Application Note

FESTO

Commissioning VTEM connected to Beckoff PLC via CPX-FB38 bus node

Commissioning VTEM connected to Beckoff PLC via CPX-FB38 bus node

VTEM

100186

Title ..... Commissioning VTEM connected to Beckoff PLC via CPX-FB38 bus node
Version 1.10
Document no. 100186
Original ..... en
Author ..... Festo

Last saved 11.10.2018

Copyright Notice

Legal Notice

Internet: http://www.festo.com
E-Mail: service_international@festo.com

Table of contents

1 Components/Software used

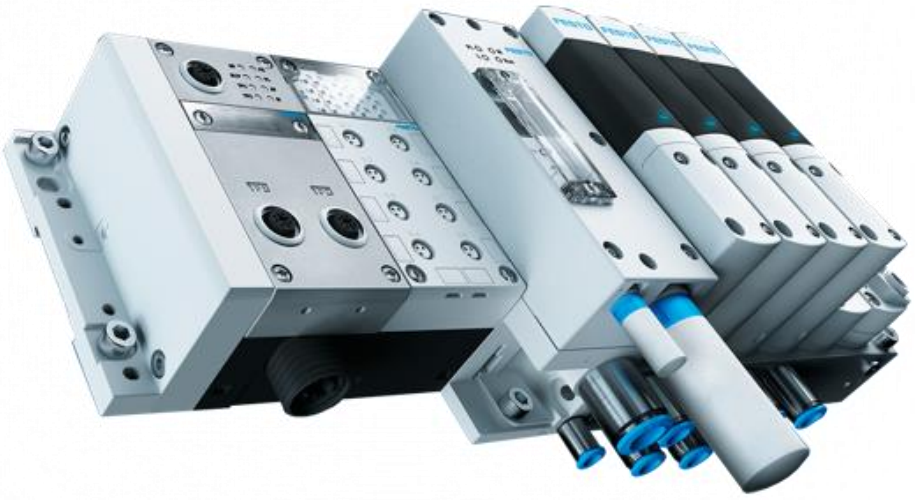
Type/Name	Version Software/Firmware	Date of manufacture
VTEM-S1-27-E1-B1-Q10-U-Q6-4C-BAPSTPDLP-2PD-2SP-AD 50E-F37GCQS-P	To filled	September 2017
EtherCAT FB-38 XML	Rev 11 (Datatype USINT & BITARR8)	16-12-2014
Codesys Library	Festo_MotionTerminal_3 1.1.88.0 (Festo AG & Co.KG)	To be filled

Table 1.1: 1 Components/Software used

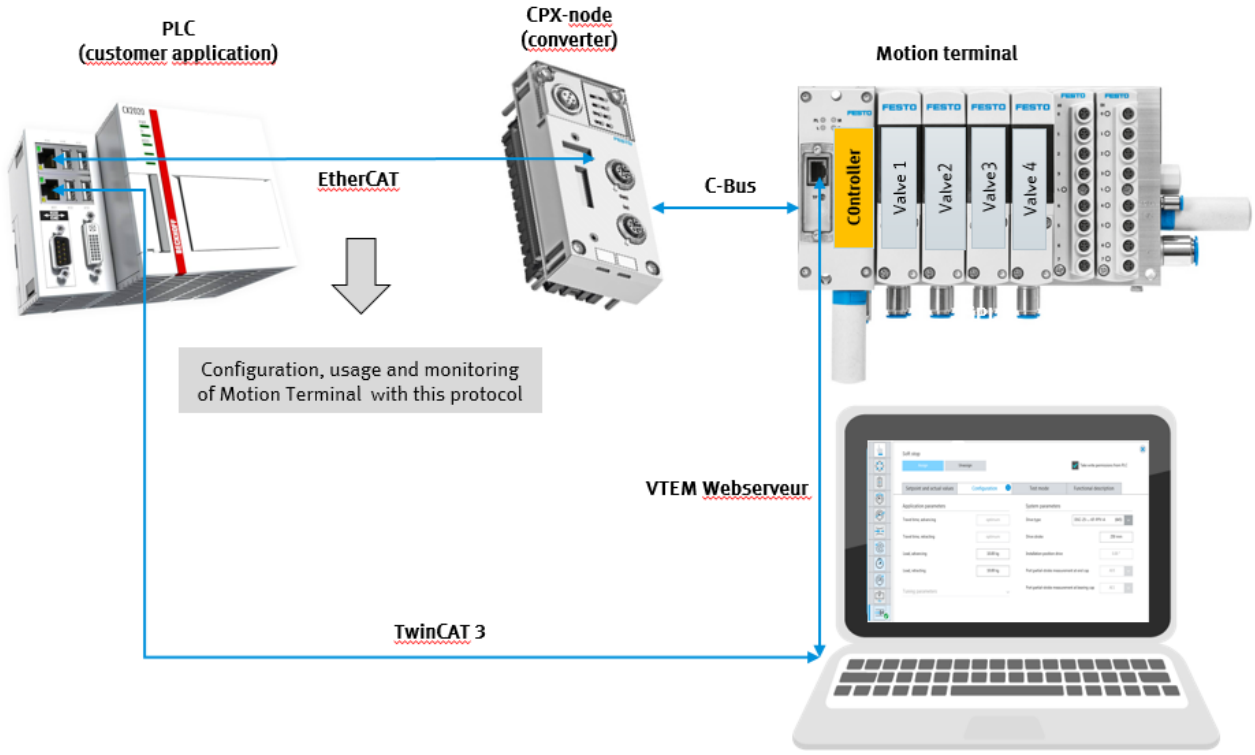
1.1 Application description

This part of the documentation describes the connection and configuration of the VTEM within a EtherCAT network. The target audience is technicians who are already familiar with this bus protocol.

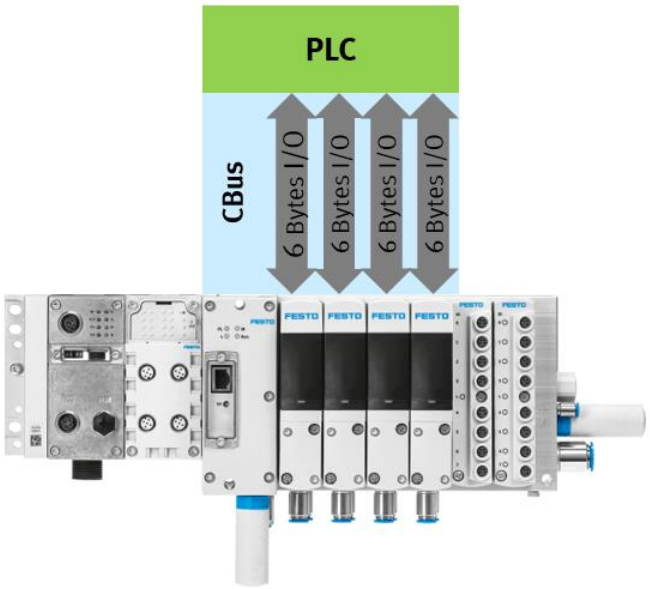
EtherCAT (Ethernet for Control Automation Technology) is an Ethernet-based fieldbus system, developed by Beckhoff Automation. The protocol is standardized in IEC 61158 and is suitable for both hard and soft real-time computing requirements in automation technology.



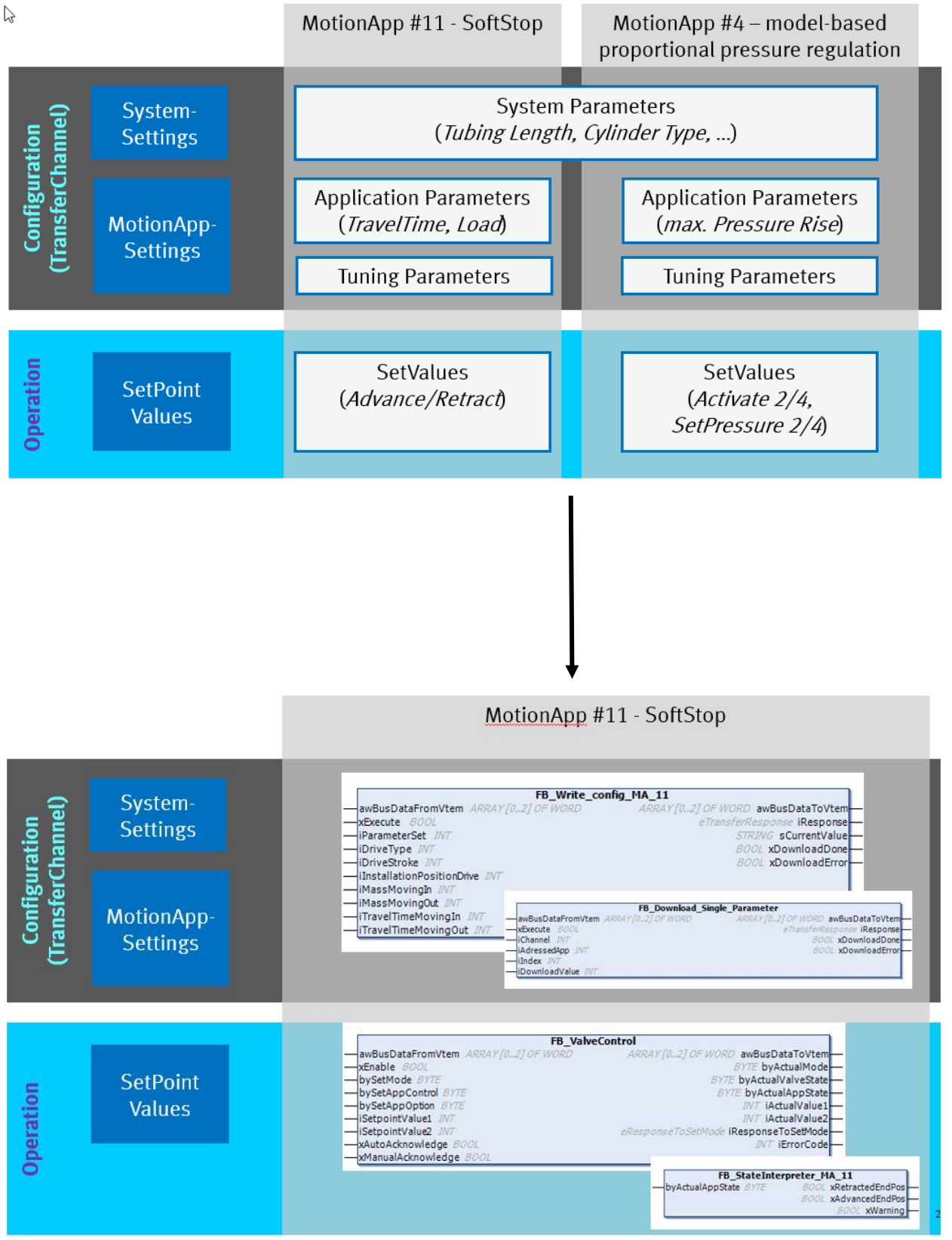
1.2 Topology



- 6 Bytes of data are exchanged cyclically between each valve and the controller
- Each valve can be used independently from each other
- Sensors connected to input modules are not accessible directly via Cbus



1.3 VTEM function block library structure



Note

- Do not use the transfer channel in the same time as Operation function block.

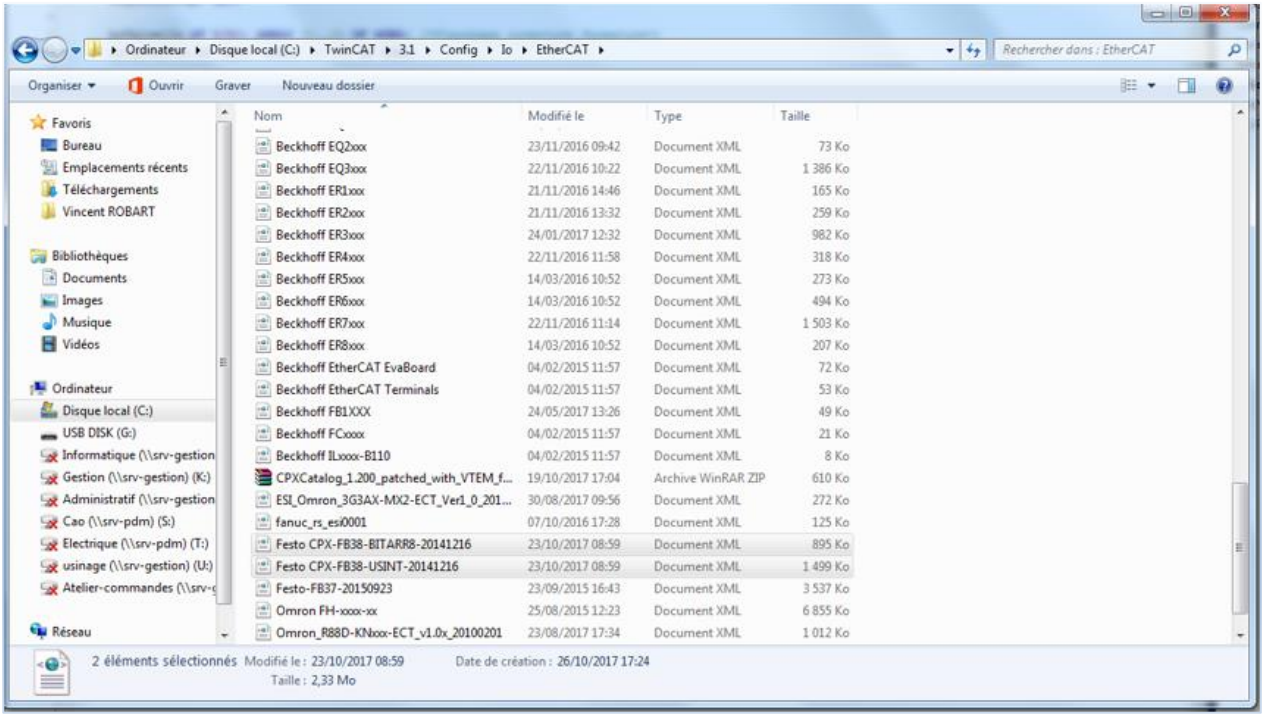
2 CPX-FB37 configuration

2.1.1 Prerequisites

We will use the CPX-FB37 bus node in FB-38 compatibility mode. Download the two CPX-FB-38 XML files in the Festo support portal: https://www.festo.com/net/fr_fr/SupportPortal/default.aspx?q=552046&tab=4&s=t#result and configure the DIL 1.2: ON (fixed I/O size active (64 byte I/O)

EtherCAT addressing type	Setting DIL switch 1	
Modular Device Profile (MDP) active		DIL 1.2: OFF (factory setting)
Fixed I/O size active (64 byte I/O) (compatible with the bus node CPX-FB38)		DIL 1.2: ON

Copy the tow XML in : C:\TwinCAT\3.1\Config\Io\EtherCAT



2.1.2 Network interface

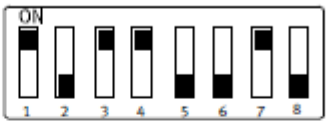
To connect the CPX to the network, there are two 4-pinM12 sockets with D-coding on the bus node (for industrial Ethernet use, corresponding to IEC 61076-2-101). The sockets are compatible with SPEEDCON® plugs.

M12 socket, D-coded	Pin	Signal	Explanation
	1	TD+	Transmission data (transmit data, TD) +
	2	RD+	Receive data (receive data, RD) +
	3	TD-	Transmitted data –
	4	RD-	Received data –
	Housing	FE	Screening/functional earth (FE)

2.1.3 Setting the EtherCAT address

You can assign a free EtherCAT address (Explicit Device ID) to the bus node by using DIL switch 3. Setting an EtherCAT address is optional (e.g. for the Hot-Connect function).

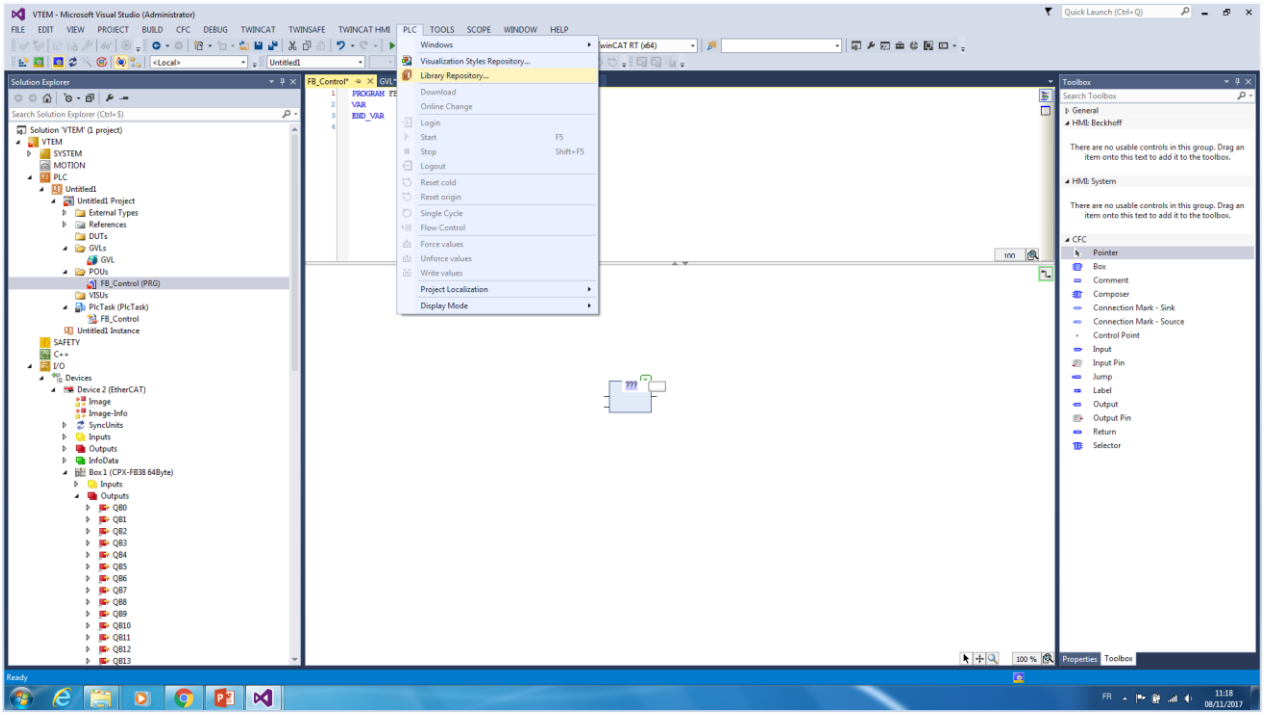
You can set the EtherCAT address in binary coded format by using switch elements 1 to 8. If the EtherCAT address 0 is set, the programmed address in the EEPROM is used (factory setting). 1 to 255 are thereby available as possible EtherCAT addresses.

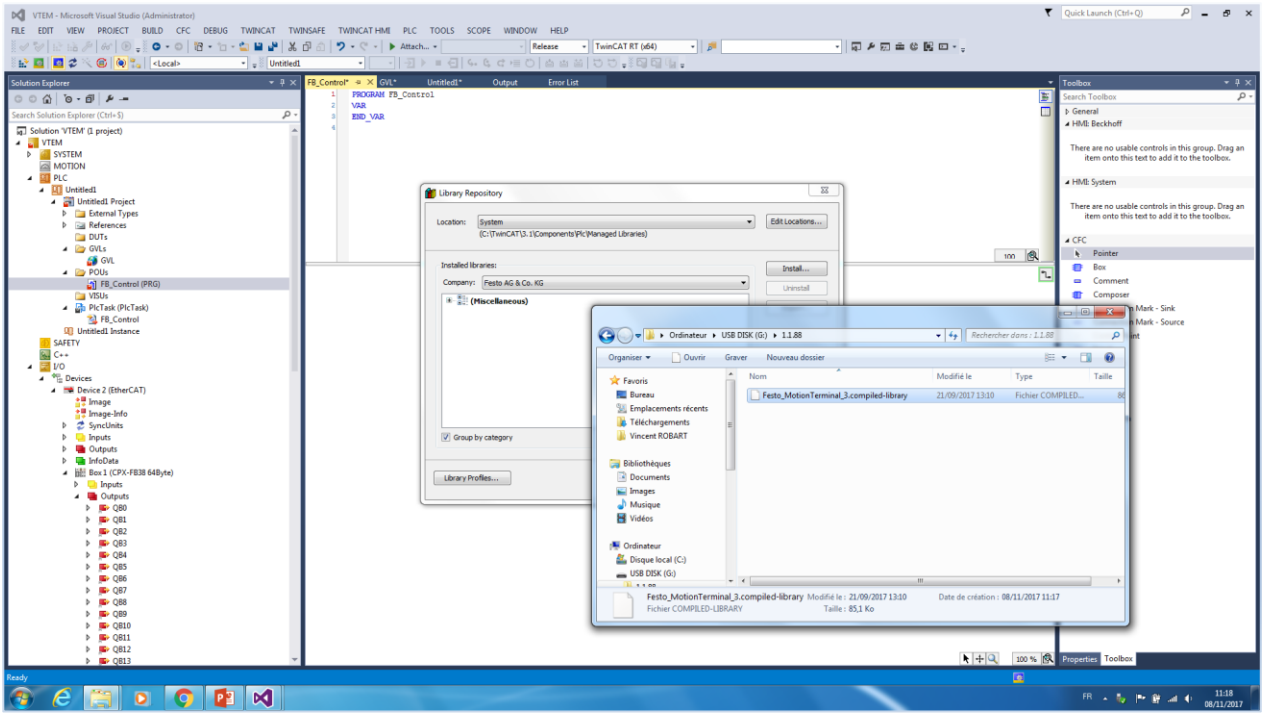
Example: Set EtherCAT address 77								
DIL switch 3								
DIL switch element	1	2	3	4	5	6	7	8
Switch position	**ON**	Off	**ON**	**ON**	Off	Off	**ON**	Off
Binary	1	0	1	1	0	0	1	0
Potency (significance)	2^0	2^1	2^2	2^3	2^4	2^5	2^6	2^7
Numerical value (decimal)	1	0	4	8	0	0	64	0
EtherCAT address	1 + 0 + 4 + 8 + 0 + 0 + 64 + 0 = **77**							

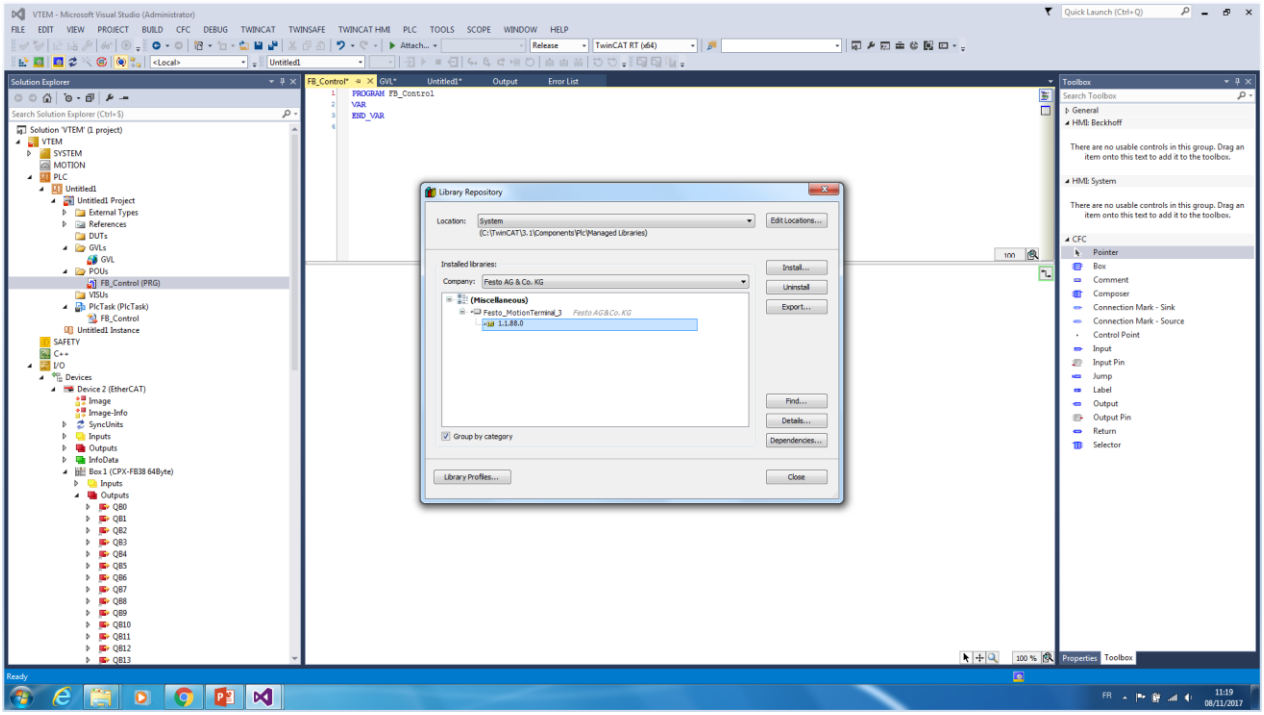
3 Implementation in TwinCAT V3

3.1 Create new project

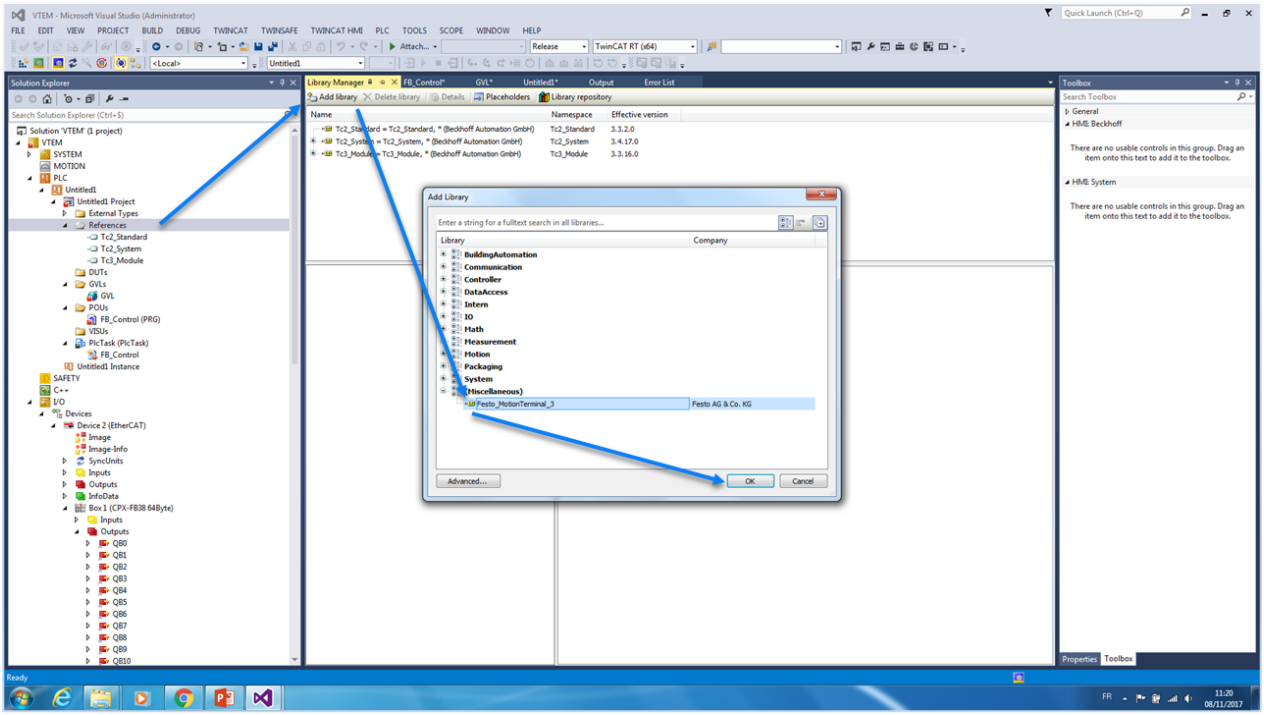
3.2 Install the codesys library



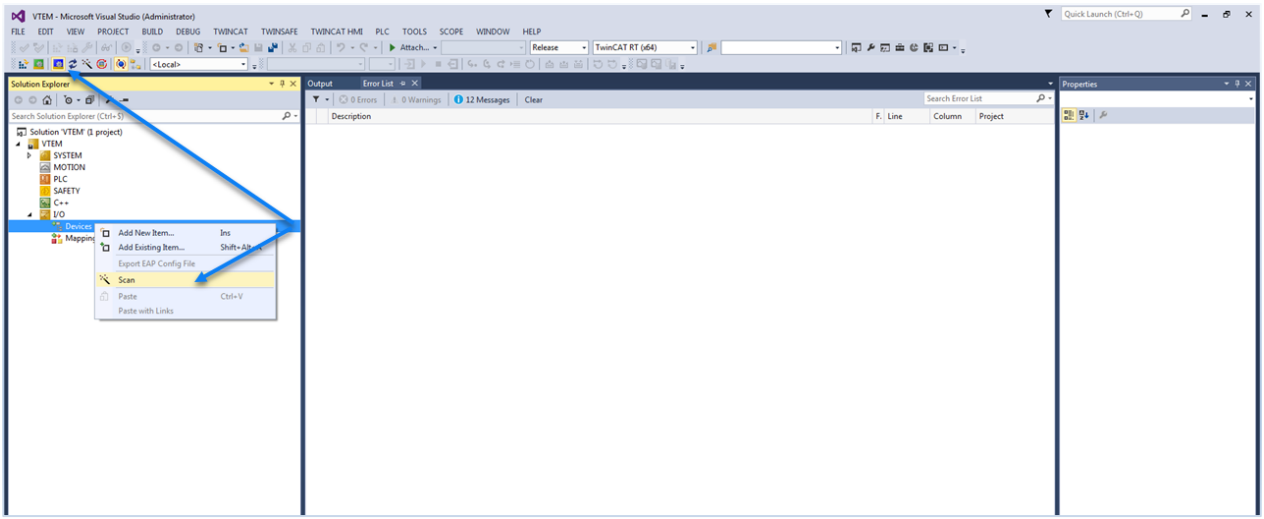


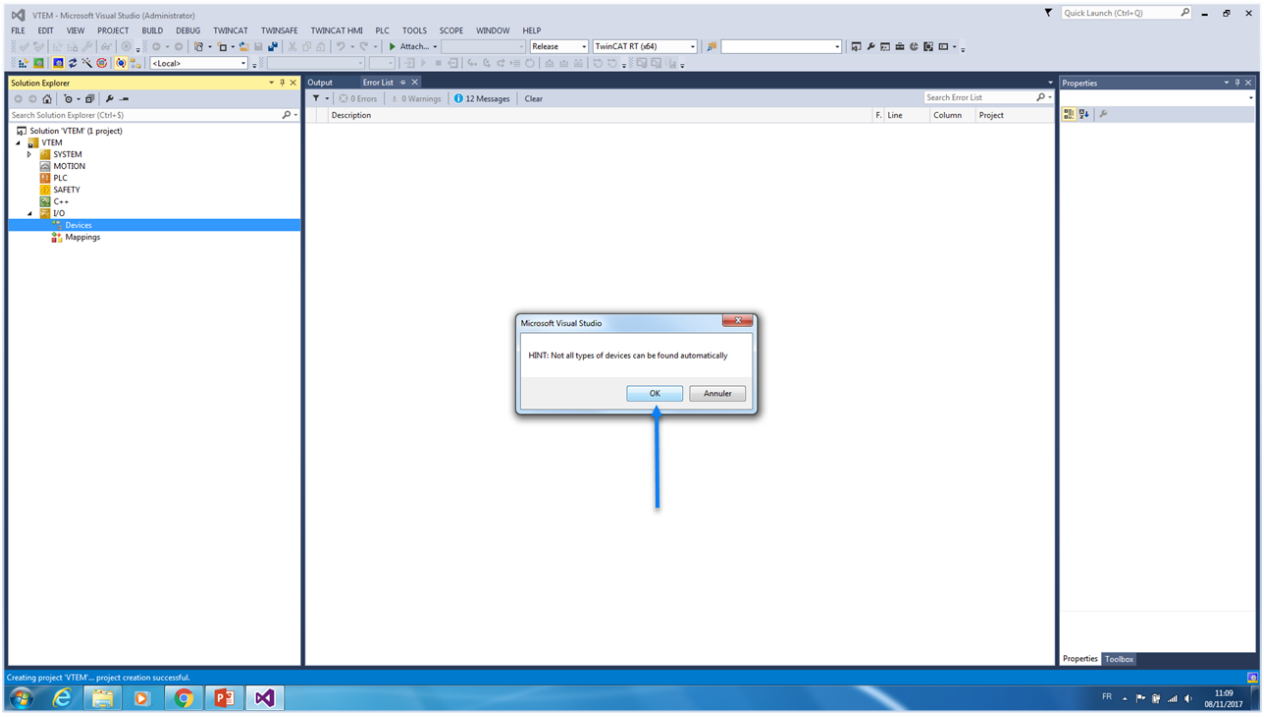


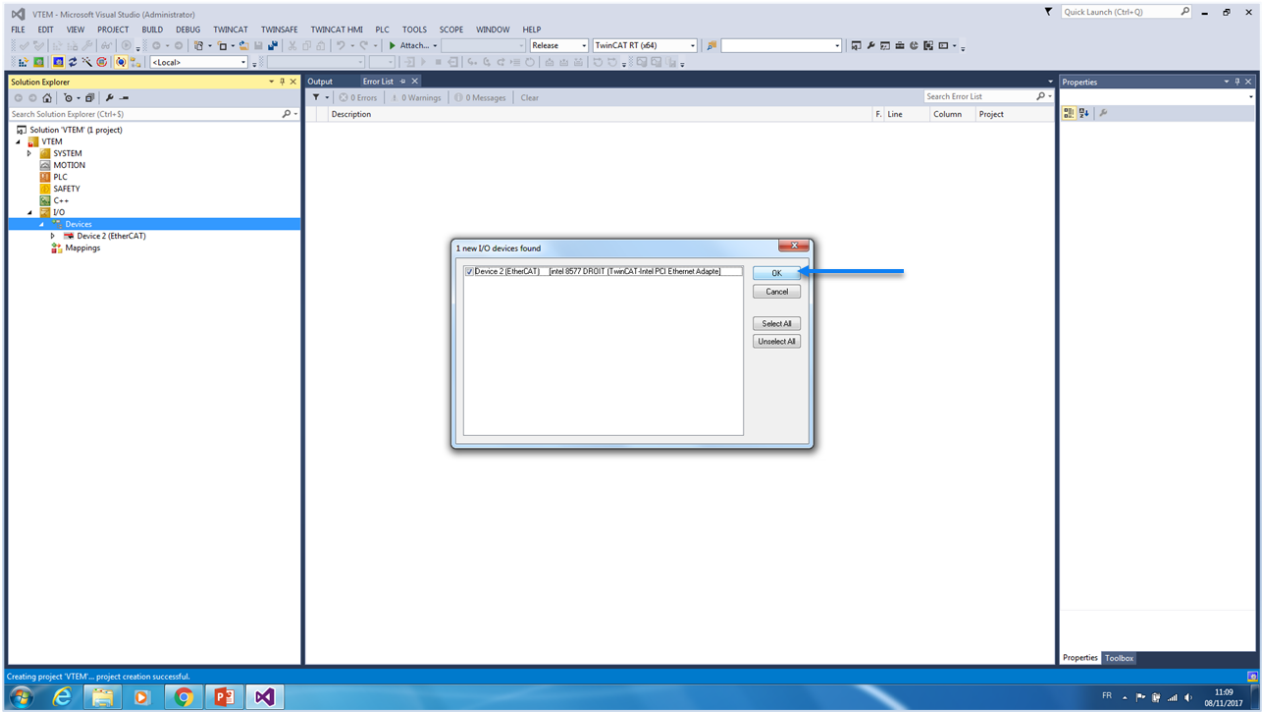
3.3 Add the library in your project

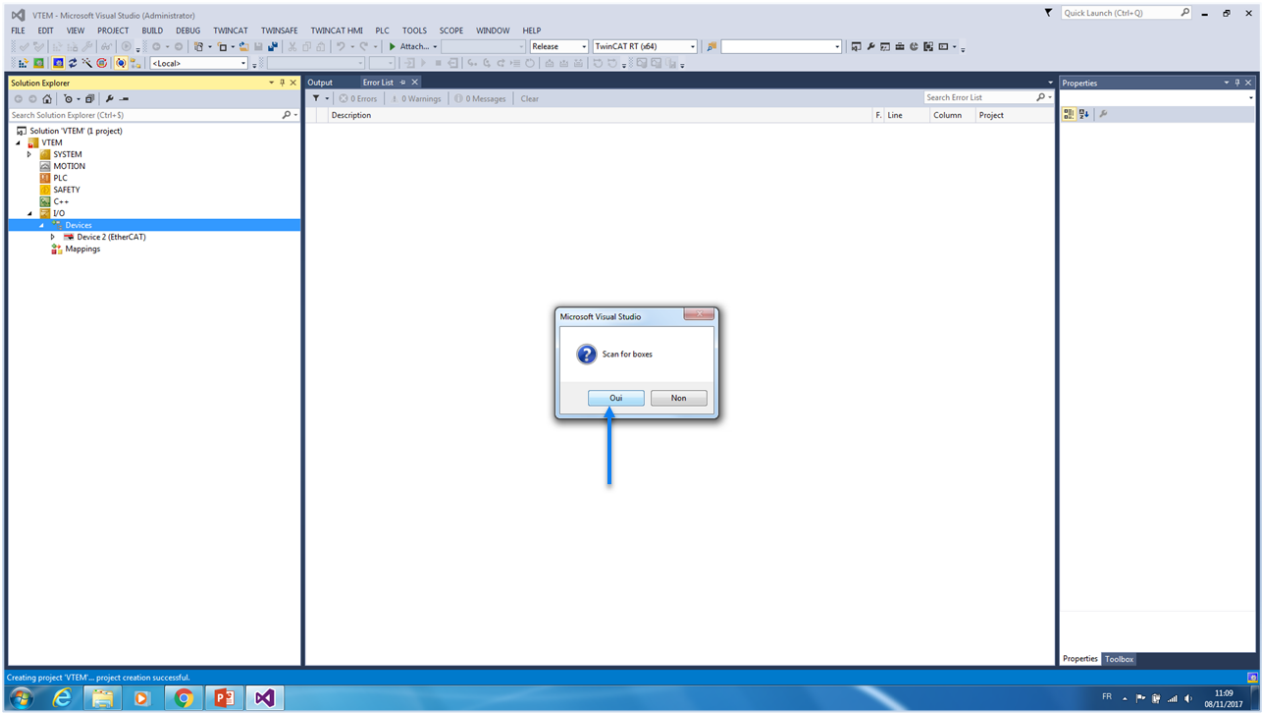


3.4 Scan the network (in config mode)

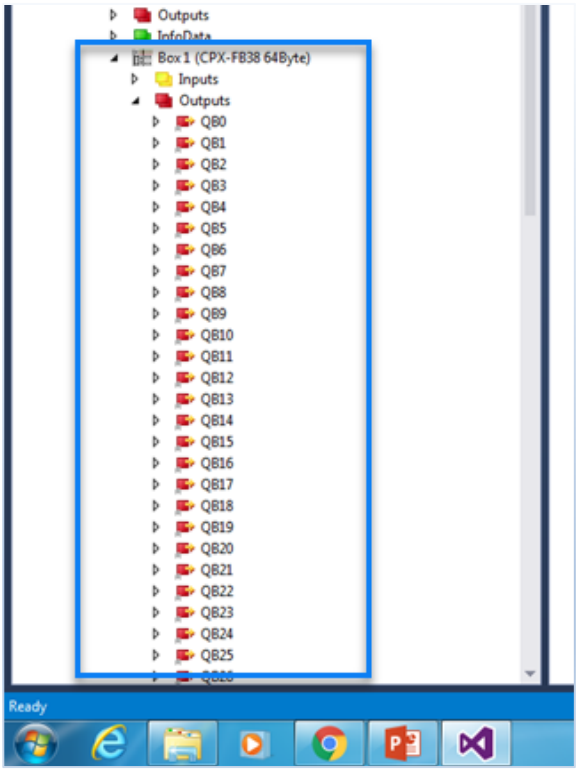



Microsoft Visual Studio
HINT: Not all types of devices can be found automatically
OK
Annuler

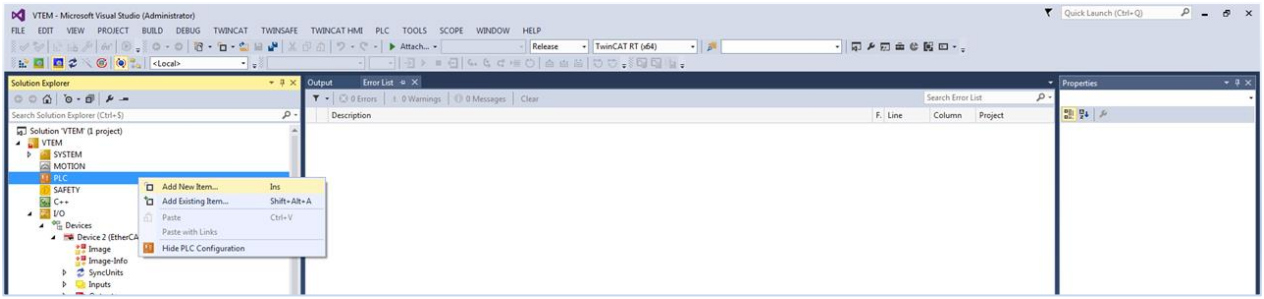

1 new I/O devices found
Device 2 (EtherCAT)
Intel 8577 DRDIT (TwinCAT-Intel PCI Ethernet Adapte
OK
Cancel
Select All
Unselect All

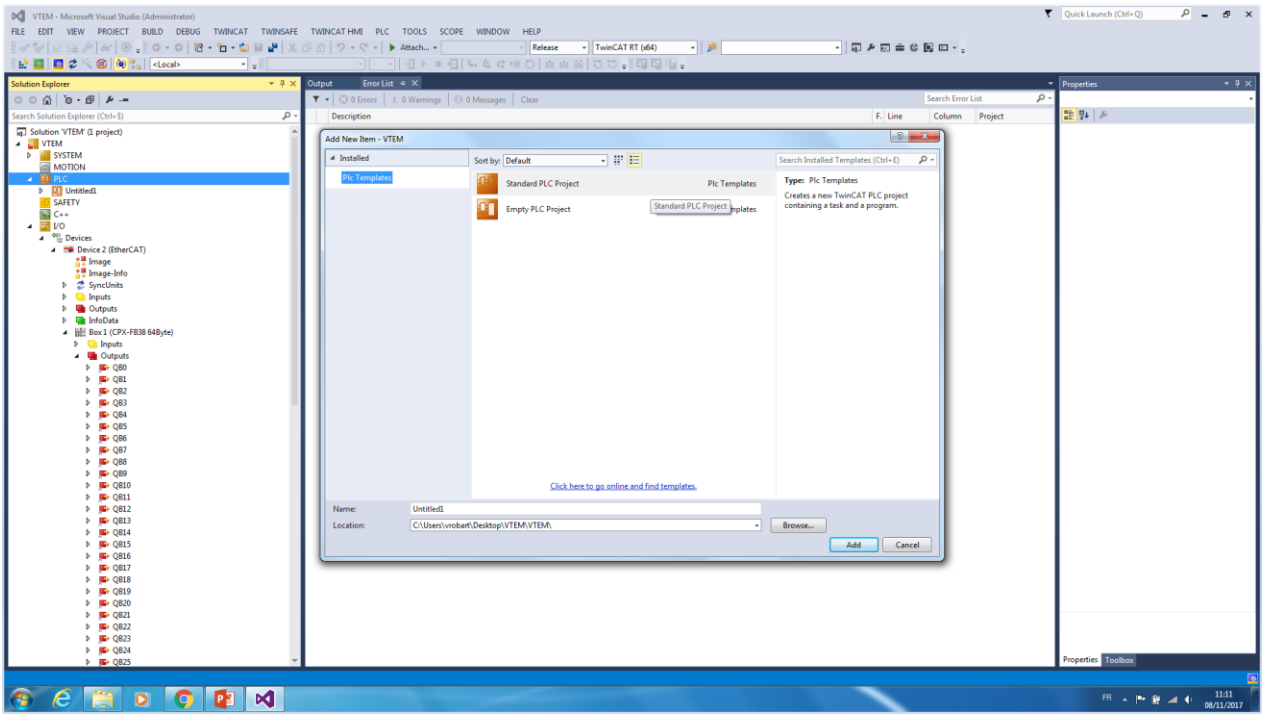


Within the solution explorer you should notice that the CPX device are connected to EtherCAT :



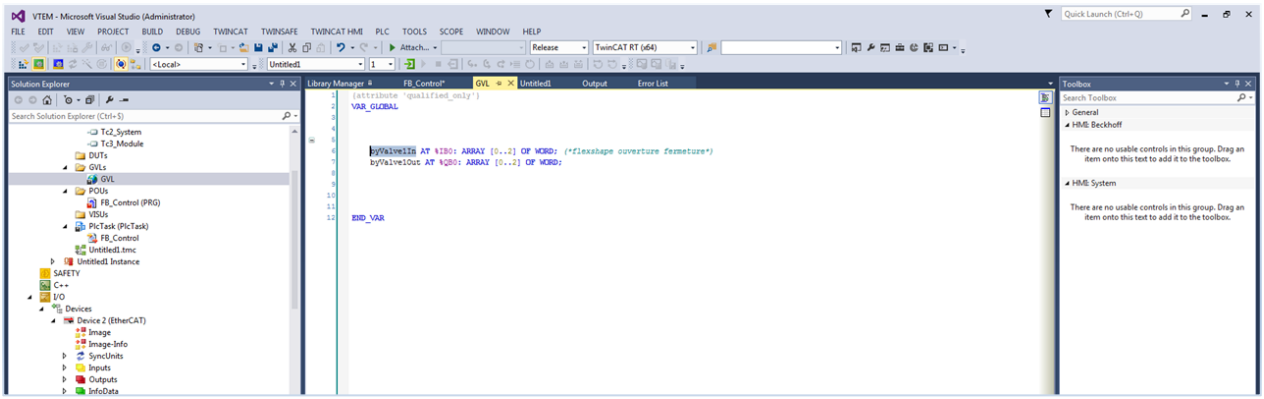
3.5 Add a new PLC item





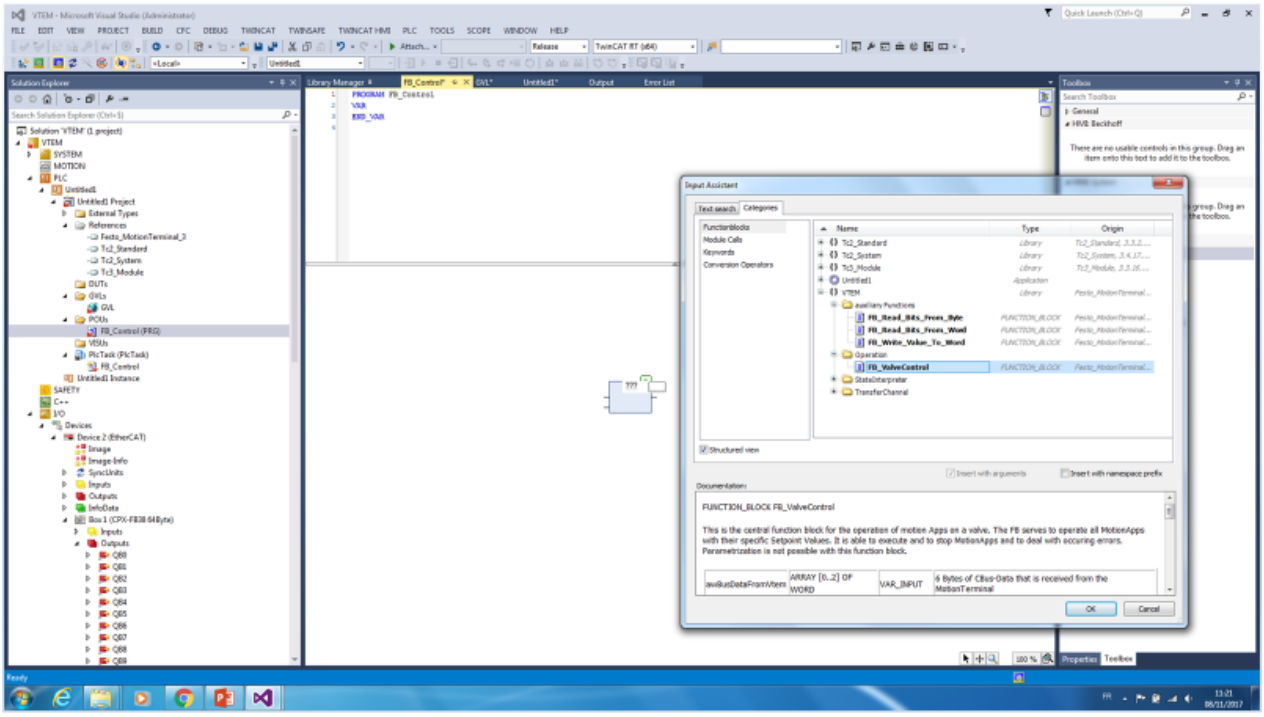
3.6 Configure GVL

In our case, it means that the first byte of the CPX-FB37 correspond to the first valve of the VTEM (because there are no other existing CPX module between the FB-37 and the VTEM):

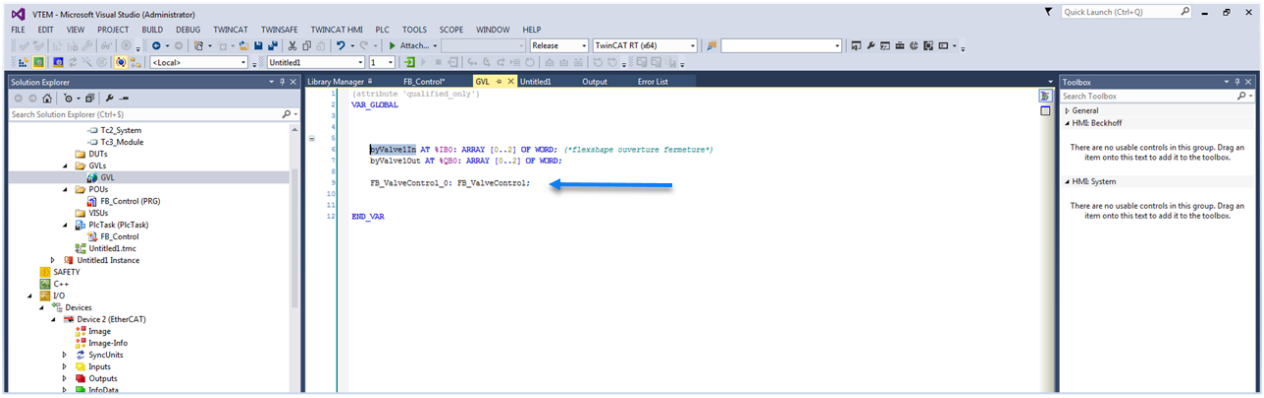


3.7 Add a new POU and function block from the VTEM library

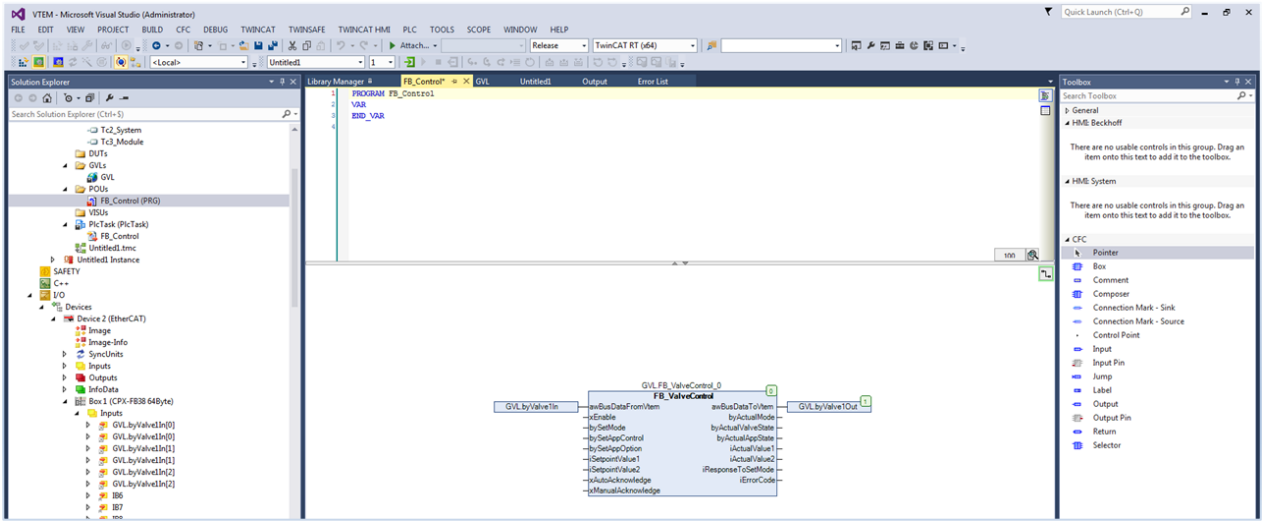
Add the function block you need (eg FB control):



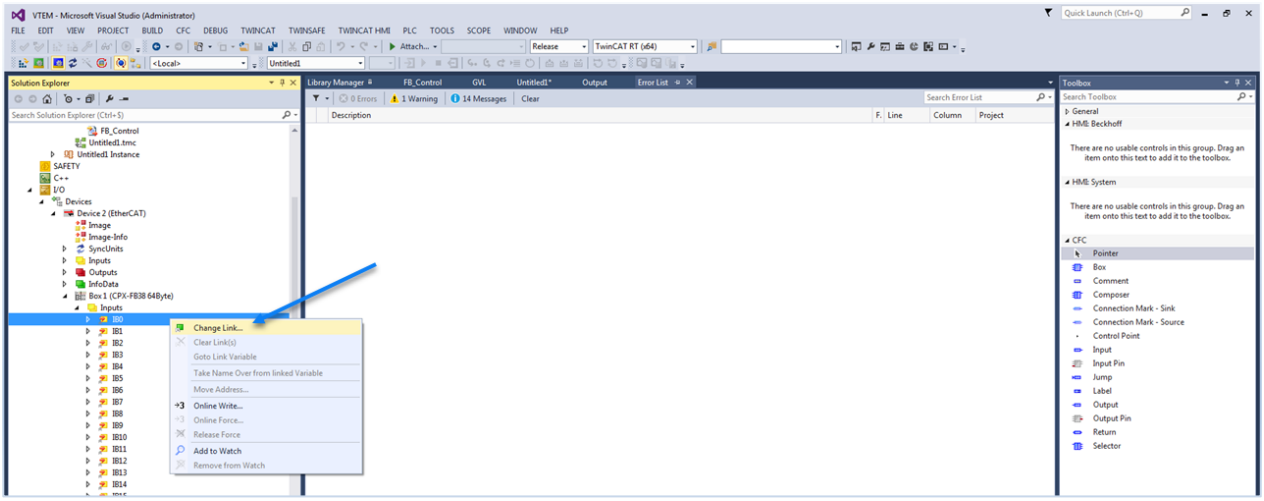
Declare FB as local or GVL:

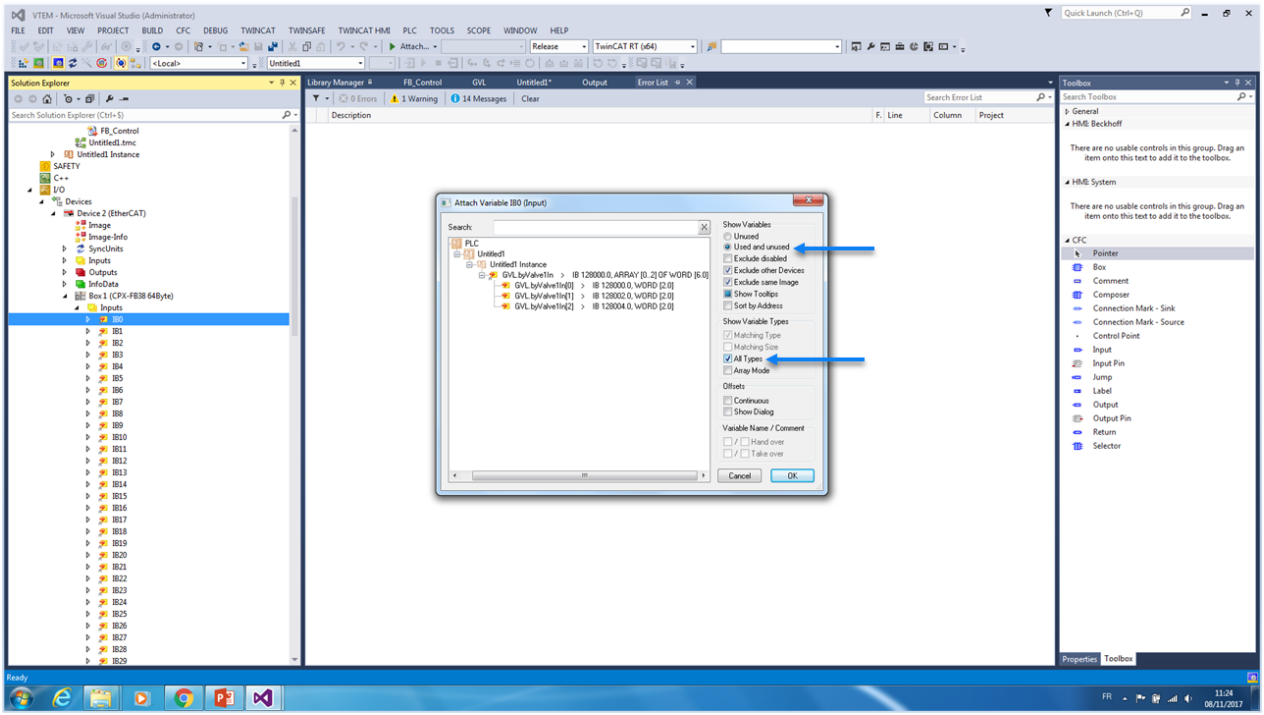


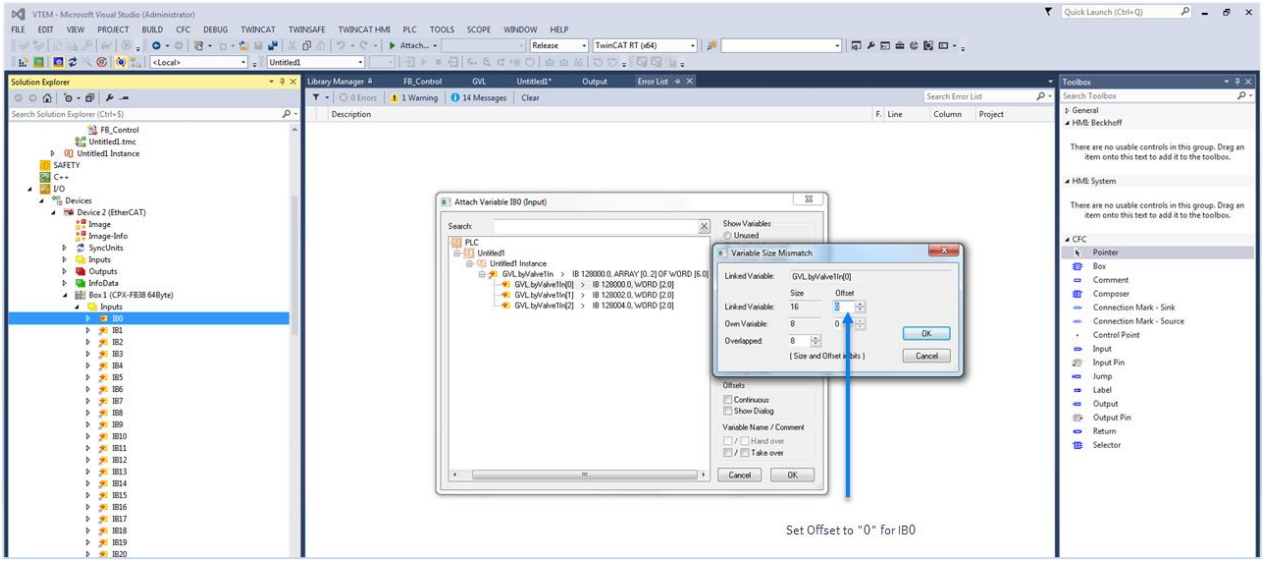
Link the I/O byte of the first VTEM valve declared in GVL:

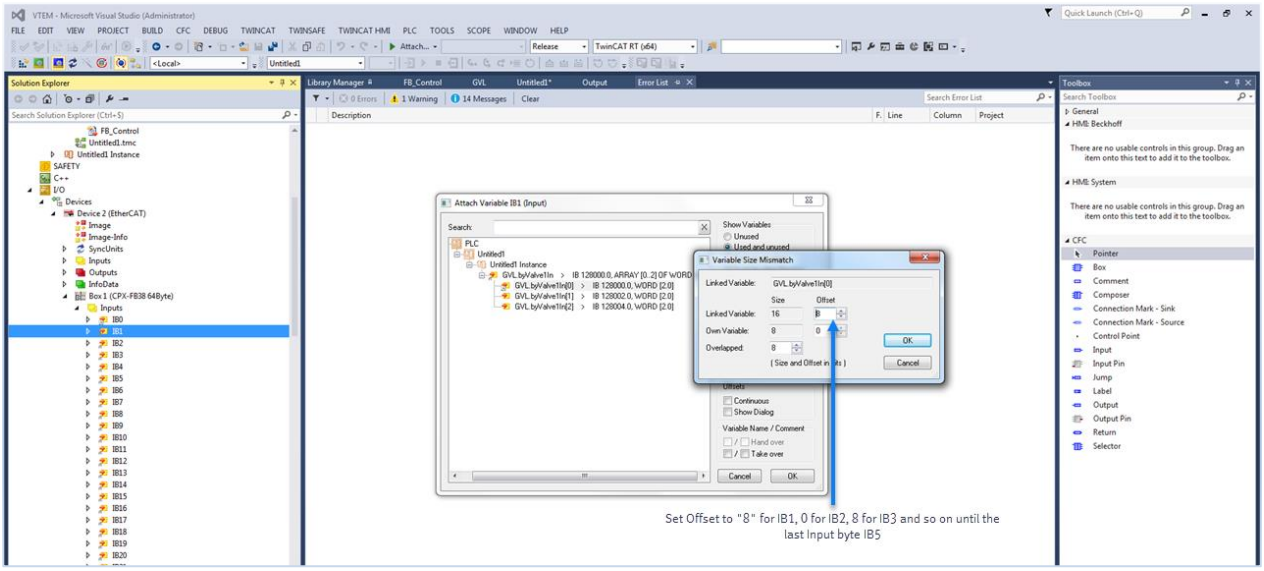


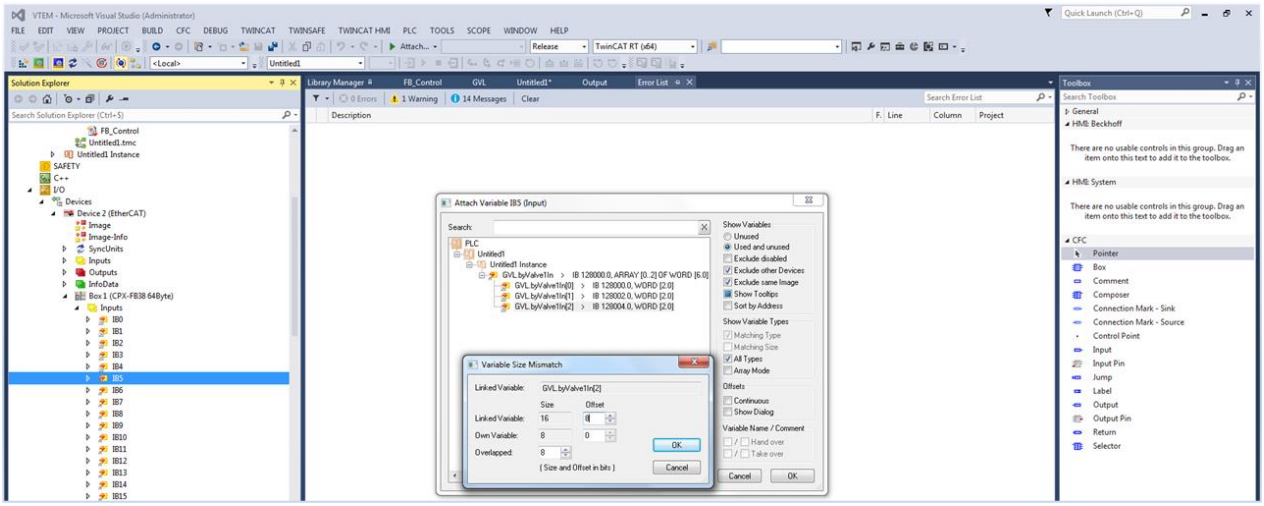
3.8 Link I/O bytes of the CPX-FB37 to the first VTEM valve



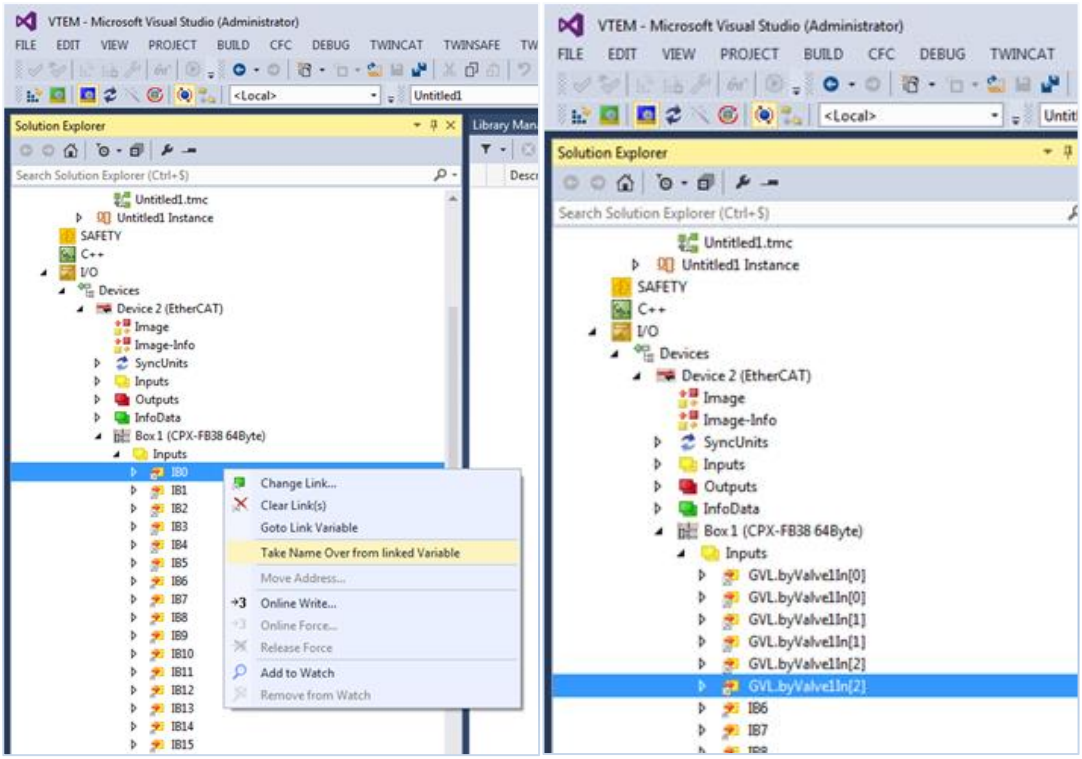




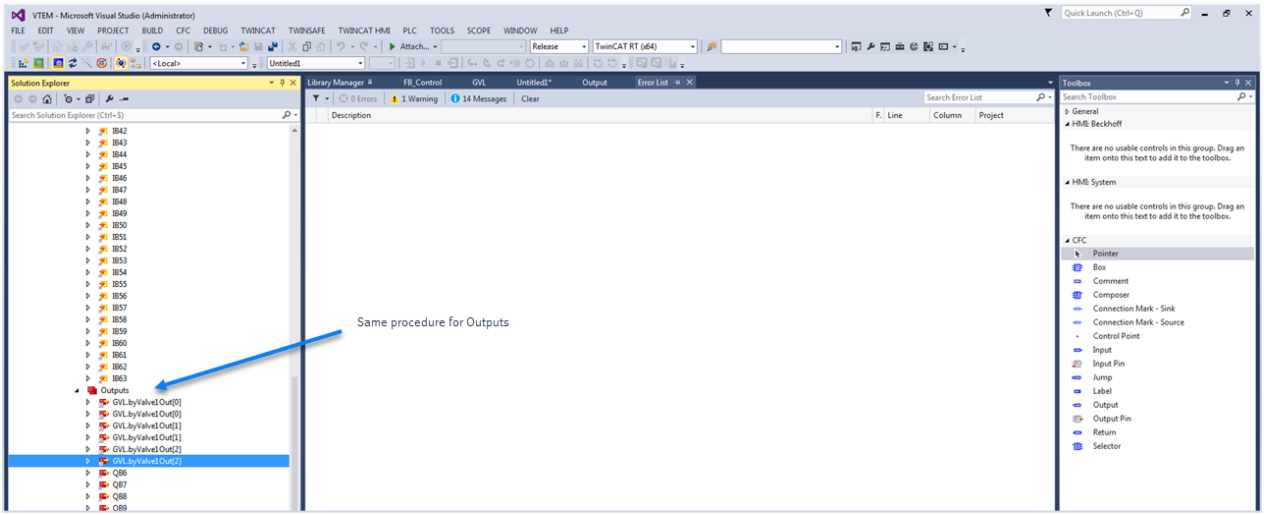




You can also take the name from the linked variable (GVL in this project):

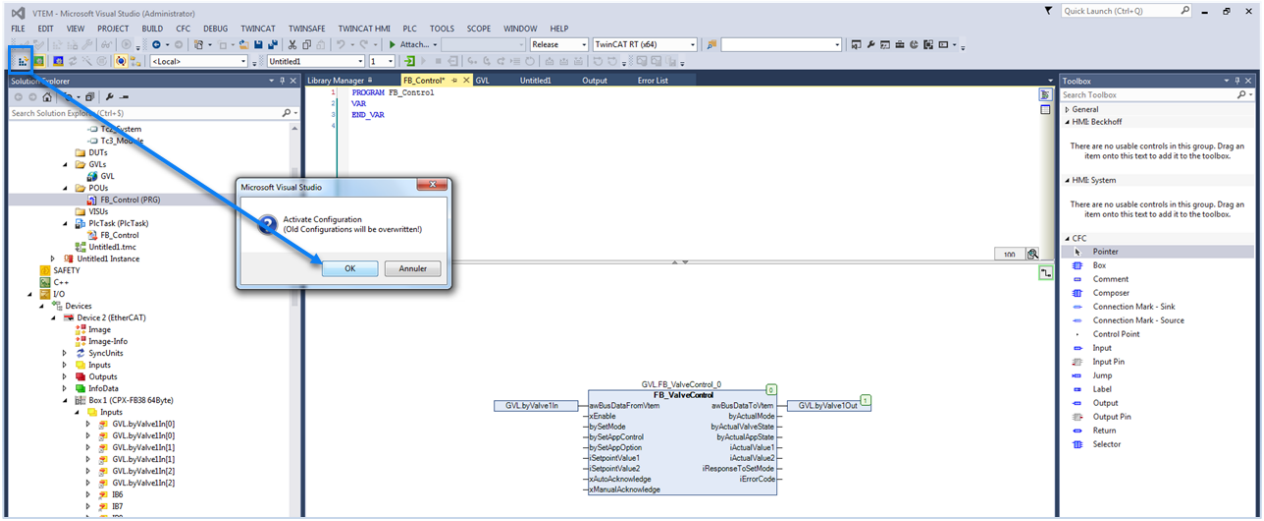


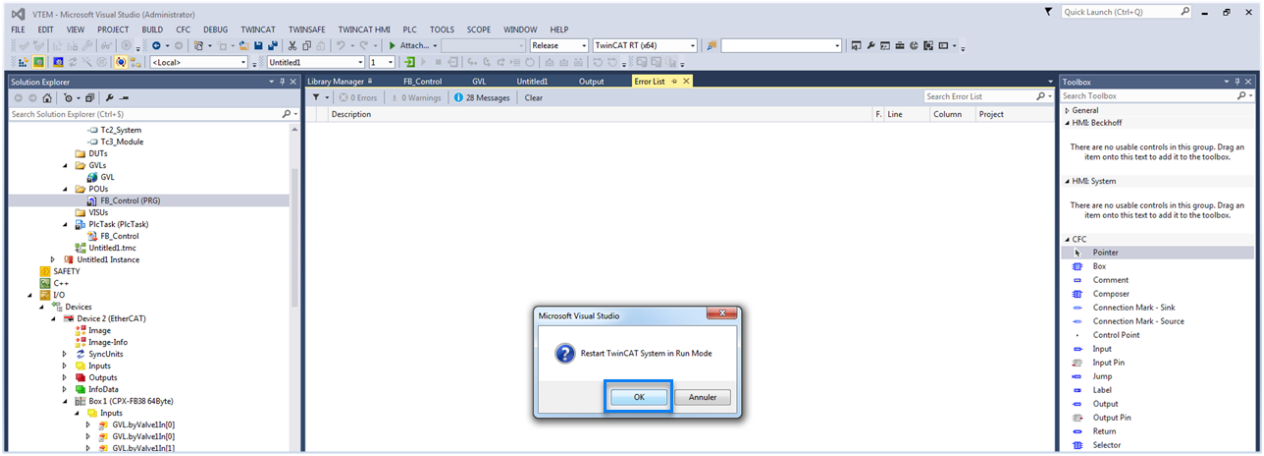
Repeat the same procedure for Outputs bytes:



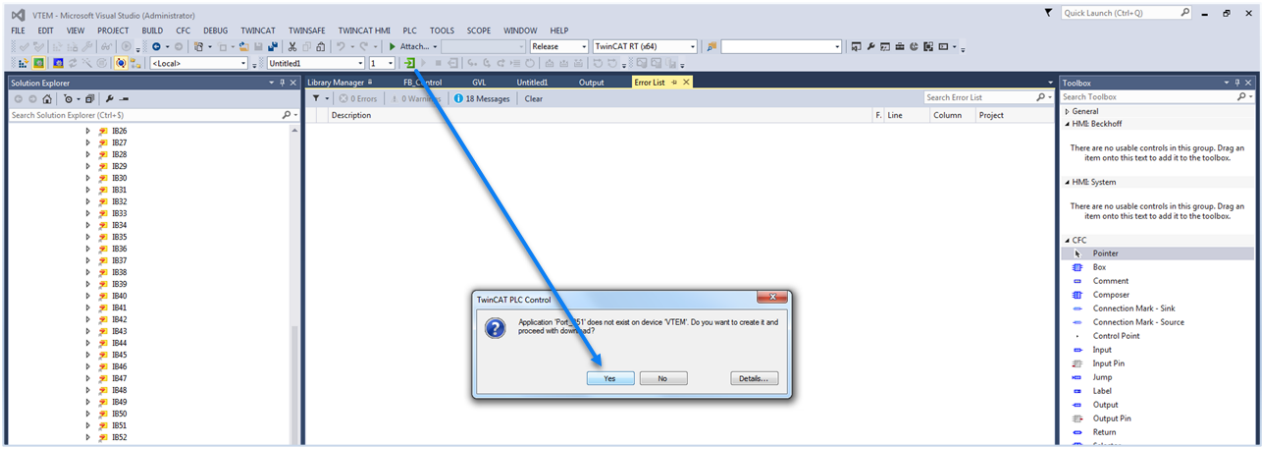
4 Run your project

4.1 Active configuration





4.2 Download to PLC and run your program



Switch PLC in Run mode: