will move the editing cursor to the line of code that has the error.

Clearing the list before building helps identify compile errors, if the XAR or hardware is also reporting errors.







Since this is a brand new project, the project structure will need to be sent to the TwinCAT XAR which executes your application.

This process is called Activating, and is performed by clicking the Activate button in the toolbar.





Activating your project will stop any PLC code that is running, so a warning / confirmation dialog box opens asking if you are sure you want to continue.

No PLC code is running at the moment, so click OK to continue.

Activate Configuration					
Project:	TC3_Project_1				
Target:	<local></local>				
	Autostart PLC Boot Project(s)				
	OK Cancel				

Since we're running our PLC code on a normal computer, you won't have any licenses installed.

TwinCAT 3 allows the use of trial licenses that last 7 days for almost all available features. Once the 7 day trial is over, a new trial license can be generated as many times as necessary to complete development of your application.

Whenever you activate your project and XAE discovers a missing license, it will prompt you to generate a trial license, so click Yes to continue, and copy the code from the top textbox into the lower textbox exactly and click OK.







Finally, TwinCAT asks if it's OK to restart the XAR into

Run Mode.

This is required in order to start up the XAR in preparation to execute your application.

Please click on OK to continue.

At the bottom of the TcXAEShell window, you should see a Blue cog icon change to Red and then Green a few seconds later. This indicates that TwinCAT XAR is in Run Mode.









Now that the XAR is in Run Mode, we can connect and download our PLC code to the PLC software component that has been created for us inside the XAR.

This is called "Logging in", and is done by clicking the Login button in the toolbar.





The first time we log in after activating, TwinCAT will ask to create the Application inside the PLC block.

Click Yes to continue.



After logging in, the PLC application has been downloaded into the XAR and is ready to run.

Start the PLC by clicking on the Start button in the toolbar.





BECKHOFF

Now that you're logged in and the PLC is running, the values of the variables are displayed next to each variable.

Double clicking on the FALSE value to change it to FALSE <TRUE> shows that a modification to that value is queued up. Set both bSwitch1 and bSwitch2 to have a queued value of TRUE.



MAIN	[Online]	🗢 🗙 Global_IO [Onl	ine]					+
тса	B_Project	_1.PLC_Training.MAI	N					
Expre	ssion	Туре	Value	Prepared value	Address	Comment		100
	1	IF (Global	IO.bSwitch	FALSE <true></true>	AND G	lobal IO.bSwitch	FALSE <true></true>	THEN
	2	Global	IO.bLED1 FA	LSE IKUE,		—		
	3	ELSE						
	4	Global	IO.bLED1 F/	LSE := FALSE	;			
	5	END_IF						
I	6							
	7	RETURN						



To send all the queued writes to the PLC, click on the ¹Write Values button in the toolbar.

TC3_Project_1 - TcXaeShell							\bigtriangledown	Quick Launch	(Ctrl+Q)	ð
File Edit View Project Build	Debug TwinCAT	TwinSAFE PLC	Team Scope	Tools Window	Help					
G • O 🏠 • 📩 • 🚔 💾 📲	よ 🗗 🏦 🦻 🗸 🤇	🤍 👻 Release	- TwinCAT RT (x64	4) – 🕨 At	ttach 🝷	- 🔎	ltpReset	_	- 🖉 🖉 🐺] 🗰 🍇
	TC3_Project_1	✓ <local></local>	~ ,	PLC_Training	• 1	- J - E	: : :) 🗏 🖸 🖆	1 U U -	
Solution Explorer 🛛 🔻 🕂 🗙	MAIN [Online] 😐 🗵	Global_IO [Online]								💌 Pro
© © ☆ 🛱 - Ĭ₀ - @ 🗡 Ï	TC3_Project_1.Pl	C_Training.MAIN								Ŷ
Search Solution Explorer (Ctrl+;)	Expression	Туре	Value	Prepared value	Address	Comment			Ī	
 Solution 'TC3_Project_1' (1 project) TC3_Project_1 SYSTEM MOTION PLC PLC_Training PLC_Training Project 										Prc
 External types References DUTs GVLs Global_IO POUs MAIN (PRG) 	 1 1F 2 3 4 5 EN 	(Global_IC Global_IC SE Global_IC D_IF).bSwitch1).bLED1 <mark>FALS).bLED1</mark> FALS	FALSE <true SE := TRUE SE := FALS</true 	AND (Global_IO.bSv	witch	2 FALSE) THE	EN	

Now, bSwitch1 and bSwitch2 are both TRUE, and as dictated by the logic, bLED1 turns on as well.

Experiment and validate the expected logic by setting bSwitch1 and bSwitch2 to True and False alternately.

bLED1 should only be True when both bSwitch1 and bSwitch2 are set to True.





Module 3 Recap

BECKHOFF

In this module we covered:

- Creating a new TwinCAT XAE project,
- Creating a new PLC Project,
- Adding new GVLs,
- Activating the configuration,
- Logging in,
- Running PLC code in Simulation.

Contact



Beckhoff Automation Ltd.

The Boathouse Station Rd Henley-on-Thames RG9 1AZ Great Britain Phone :+44 1491 410 539E-mail :info@beckhoff.co.ukWeb :www.beckhoff.co.uk

© Beckhoff Automation GmbH & Co. KG April 2020

All images are protected by copyright. The use and transfer to third parties is not permitted.

Beckhoff[®], TwinCAT[®], EtherCAT[®], EtherCAT G[®], EtherCAT G10[®], EtherCAT P[®], Safety over EtherCAT[®], TwinSAFE[®], XFC[®], XTS[®] and XPlanar[®] 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.

The information provided in this presentation contains merely general descriptions or characteristics of performance which in case of actual application do not always apply as described or which may change as a result of further development of the products. An obligation to provide the respective characteristics shall only exist if expressively agreed in the terms of contract.