Product	AMD	Type/Series	C/CP2000	Appl. Note Nr.	Delta C2000
Issued by	DEN	Author	Marcel Dorti	Release Date	May , 2016
Title	Delta C2000 & Siemens S7 1500/1200 - PROFIBUS communication.				

Devices and special tools/equipment

- ✓ Delta C2000
- ✓ CPC Communication Card
- ✓ S7-1500 with Profibus
- ✓ TIA PORTAL V12/V13
- ✓ Profibus DP cable with connectors and resistors

Test setup
N/A

CONTENTS:

1 Introduction

PROFIBUS DP is a widely used layer 2 industrial network protocol created by Siemens and a default protocol in many Siemens automation products, in this paper we will learn how to exchange data between Delta C2000 drives and S7 1500 PLC through PROFIBUS Network. On the first part hardware setup of it will be covered, including GSD installation. After that we will check how to communicate through PZD (cyclic data), memory addresses, words and bits on both sides of communication with practical examples of C2000 Delta drives, then an example will show how to address PKW (acyclic data) for parameter addressing.

Figure 1 - Topology of a Profibus Network with the Master and 2 Slave peripherals.

2 Connecting Hardware

In order to use Profibus on C2000 you need to install a **CMC-PD01** Communication card on it. Insert it on the communication card slot on the right side of the board panel of the drive according to the picture. In order to avoid electrical shocks you should do this with the equipment turned off, switched off from mains to the drive and switched off from control (24V) voltage to the PLC.

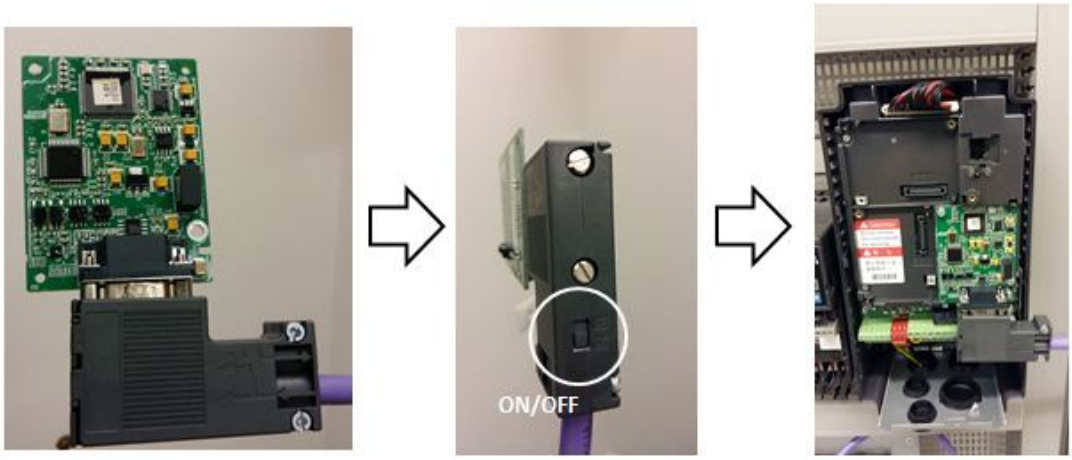


Figure 2: Screw the connector into the CMC-PD01; Check ON/FF status of Profibus network, insert it and screw it to the right side of the board panel. If your C2000 in the network is ***not the last or the first****, leave resistor* ***OFF****. This will make no difference on a bench test, but may cause* ***a lot of problem if done incorrectly in a field network****. Use a terminal screw drive to remove the board, no force is required to remove the board, just use the screw drive to move the side brackets.*

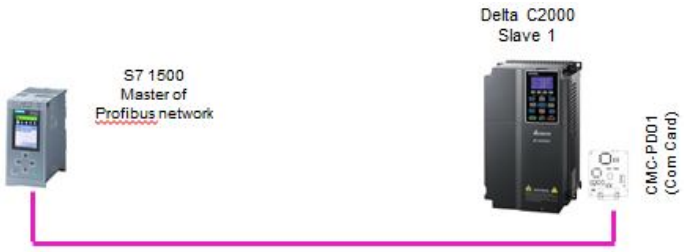


Figure 3 - Example of our first Profibus network example with S7 1500 Master and Slave.

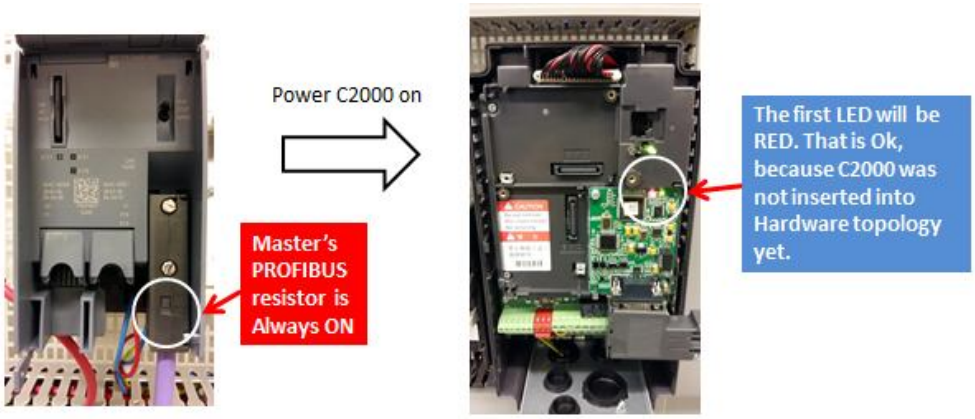
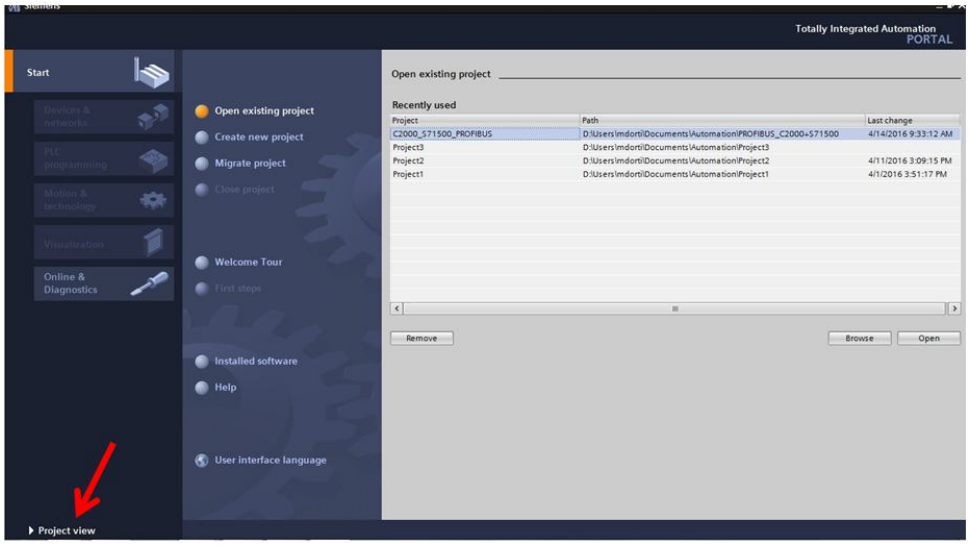


Figure 4 - Connect the Profibus Cable with the resistor ON to the Master and then, power on Delta C2000

3 Communicating to S7 1500 / First download

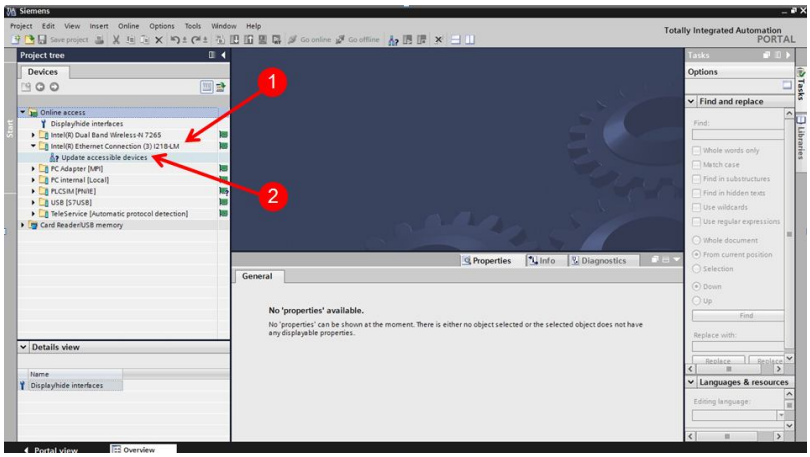
If you already have a project and just wants to insert C2000 in your already set network, **you can skip this chapter**, since this chapter will teach you how to assign an IP to the S7 1500 and download your first hardware configuration into the PLC, usually a new one.

New S7 1500 PLCs CPU do not come with a valid IP address (0.0.0.0) from factory, they are only recognized by their MAC address. So we should connect our Ethernet cable CAT5 (no peer-to-peer), a regular blue Ethernet cable will do it. Connect it to the PLC and then to the computer's Ethernet card.



On the title screen, before creating a project we should do the **online setting**, so click on **Project view.**

Figure 5 - First screen of TIA PORTAL



1 – Find your Ethernet Card.

2 – Click on **Update accessible devices.**

TIA PORTAL takes long time for online diagnostics, take some time while it discovers the PLC in your network.

Figure 6 - Online diagnostics screen of TIA PORTAL

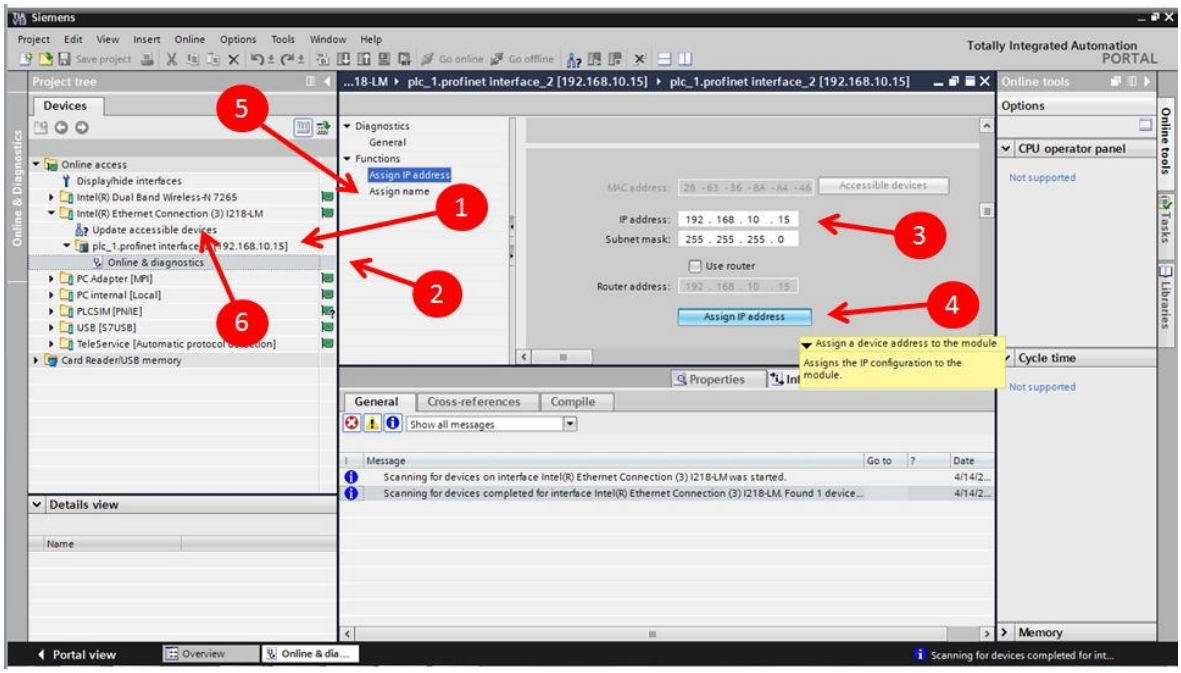


Figure 7 - TIA online settings

1 - PLC discovered

2 – Click on **online settings**

3 – Assign an IP and a proper Sub-net mask

4 – **Click on Assign IP address.** If you just assign the default Address, TIA will change the address of the Ethernet card ir order to be compatible with the PLC default IP address and it will ask you to do so, put YES when it does.

Note: Many of the problems with communication between computer and PLC happen because IP address of the computer is not compatible to the one set in PLC. Ex.: If you have 192.168.18.10 address on the PLC, Ethernet network card should be 192.168.18.XX until 255. Never use .1 because this is usually the gateway of the networks. This is done automatically with V13, older versions of TIA may require you to do it manually.

5 – Assign a name, **Click on assign name,** S7 Ethernet port needs a PROFINET name to be assigned, (scroll down) check the project to name it, it is not easy to change the name after the programs are already running, if you are only testing it leave the default name.

6 – Click on **Update Accessible devices** again, you should see the IP address of the PLC in brackets on the side of the name. Now your Profibus Master has an Ethernet IP address assigned and you can access it from your computer or from any internal network.

4 Creating a project and inserting a GSD file

4.1 Creating a project

You need to create a project, where all your hardware configuration and software will be available, to do this, on the title screen of TIA portal, 1 - insert relevant data, 2 - click on New project, 3 – Click on project view. If you already have a project, skip this page.

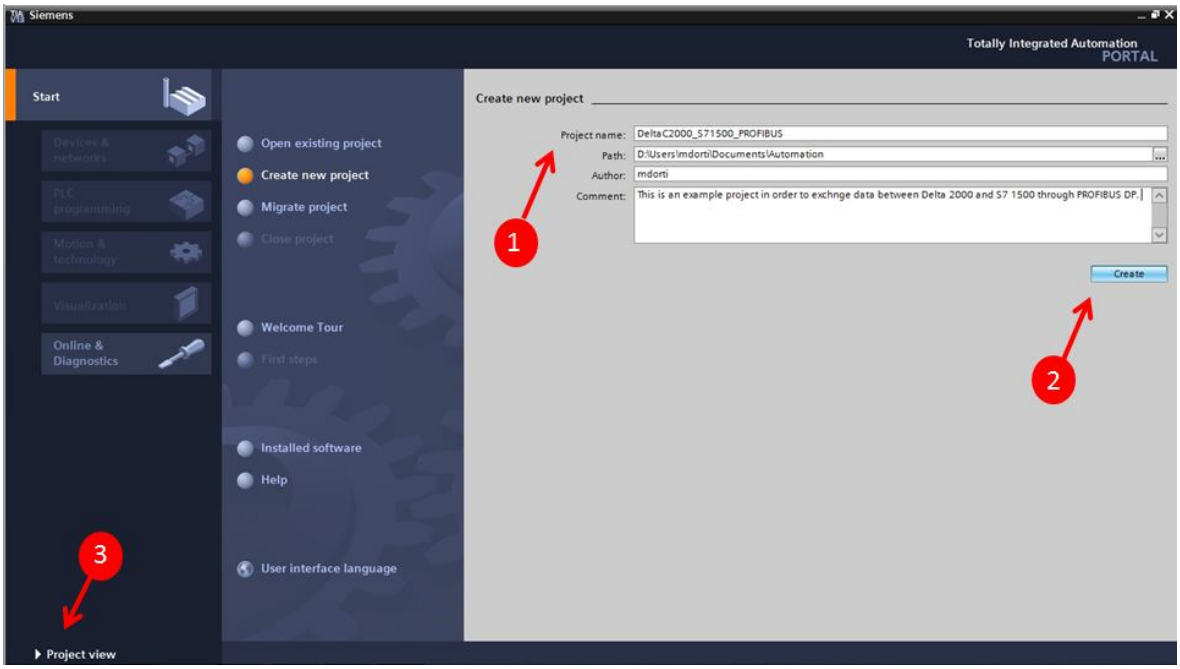


Figure 8 - Creating a project

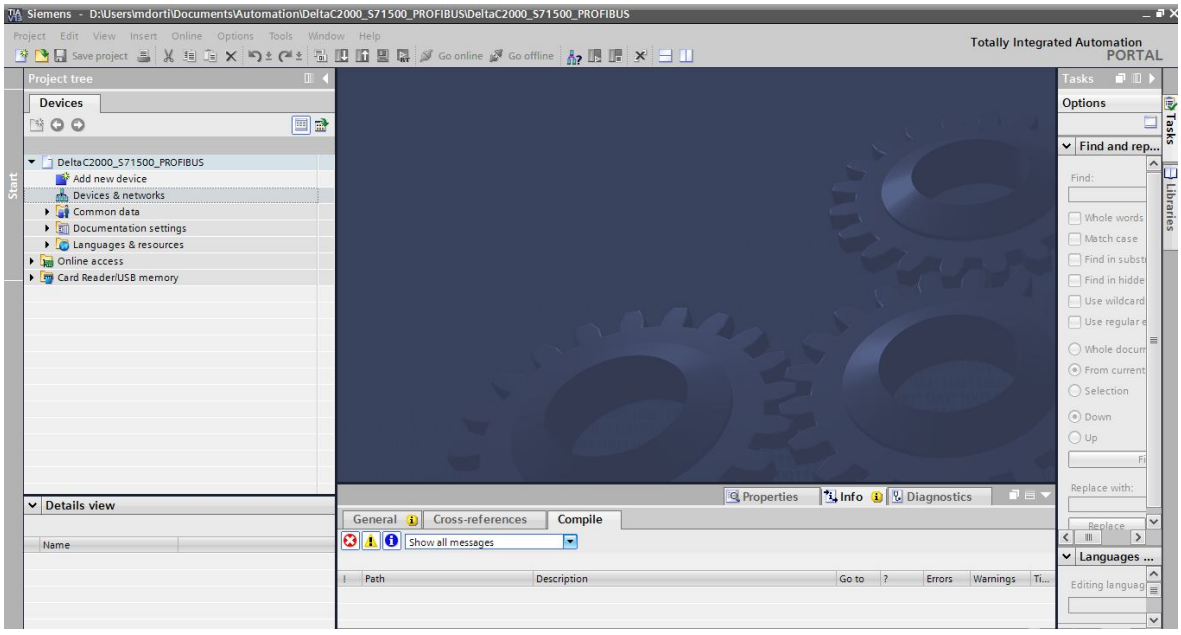


Figure 9 - Initial Start Screen of TIA PORTAL with project created, with IP set, ready to set up hardware

4.2 Downloading GSD

For a Profibus Master to communicate with its slaves it requires a system file of its hardware called GSD (General Station description), this file contains the instructions to read and write data from a master to a slave. Delta C2000 GSD can be downloaded at Delta EMEA website and click on **Products - Industrial automation,** go down through the scroll bar to **Download Center.**

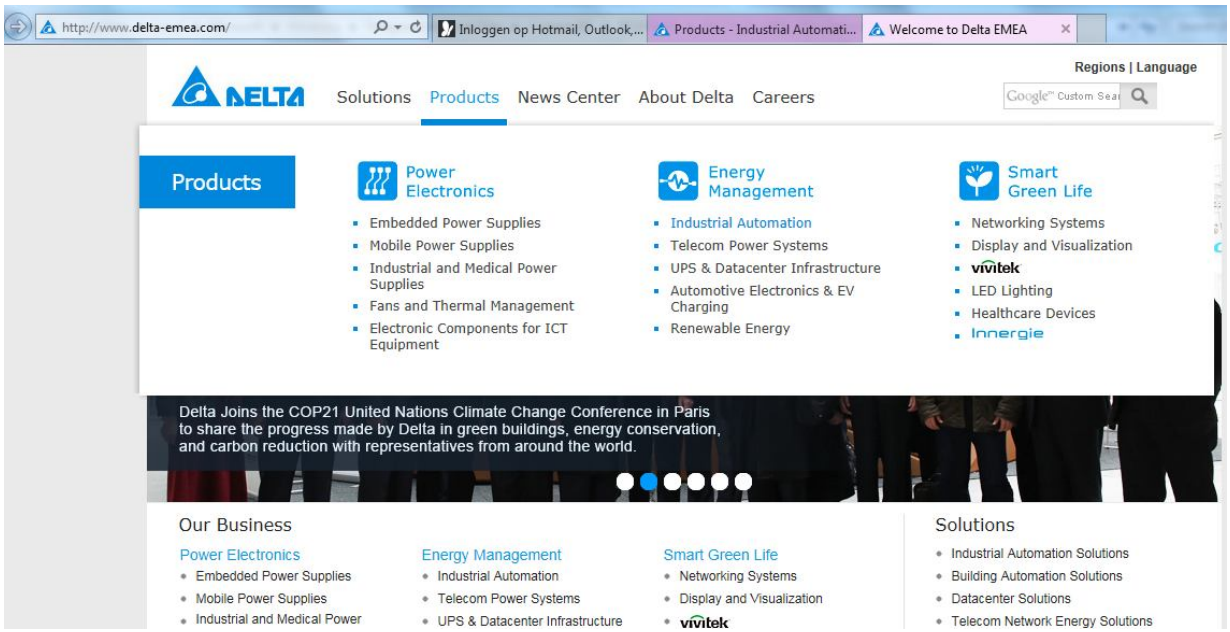


Figure 10 - Delta EMEA website - here you can find information about Delta Products

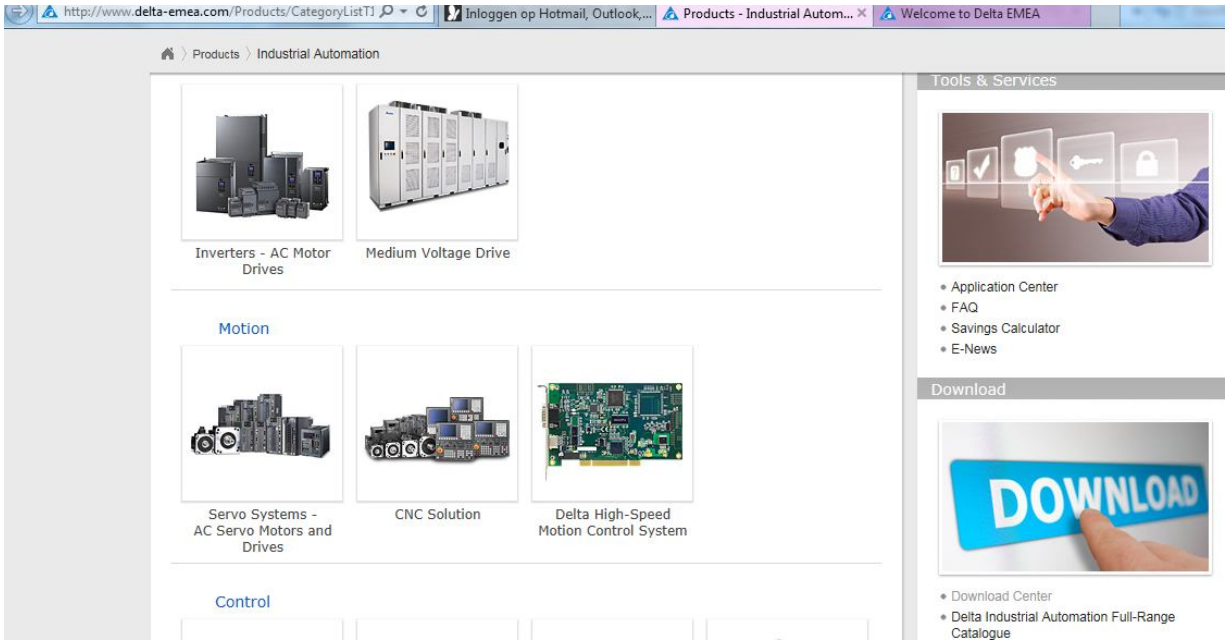
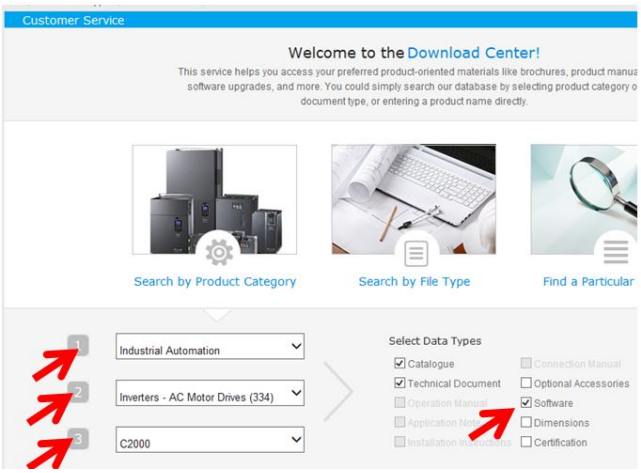
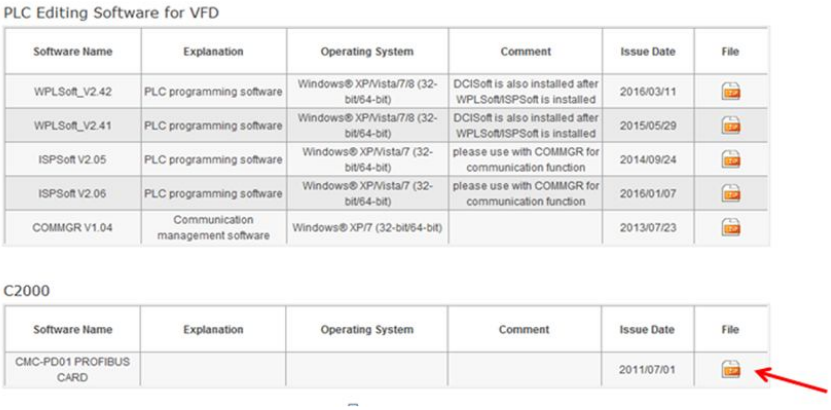


Figure 11 - Industrial Automation Products - Download center.



Look for Industrial Automation, **Inverters – AC Motor Drives (334)** and **C2000**, click on software.

Figure 12 - Download center options



PLC Editing Software for VFD

Software Name	Explanation	Operating System	Comment	Issue Date	File
WPLSoft_V2.42	PLC programming software	Windows® XP/Vista/7/8 (32-bit/64-bit)	DCISoft is also installed after WPLSoft/ISPSoft is installed	2016/03/11	
WPLSoft_V2.41	PLC programming software	Windows® XP/Vista/7/8 (32-bit/64-bit)	DCISoft is also installed after WPLSoft/ISPSoft is installed	2015/05/29	
ISPSoft V2.05	PLC programming software	Windows® XP/Vista/7 (32-bit/64-bit)	please use with COMMGR for communication function	2014/09/24	
ISPSoft V2.06	PLC programming software	Windows® XP/Vista/7 (32-bit/64-bit)	please use with COMMGR for communication function	2016/01/07	
COMMGR V1.04	Communication management software	Windows® XP/7 (32-bit/64-bit)		2013/07/23	

C2000

Software Name	Explanation	Operating System	Comment	Issue Date	File
CMC-PD01 PROFIBUS CARD				2011/07/01	

Go to C2000 and Download PROFIBUS GSD, the file name which contains the GSD of the file is **DELA08DB.gsd** unzip it and transfer to a folder in a **known location**.

4.3 Installing the GSD

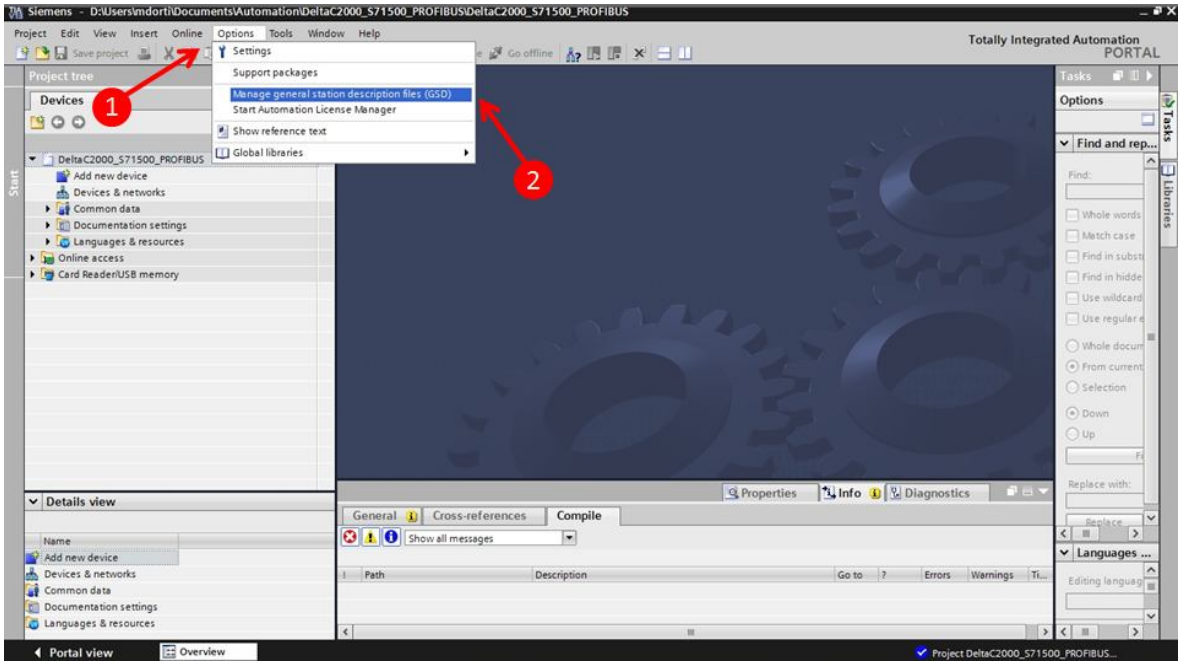


*Figure 13 - 1 - Click on **Options**, 2 - **Manage General Station Description files**.*

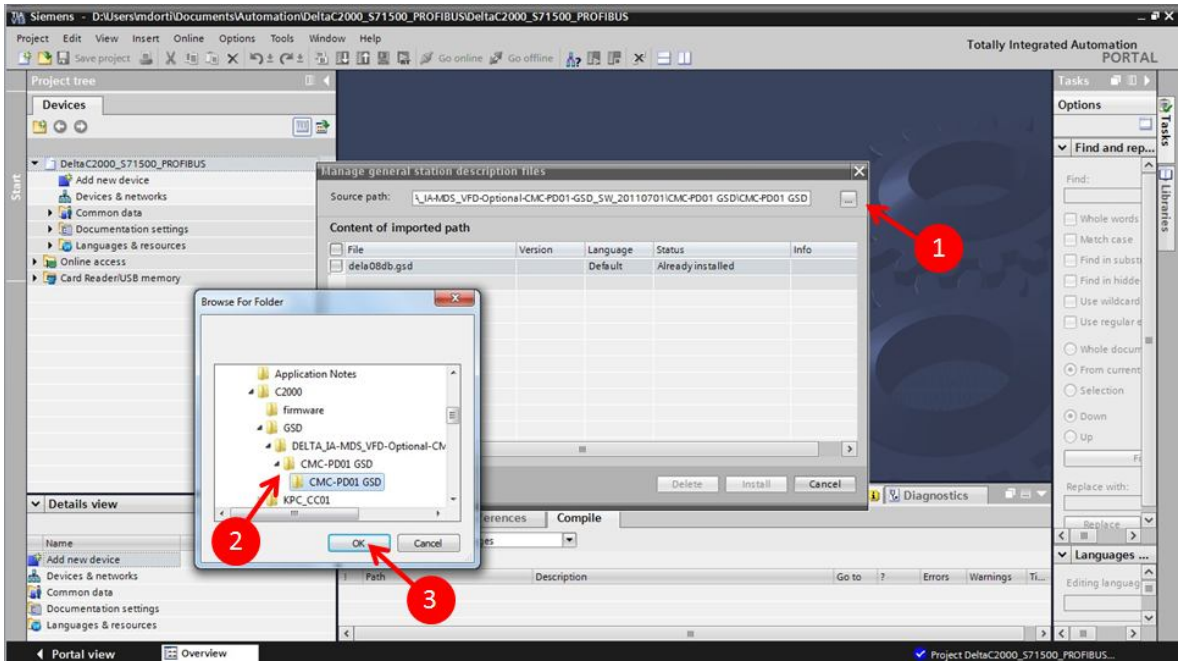


*Figure 14 - 1 - Click on the button to assign the local on GSD file you downloaded, 2 - Specify the folder, 3 - Click on **OK.***

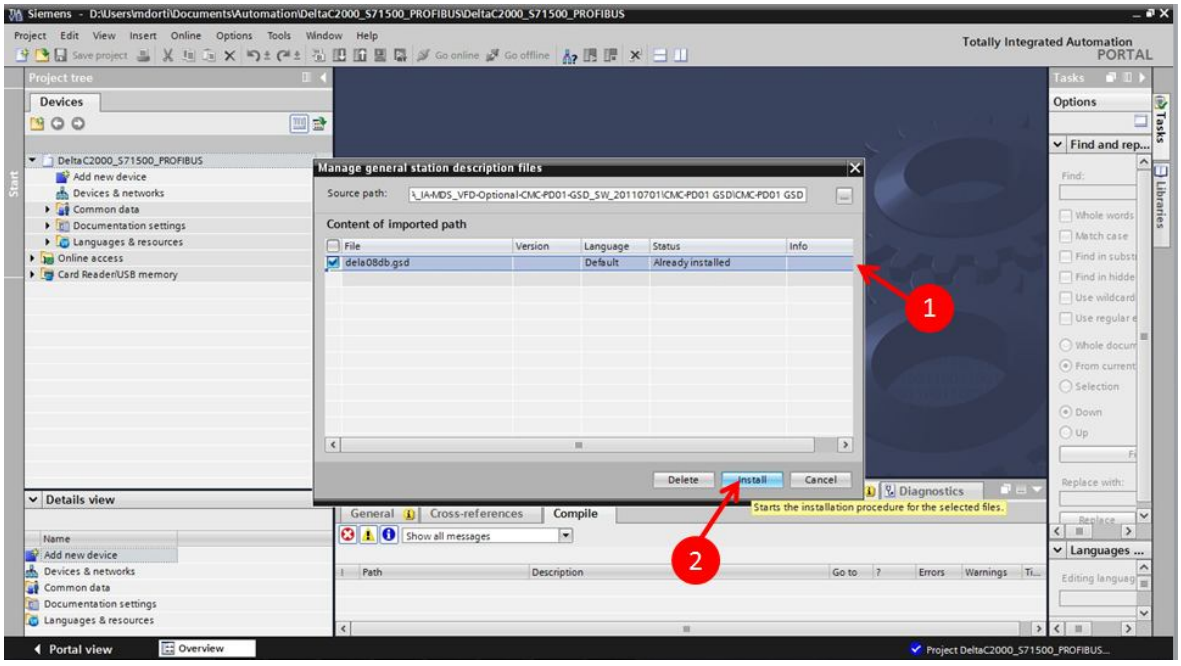


*Figure 15 - 1 - Click on the the GSD file, 2 - Click on **Install.***

After you do this, Wait for some time, TIA will install the GSD. TIA portal will show a message displaying **"Installation completed successfully"**, then click on **Close** and it will show a message: **"Updating hardware catalog".** TIA portal will restart automatically then.

Once you have installed the GSD of Delta C2000, restarted TIA and opened your project again, you can check it in the hardware catalog, to do this go **to 1 - Devices and networks, 2 – Catalog, 3 - other field devices – PROFIBUS DP – Drives, 4 - Delta Electronics** and check the GDS there **PD01**.

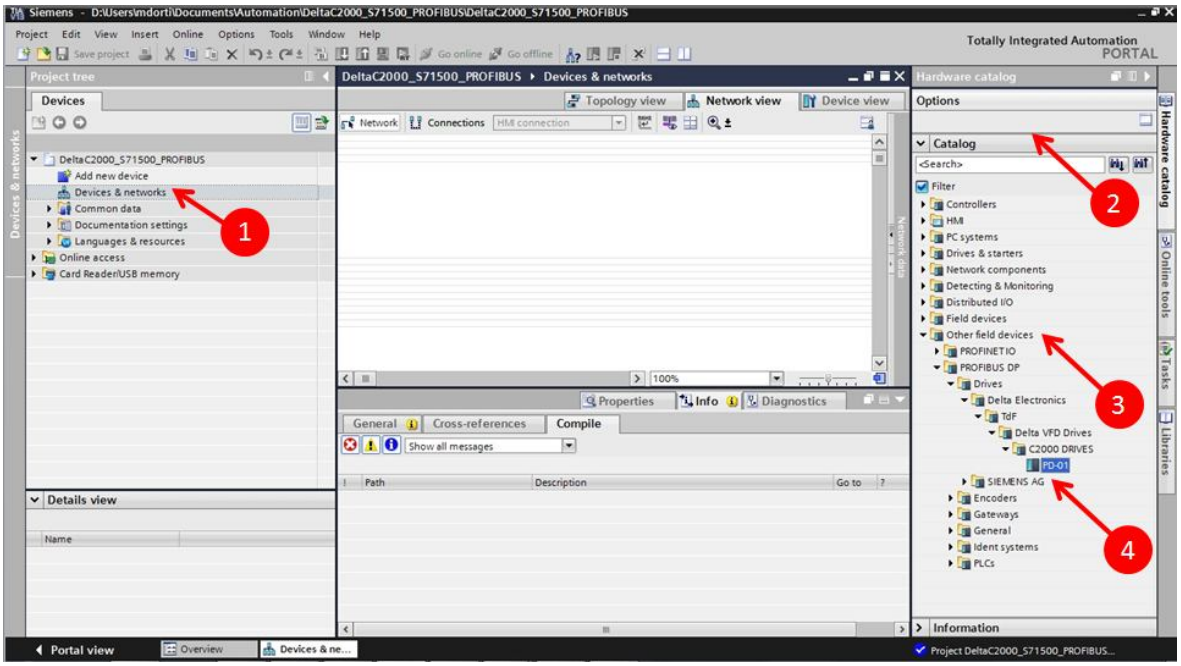


Figure 16 - CMC - PD01 Delta C2000 installed successfully in the hardware catalog

If you can check that, congratulations, you have successfully installed Delta C2000 GSD. Now let's proceed to set up our hardware configuration.

5 Setting up Hardware config. of the Master (S7 1500)

Once you can connected via Ethernet, created a project and was able to install the GSD, now we will start the real fun of network protocols. If you already have a project skip this, since we will check how to set S7 1500 and Delta C2000 into hardware configuration. Siemens S7 is composed structurally by two main components:

Hardware topology: All the connections and hardware settings of PLC, Drives, cards and accessories connected to the PLC.

Programs: Main routine (OB1), Functions Blocks (FB), Functions Charts (FC) DataBlock (DB). Main routine is where the program runs, FC and FB are field blocks with determined functions, DB are memory places to store data. For more details check S7 1500 manual.

If you have a new S7 1500 and never used it, just assigned an IP it probably is in STOP mode (yellow light), and it needs a first hardware download, to identify its own type of PLC into its topology. You can do this manually, but connect and permit the PLC to check it is much easier.

Go to **1** - **Add new Device**, **2 - SIMATIC S7 1500**, **3** - **Unspecified S7 1500** – **6SE7 5XX-XXXX-XXXX** and then click **OK**.

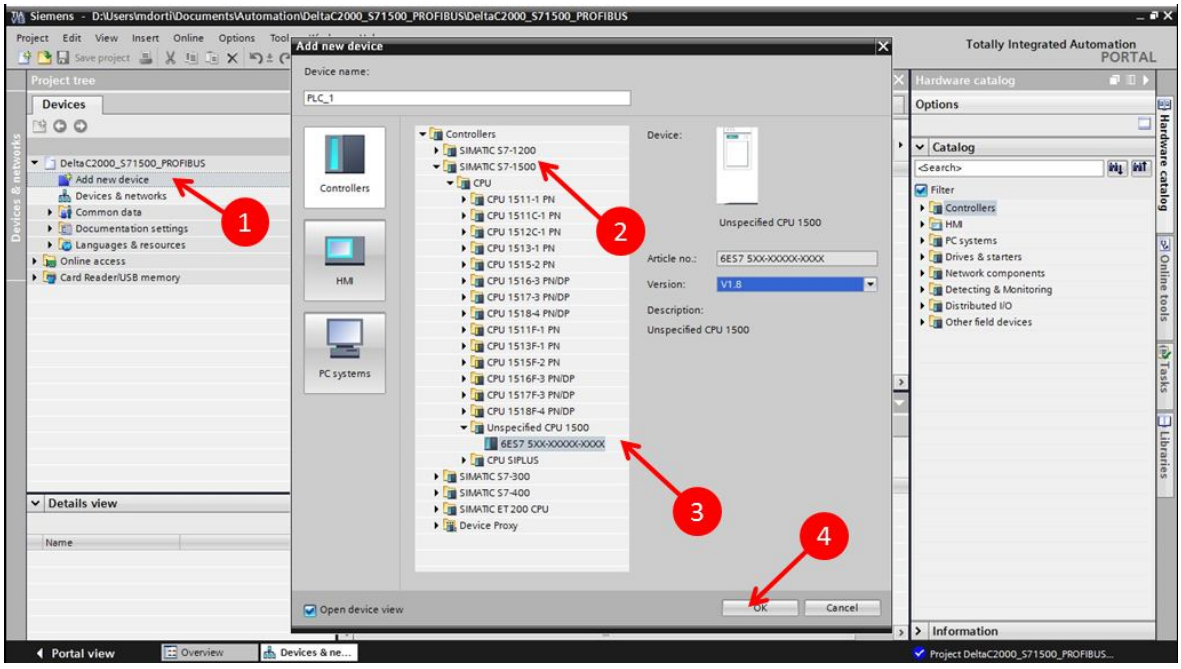


Figure 17 - Adding a S7 1500 into Hardware

After that, there will be a time for the PLC to be inserted, wait patiently and then, it will come to a screen where the PLC S7 will be in blank. 1 – S7 1500 Unspecified CPU inserted, 2 – Click on the **Detect** the configuration.

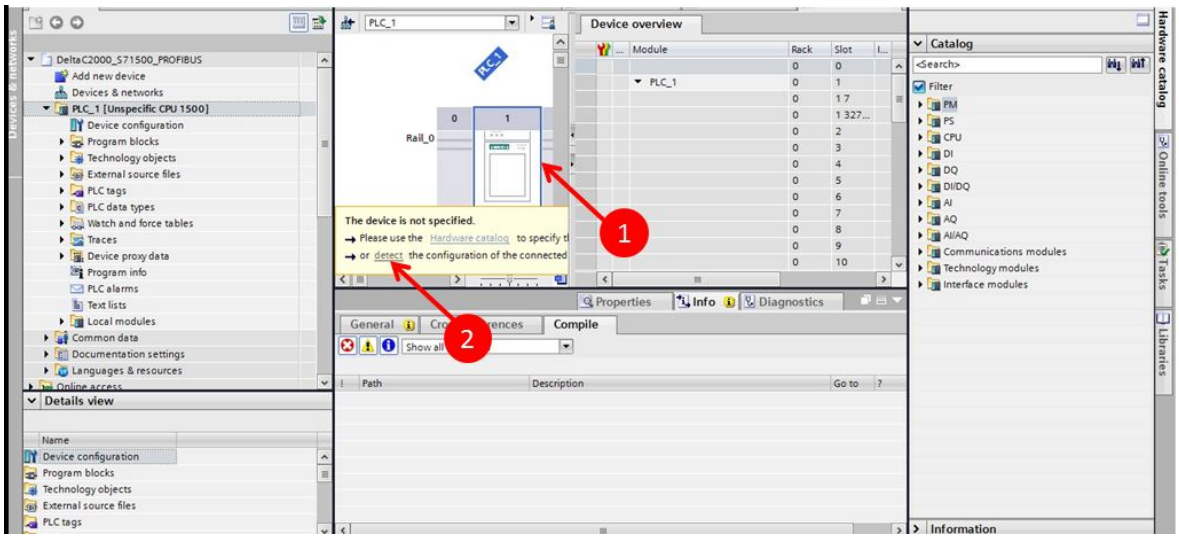


Figure 18 - S7 1500 unspecified CPU inserted from hardware catalog

Once you do it, you will be transferred to the connection screen, where you will assign an Ethernet card for communication and identify your PLC. 1 – **Select your Ethernet card**, if you have any IT policy on this computer, disable the Windows firewall if any problem occurs. 2 - Click on **Start Search**.

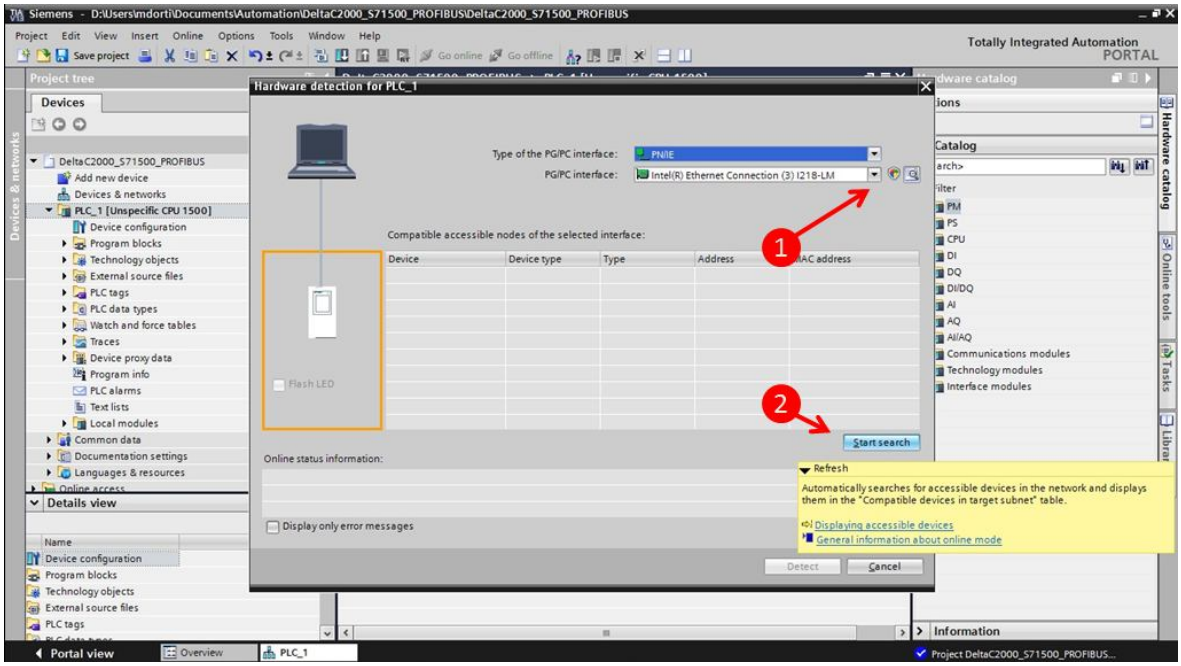


Figure 19 - Connection Screen of TIA Portal

Identify your PLC in the network and then, click on it and then click on **Detect**. To establish a connection through Ethernet TIA portal will assign an IP address which is compatible to the one in the PLC automatically, so click **Yes** and it will change the connection IP address of the computer, older versions of TIA may not have this resource, so you would have to change your Ethernet card address manually. After that, click on **OK** and proceed.

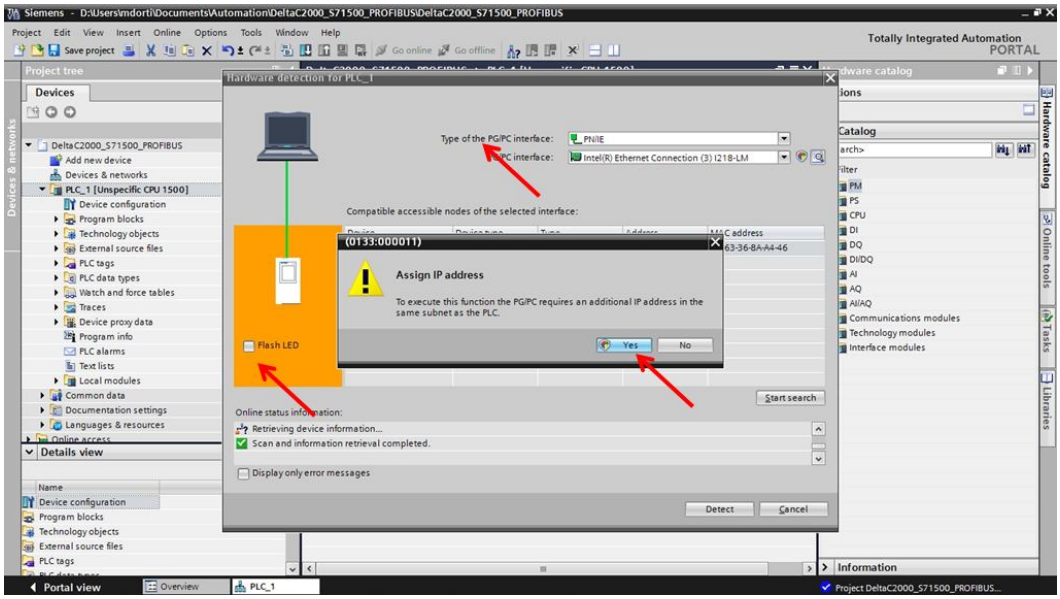


Figure 20 - TIA Portal Connection screen

Now, S7 PLC will take some time to recognize the CPU, wait patiently, when it detects the CPU, it is displayed on **Device View** a screen showing the CPU, with all specifications on **Device View** window.

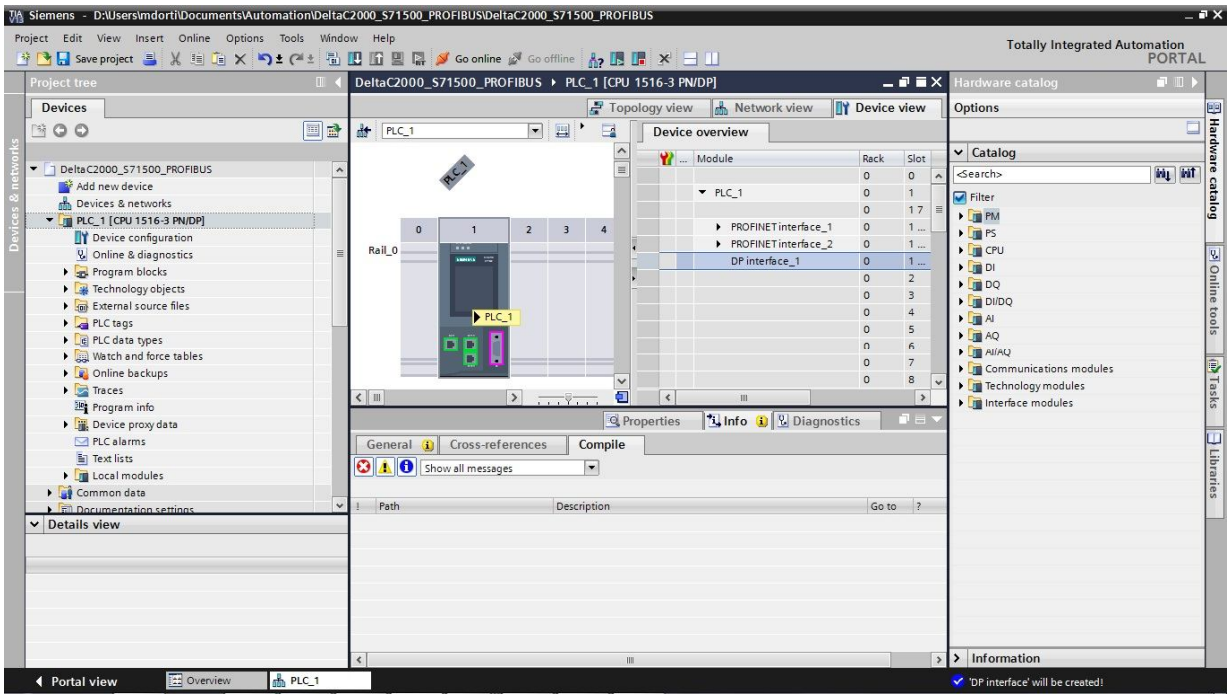


Figure 21 - Device screen with S7 1500 CPU recognized in Hardware topology

Now you have an S7 1500 CPU recognized and you can click on the network interfaces of it, click on the Profibus connection in order to check relevant data to the network, such as **Profibus address**, a network with equal Profibus Address will not work at all and there will not be any message telling you the cause, so **check the Profibus Address of the Master** of the network not to put the same address on the **slave peripherals**.

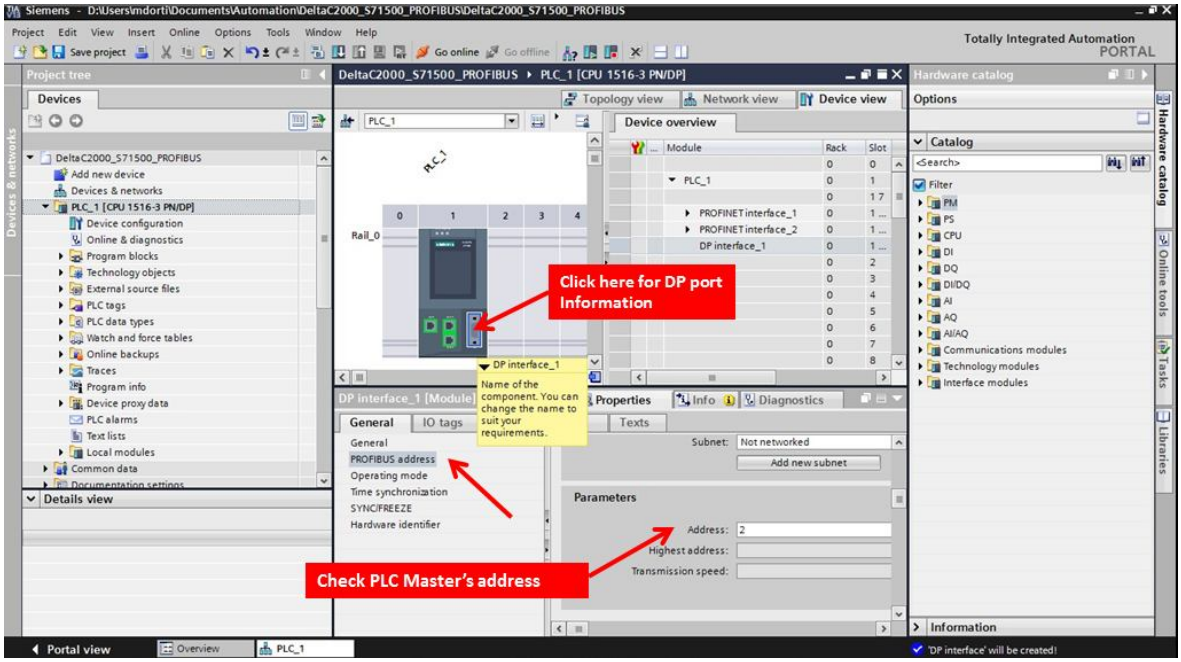


Figure 22 - Device view and General information of Profibus DP port, such as address

Now you already have an specified S7 CPU in your hardware configuration, you can compile the hardware topology and download to the PLC. So, go to 1 – PLC (left click), 2 – **Compile**, 3 – **Hardware & Software (rebuild all).** After the first, you can just compile changes.

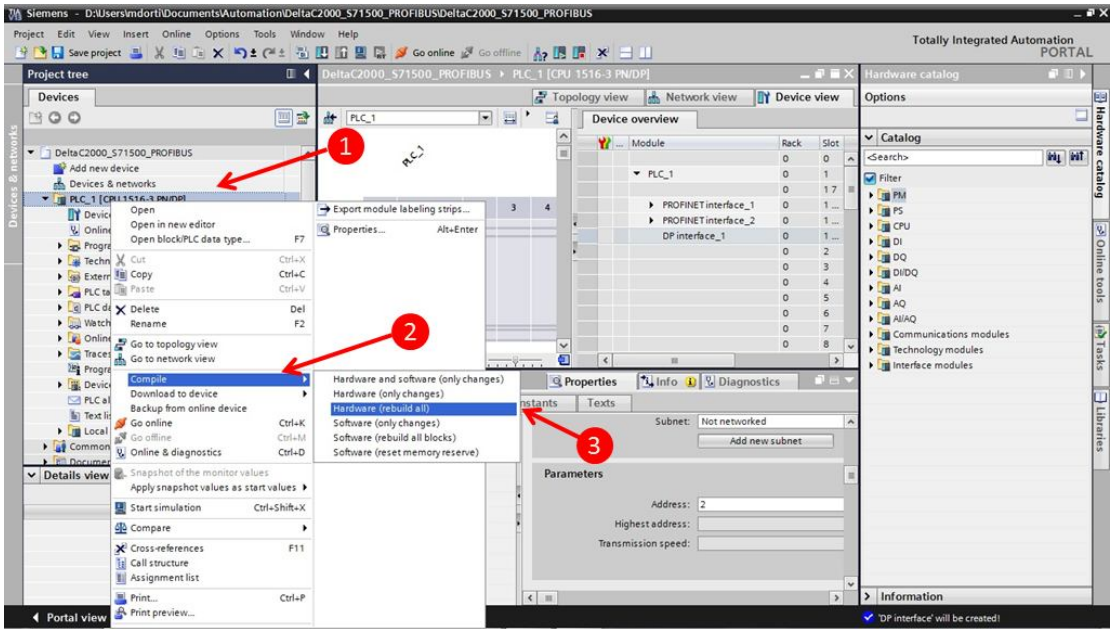


Figure 23 - Compiling your first Hardware setup

If it reports just one warning, it is ok, because we had not set any protection passport in the PLC, that's why this alarm appears. Now, time to download to the PLC, Go to 1 - **PLC (left click)**. 2 – **Download**, 3 – **Hardware and software** (only changes).

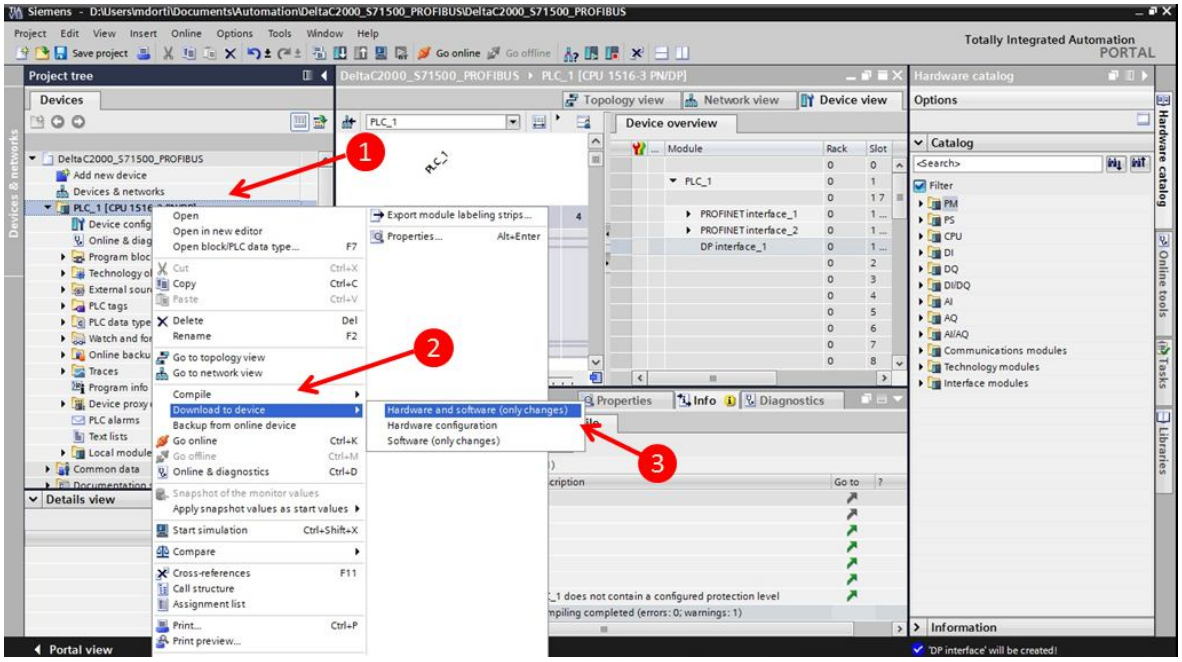


Figure 24 - Downloading the Hardware Configuration

You will be taken again to the connection screen, you probably already know what do to, identify your PLC – **Start Seach** and then after it is found, just click on **Load**.

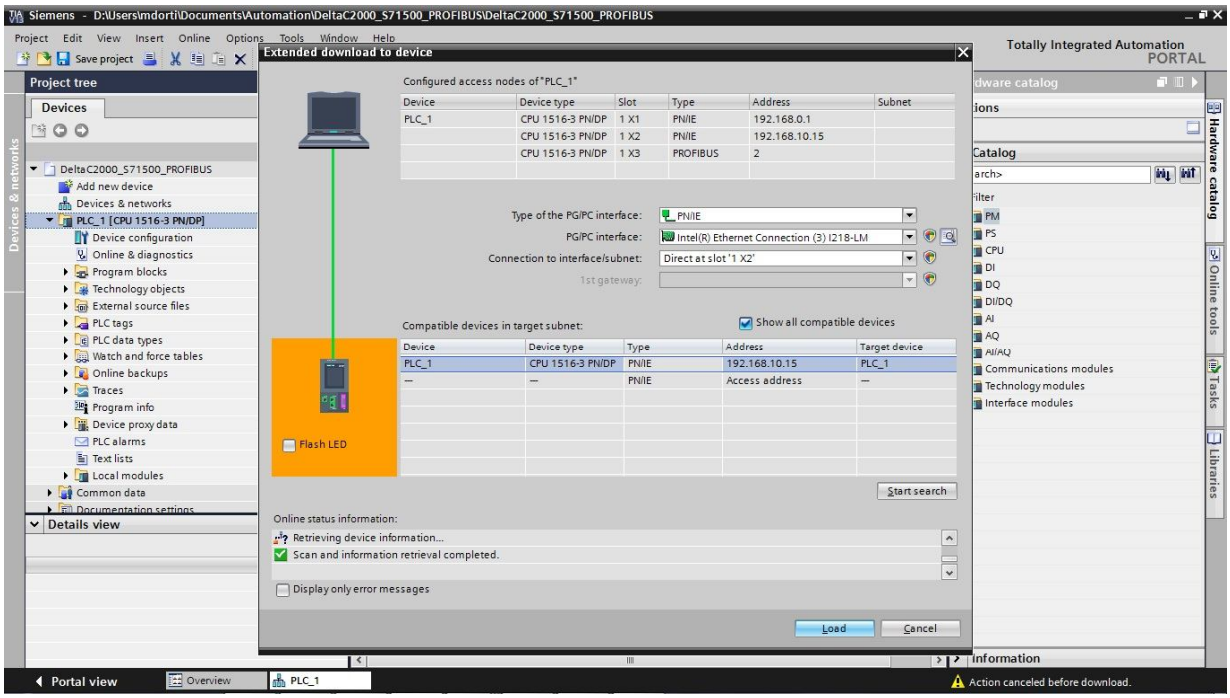
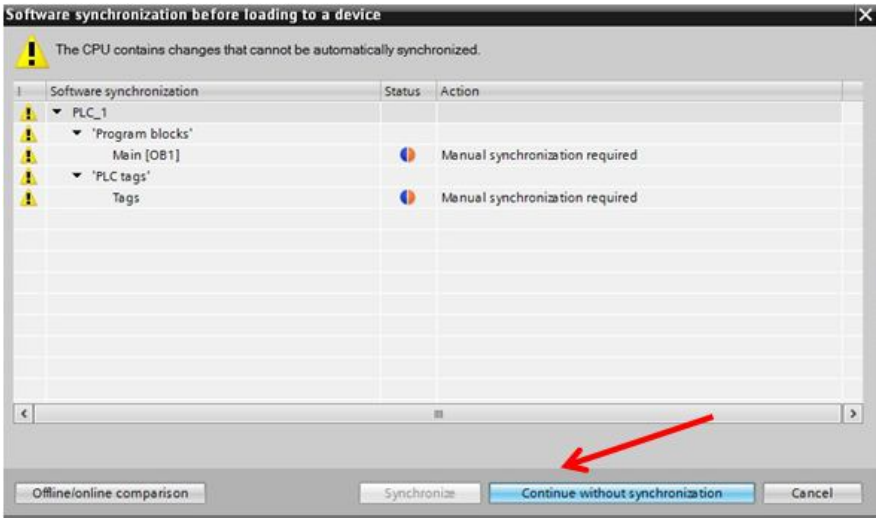
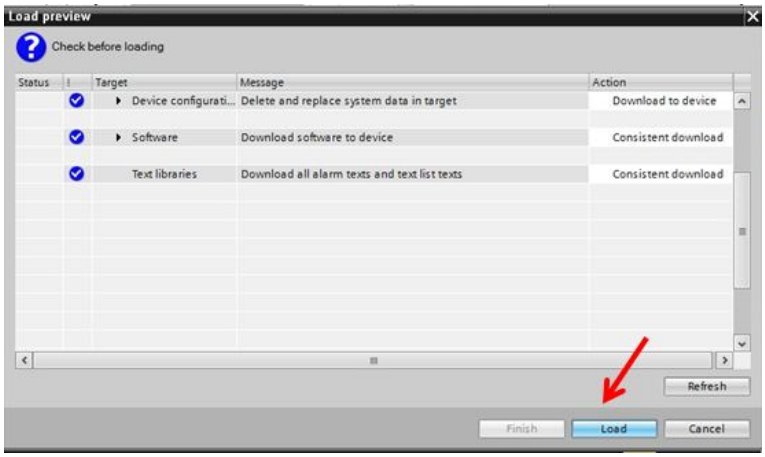


Figure 25 - Connection screen to download the setting to the PLC

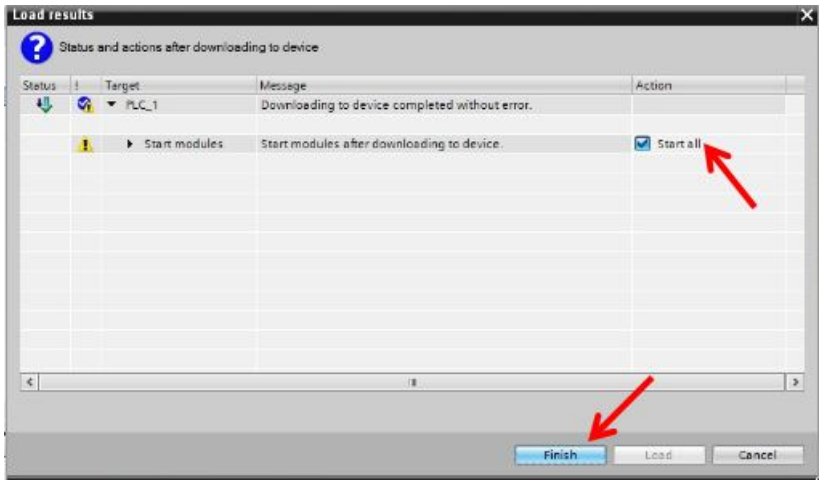
On the following screen, because you have no direct access to the memories of the PLC, it will tell you need to synchronize it, select **Continue without synchronization**.



Now proceed with **Load** it is just telling you how this download will be performed, and it will be done in consistent way, which means all data is going to transferred in a row, all together.



Click on **Load**.



Now TIA is telling you that, it will make the CPU go to **STOP** and then **Start all** again, If you are doing this in a running configuration, be aware that all processes controlled by the PLC will be stopped, and there's no way to change hardware online. Click on **Finish**.

Now, if your hardware download was ok. You should see this screen after it.

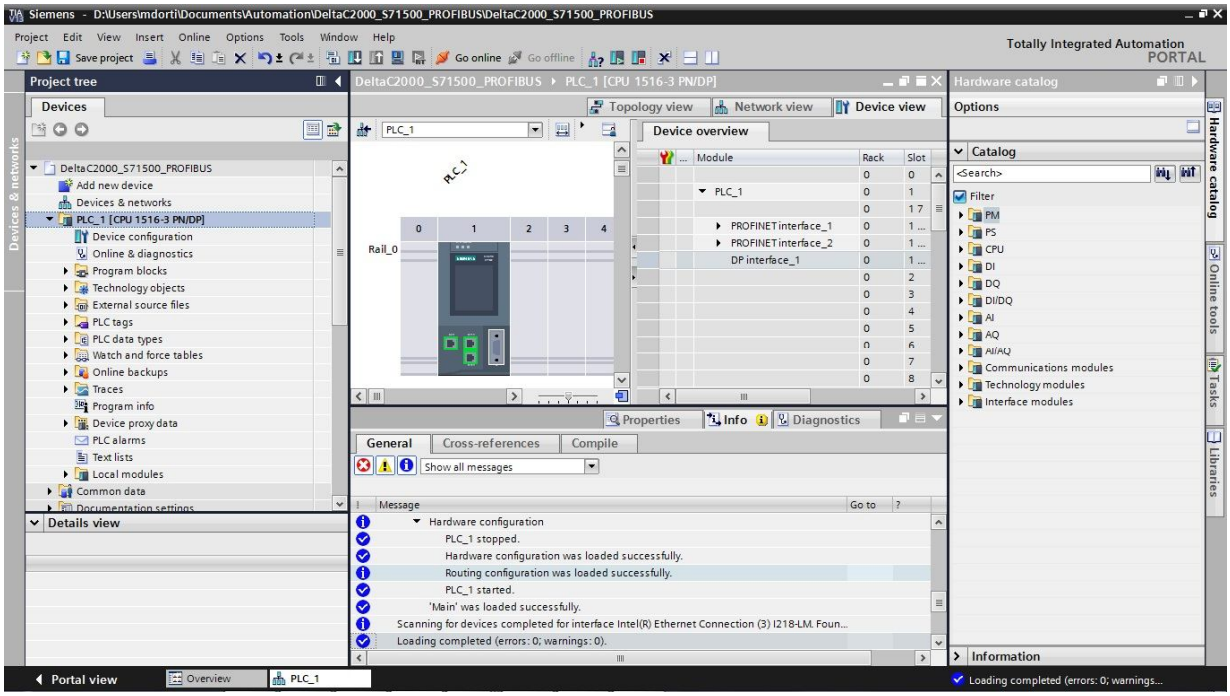


Figure 26 - Device view after hardware download.

Now, if you look at the LED indicator of the hardware, S7 1500 and C2000, it should be like this.

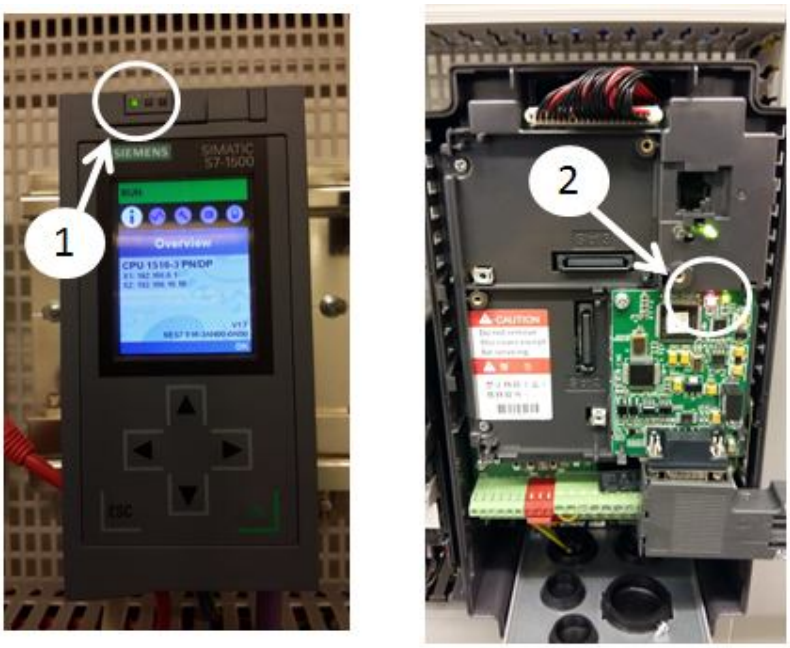


Figure 27 – 1 - CPU online (master), 2 - C2000 not in hardware (slave).

The red Led in Delta C2000 happens because the master cannot read this device and synchronize data with it, which is natural, since we did not declare it on Hardware configurator, and this is our next step.

6 Setting up Hardware config. of the Slave (C2000)

With the Master set, now we must set Delta C2000 in order to be part of the hardware topology, please, if you running a motor, commission the motor with the tuning application note in order to perform auto-tuning and load tests, check if it can run and increase/decrease frequency through keypad, here we are going to cover only data exchange of this configuration.

Change the following parameters in Delta C2000:

- ✓ 09 – 70 = 3 (address of the slave)
- ✓ 00 – 20 = 8 (Set frequency from Communication card CMP-PD01)
- ✓ 00 – 21 = 5 (Command from Communication card CMP-PD01)
- ✓ P 09-30 = 1 (60XX decoding)

Power off C2000, when Parameter **P 09-70** is changed it only takes effect on **the next power on**, and then **power on** again. Check **P 09-70** in order to check the address of Delta C2000. Naturally, you can do this for any address you prefer, but be sure not to repeat it and be sure to declare it on the hardware topology of TIA portal, as we will see further in this chapter. 1 – Click on Device view, 2 - double click on the DP slave, 3 – Click on **PROFIBUS address**, 4 – change the address to the one you set in parameter 09-70 of the C2000.

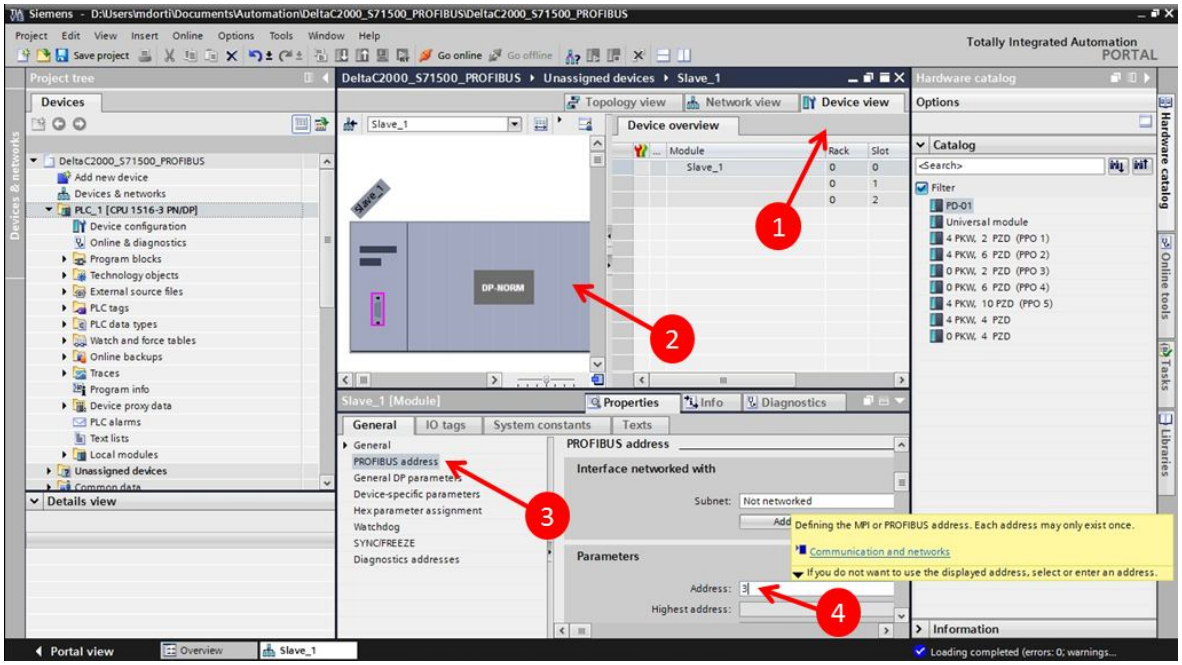


Figure 28 - TIA Portal device view in order to change PROFIBUS address

1 - Click on **Network view**, and then, on the **Profibus color connector** and **DRAG** a wire to the master PLC. When you finish it a cable connecting both peripherals will be shown, like the picture on Fig. 30. Then 1 - click on PLC_1 on the slave and then, 2 – define it as a Master, click on the pin of the **Highlight Master System**. The cable must be dotted and not continuous like in Fig 31. That means your Slave C2000 has a Profibus Master System. If it does not get dotted in magenta. It will not work.

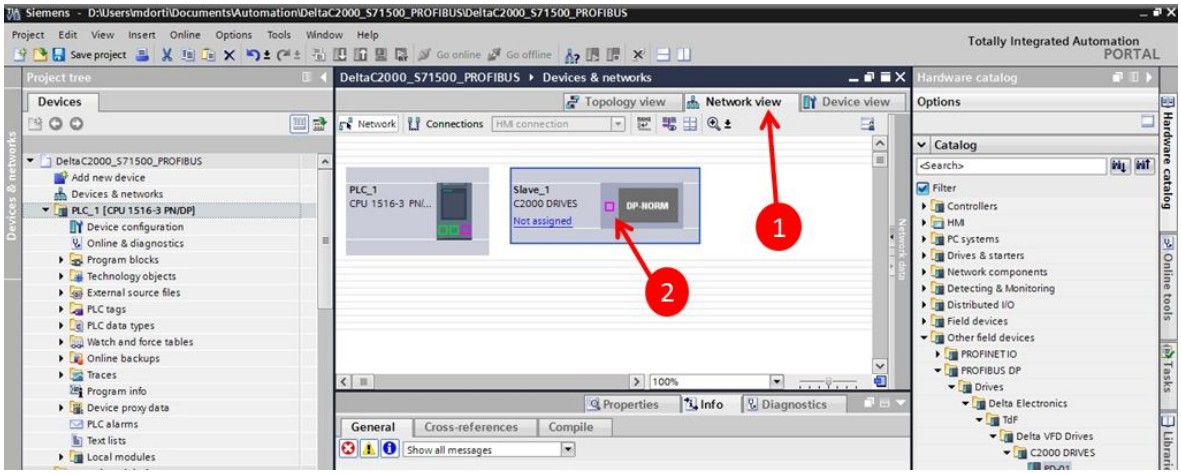


Figure 29 - Profibus connection

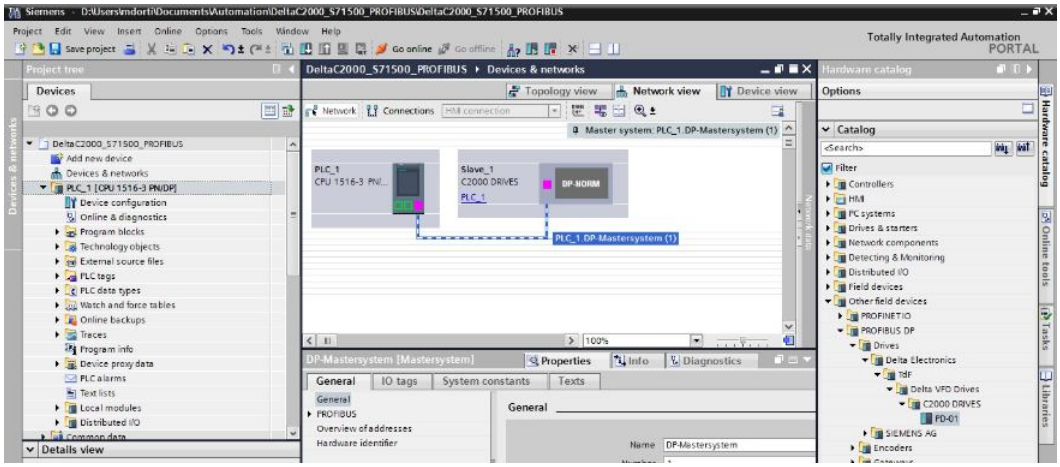


Figure 30 - Dragging a wire to the PROFIBUS master

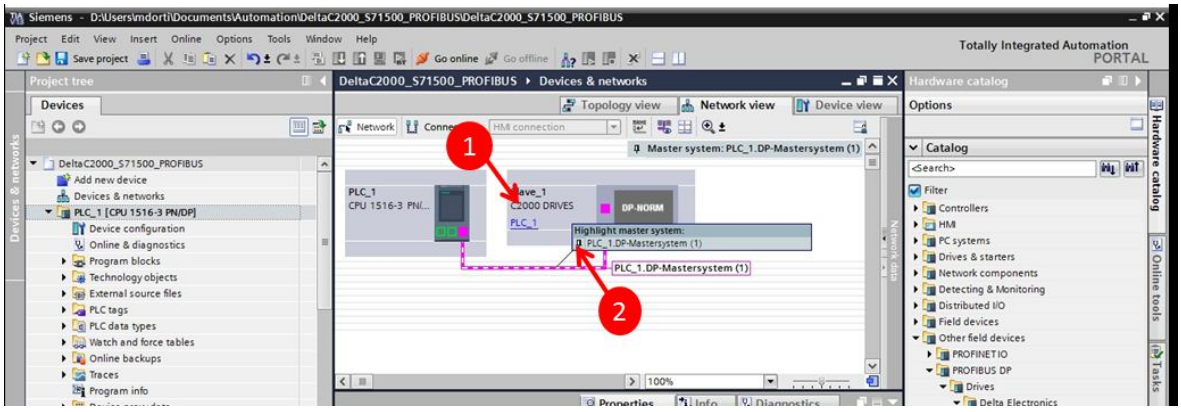


Figure 31 – Profibus Slave and Master system

Now the connection has already been stated and declared on the network view window, it is time to define the content of the slave and memory addresses, so Click on 1 – **Device view**, 2 – Double Click on the **device C2000**, 3 - click on the **4 PKW, 4 PZD** data pack. 4 – Check the slots and memory address, those will be the input/output data channels.

Go to PLC, compile and compile hardware changes, verify if there are no errors. Again to PLC, download, download hardware and software. The connection screen will appear, you already know what to do. Define your Ethernet Card and find your PLC on the **Start Search**. Click again on PLC and **download** it to the CPU, **Load**, **Finish**. Now the leds in C2000 should be all green, indicating communication.

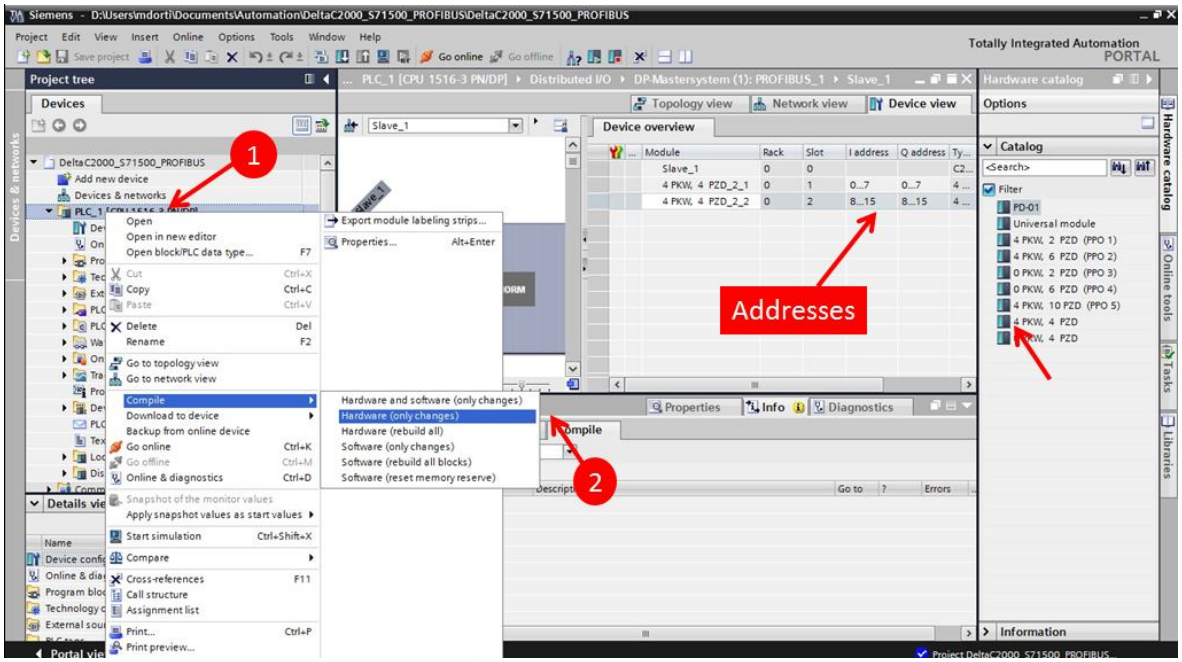


Figure 32 - Compiling a Program into PLC with hardware configuration set.

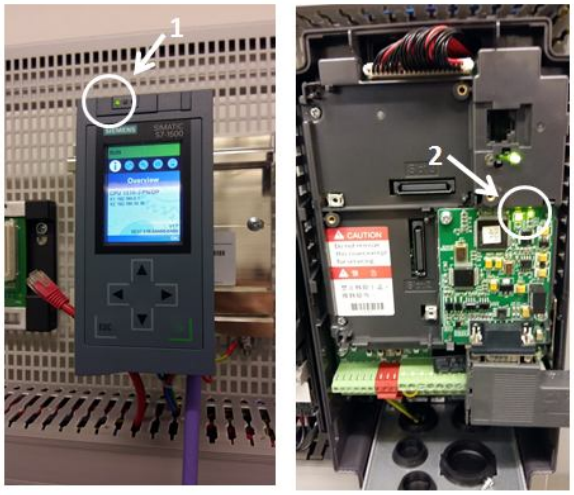


Figure 33 – 1 – Master, 2 - Slave cyclic communication established

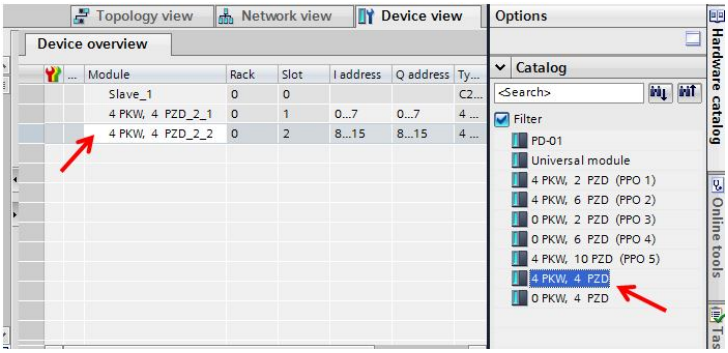
7 Transmitting and Receiving data through PZD

Siemens Profibus uses 2 ways of transmitting and receiving data:

PZD: – a way of transmitting data in cyclic period, this information is cyclic and is always present in the communication, ex. On/Off command, setpoint, frequency monitoring.

PKW: – a way of transmitting and receiving acyclic information, for ex. Change parameters and read parameter data, this data is requested and then read or written. When we selected the content of the Profibus bus slave on the previous chapter we selected, 4 PKW, 4 PZD, which means: Send and receive 4 PKW words, send and receive 4 PZD words.

1 word means 2 bytes, 16 bits of information. Siemens S7 PLC family is byte oriented, which means every number you see in STEP 7 controllers is a byte. 1 word = 2 bytes. On the example, we have I address and Q addresses. So, from byte 8 to 15 I, master receives data from the slave, and from 8 to 15 Q, master sends data to the slave, input and output on the master's view on PZD data. PKW we will see further, which is on the first slot.



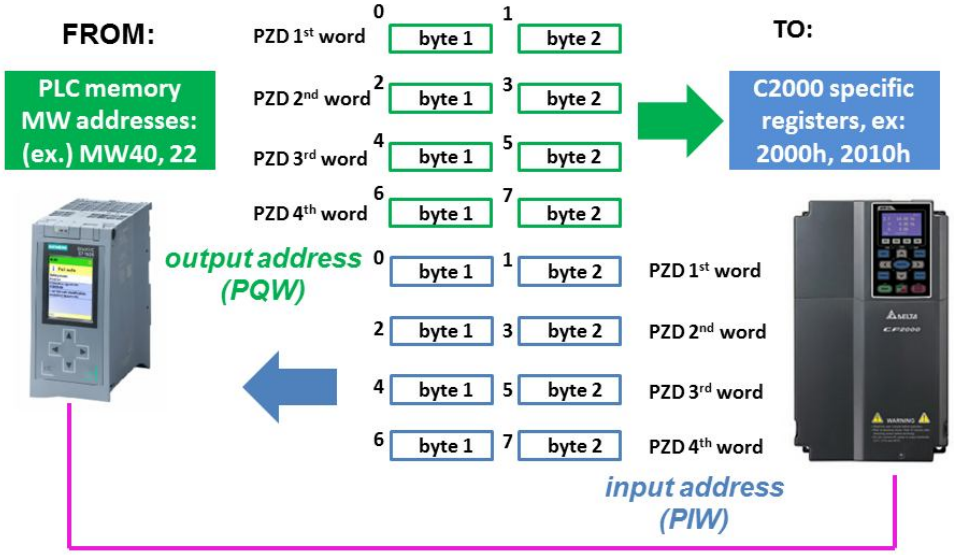


Figure 34 - Profibus Master Slave communication

This was a basic introduction on how data is transmitted on Profibus, we will do a practical example, following the picture, so 1 – Go to **Network view**, double click on C2000, you will be taken to the 2 - **Device view screen**, now go to Device-specific parameters. 3 – Data output parameters (CMC-PD01 -> VFD), that means, data from the PLC, to Delta C2000 drive. 4 – Insert those values (they will be explained on the next pages). For better visualization values, go to the next table, where you will find the information of the addresses. Scroll down on Fig. 37 – Continue on **Device specific parameters**, now it is time to set (VFD-CMC-PD01) from DeltaC200 to PLC and insert values listed.

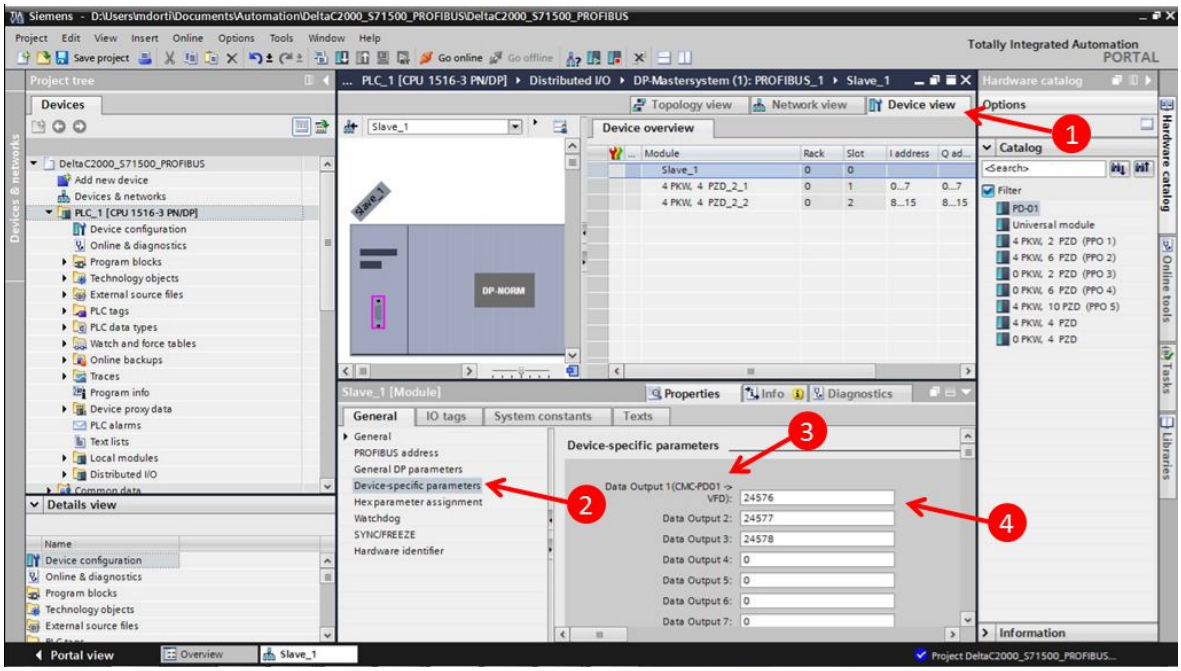


Figure 35 - Setting Data input parameters

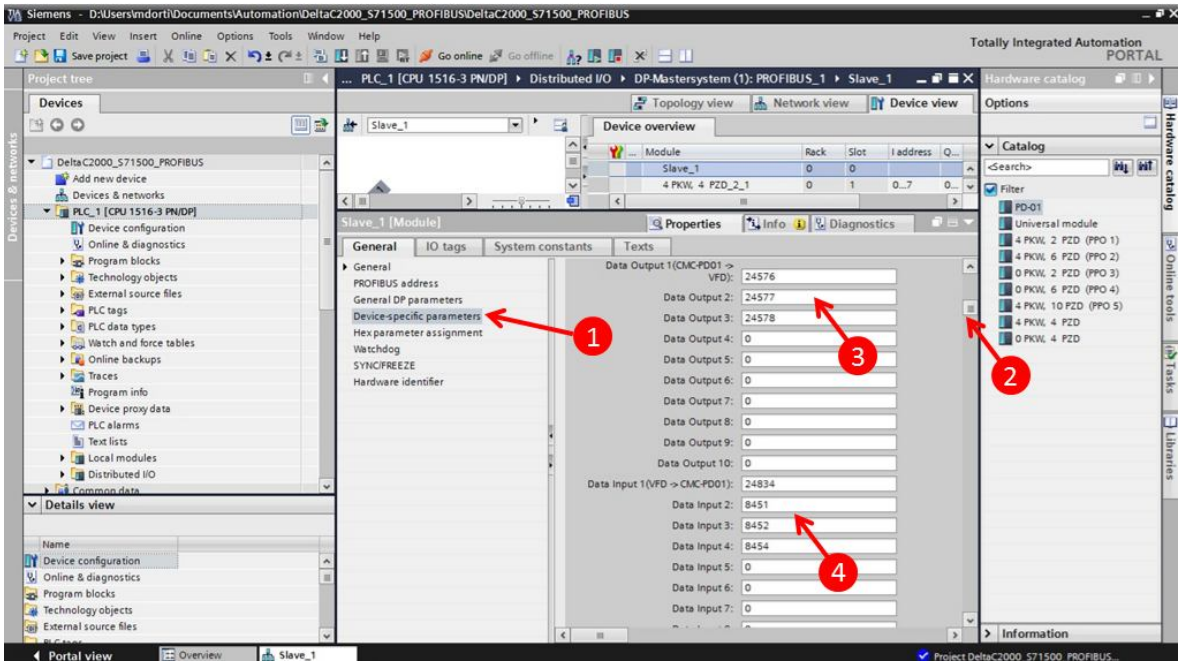


Figure 36 - Setting data output parameters

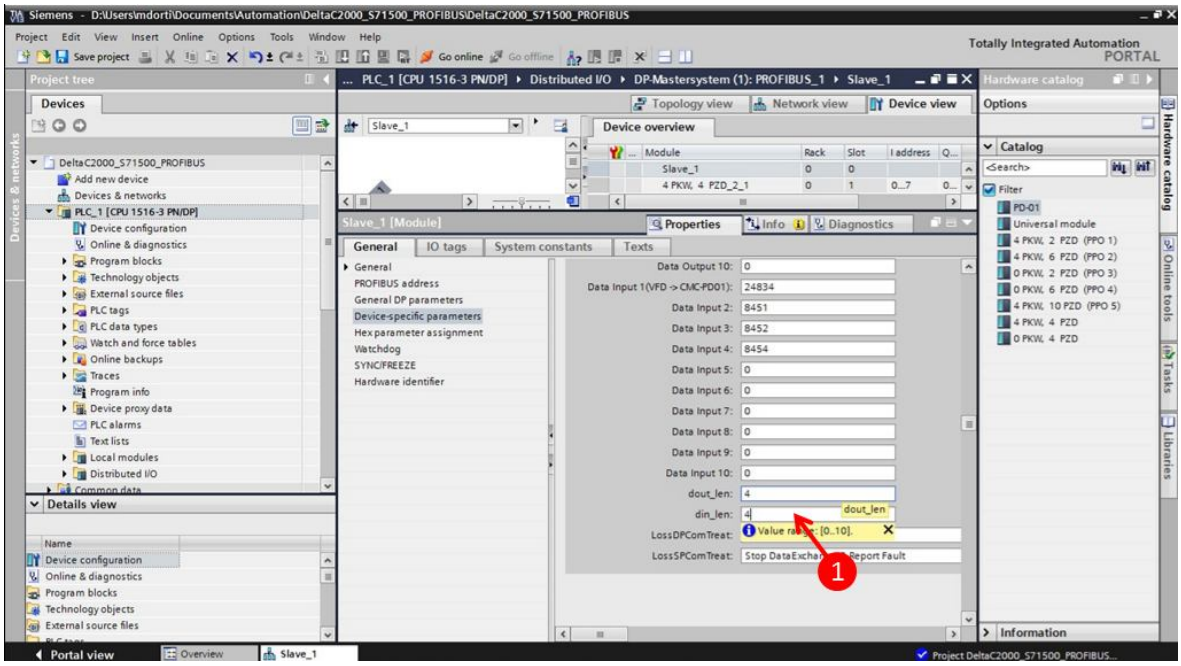


Figure 37 - Setting data length.

On the final part of it, scroll down properties and in **dout_len** and **din_len**, set 4. Which is the number of words which are going to be transmitted.

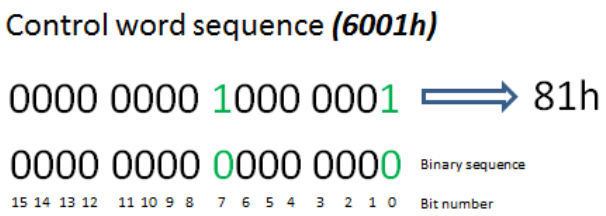
Set **P 09-30 = 1**, which means new standard for decoding **60XX**, this is the best decoding because with it you can send many other types of data to the converter.

CMC-PD01 -> VFD	Value(dec)	Value(hex) P0931 = 60XX
Data Output 1	24576	6000 **(control word)**
Data Output 2	24577	6001 **(control mode)**
Data Output 3	24578	6002 **(setpoint)**
Data Output 4	0	0

VFD -> CMC-PD01	Value(dec)	Value(hex) P0931 = 60XX
Data Input 1	24836	6104
Data Input 2	8451	6103
Data Input 3	8452	6102
Data Input 4	8453	6103

Table 1- C2000 addressing for data exchange with PROFIBUS

If you go the manual of Delta C2000, to page 408 of .pdf manual, you will find a table called address list. There you can find all information which can be sent to **Delta C2000** and all the information which can be received through those addresses. The list is in **hexadecimal** but the address must be entered in **decimal** in the **Device Specific Parameters**, when using P0931=60XX, replace the first number “2” by 6” and **convert it to decimal** with Windows calculator. When doing this with the **input data**, do this **ONLY** for the **first word**, like the example above. Here follows a schematics of how words are sent into the registers:



bit	Function
0	Enable setpoint
1	Direction (0 = FWD, 1 = REVERSE)
2	External command 1 (Only for positioning)
3	HALT (0 = runs to target speed / 1 = runs according to acceleration)
4	LOCK (0 = Continue to target speed, 1 = Stop in actual frequency)
5	JOG (0 = JOG OFF/ 1 = JOG ON)
6	Quick Stop (0 = No quick stop, 1 = quick stop)
7	Servo ON/OFF (0 = OFF/ 1 = ON)
8	External Command 2 (Only for positioning)
9	Clear absolute position (Edge 0 ->) (Only for positioning)
10	RESERVED
11	RESERVED
12 -13	13 – (Decimal command: 00 = decimal 2, 01 = Decimal 10 = Decimal 1, 11 = Decimal 0)
14	RESERVED
15	Edge->0 = fault acknowledgment

Figure 38- Important address on C2000, 6001h forms a control word

Which means, if you send, bits in the following sequence, the drive will respond to its functions when there is “1” , example: 0000 0000 **1**000 000**1** = 81h – ***setpoint enable*** and ***SERVO ON***, so the drive will **RUN**. Another important word on the Profibus communication is the status word, it tells the PLC basic status of drive from an address.

bit	Function in status word
0	Setpoint present (0 = setpoint not present / 1 = setpoint present)
1	Direction (0 = FWD / 1 = REV)
2	Warning (0 = No warnings / 1 = warning active)
3	Error (0 = No faults / 1 = fault active)
4	
5	JOG (0 = No jog / 1 = Jog mode on)
6	Quick stop (0 = No quick stop / 1 = quick stop ON)
7	Servo ON/OFF (0 = Drive stopped, 1 = Drive ON)
8	Position Match (0 = Not referenced cam / 1 = Referenced cam)
9	RESERVED
10	RESERVED
11	RESERVED
12	RESERVED
14	RESERVED
15	RESERVED

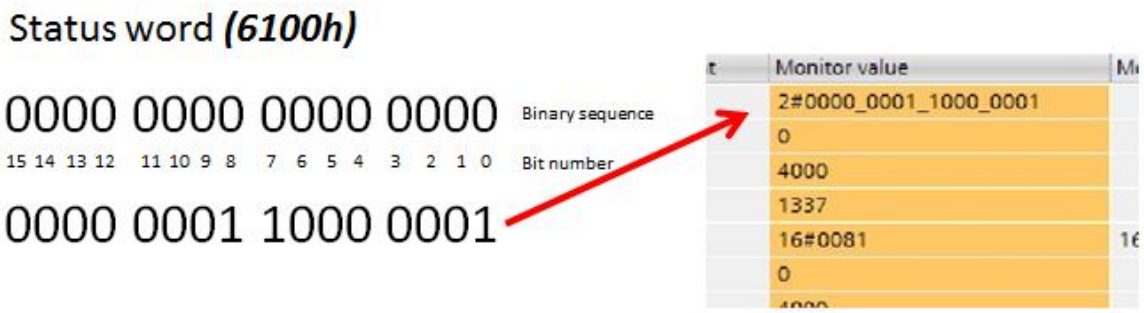


Figure 39 - Important Address 6100H is the Status of the drive

Control word and status word of the drive, are the 2 basic words which complete communication, the status word not always must be received, depending on the address set on **"Data parameter specific parameters". Our example does not contain the status word, however, you can add it, get its bits and insert in your program.**

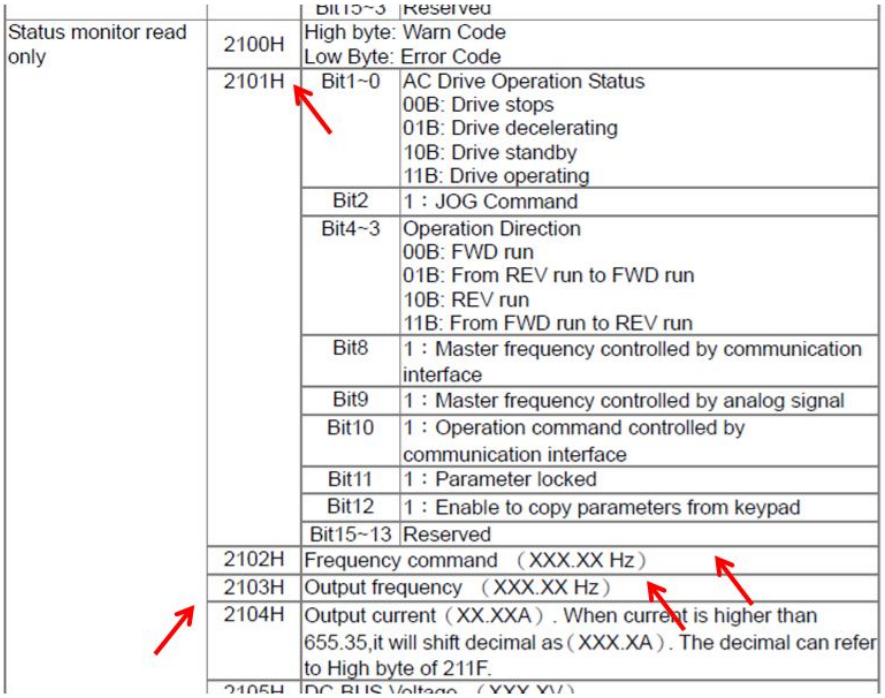
The second word to set is the drive **control mode**, if you set the address **6001h**, and send **0,** it will be set velocity mode, **6001h = 0**, **velocity mode**, the other numbers are **torque mode = 2**, **position = 3 mode** and **homing mode = 1**, which we are not using in this example.

The third word to set is the ***setpoint*** and that goes on address **6002h**, which was shown in the example which means, if you send a value, of **5000** to it, it will make the drive go to a frequency of **50Hz**. The value must be inserted in decimal **x 10^{-2}** so it can be controlled correctly. So, now we have a **control word** address and also a **frequency command** address. Compare these values of the manual with the ones we inserted on the **Data specific parameters** on the previous page.

Table 2 - Addresses and functions for command with Delta C2000

Data specificic parameter (Decimal)	Address on 60XX coding (hexadecimal)	Function
24576	6000	Control word
24577	6001	Control mode
24578	6002	Speed command
24579	6003	Torque limit
24580	6004	Position command
24581	6005	Position command
24582	6006	Torque command
24583	6007	Speed limit

The way back, from the VFD to the drive works the same way: **2104**, **2103**, **2102** and **2101** represent: **Output current**, **Output frequency**, **Frequency command**.



		Bit15~3	Reserved
Status monitor read only	2100H	High byte: Warn Code Low Byte: Error Code	
	2101H	Bit1~0	AC Drive Operation Status 00B: Drive stops 01B: Drive decelerating 10B: Drive standby 11B: Drive operating
		Bit2	1：JOG Command
		Bit4~3	Operation Direction 00B: FWD run 01B: From REV run to FWD run 10B: REV run 11B: From FWD run to REV run
		Bit8	1：Master frequency controlled by communication interface
		Bit9	1：Master frequency controlled by analog signal
		Bit10	1：Operation command controlled by communication interface
		Bit11	1：Parameter locked
		Bit12	1：Enable to copy parameters from keypad
		Bit15~13	Reserved
	2102H	Frequency command （XXX.XX Hz）	
	2103H	Output frequency （XXX.XX Hz）	
	2104H	Output current（XX.XXA）. When current is higher than 655.35,it will shift decimal as（XXX.XA）. The decimal can refer to High byte of 211F.	
	2105H	DC BUS Voltage （XXX.X V）	

Figure 40 - General addresses of the list, which can be read

Once you understood this **address table**, there are no limits for data exchange between **S7 1500** and **Delta C2000**, you can now check any actual value from the drive and process it into your Main Program routine. The addresses on this list work following the example, the **first address** must be converted to 60XX, but the ones following do not need. An example is shown on the table below.

Table 3 - How reading address should be set

Data specific parameter (Decimal)	Address on 20XX coding (hexadecimal) - Address list	Address on 60XX coding How you should insert in Data specific Parameters	Function
8450	2102	**24834**	Frequency Command (Hz)
8451	2103	8450	Output frequency (Hz)
8452	2104	8451	Current (A)
8453	2105	8452	Torque limit
8454	2106	8453	DC link voltage

8 Programming the data transfer in OB1 (Main routine)

We already have the connection working, the hardware topology declared and functional and we just understood about the address and data exchanged, now it is time to put it into a program so we can create automation application with S7 master and all the functionalities of Delta C2000.

S7 CPUs have a main routine called OB1, and it is there that tha magic of the program runs, let's open the Main Routine through TIA Portal.

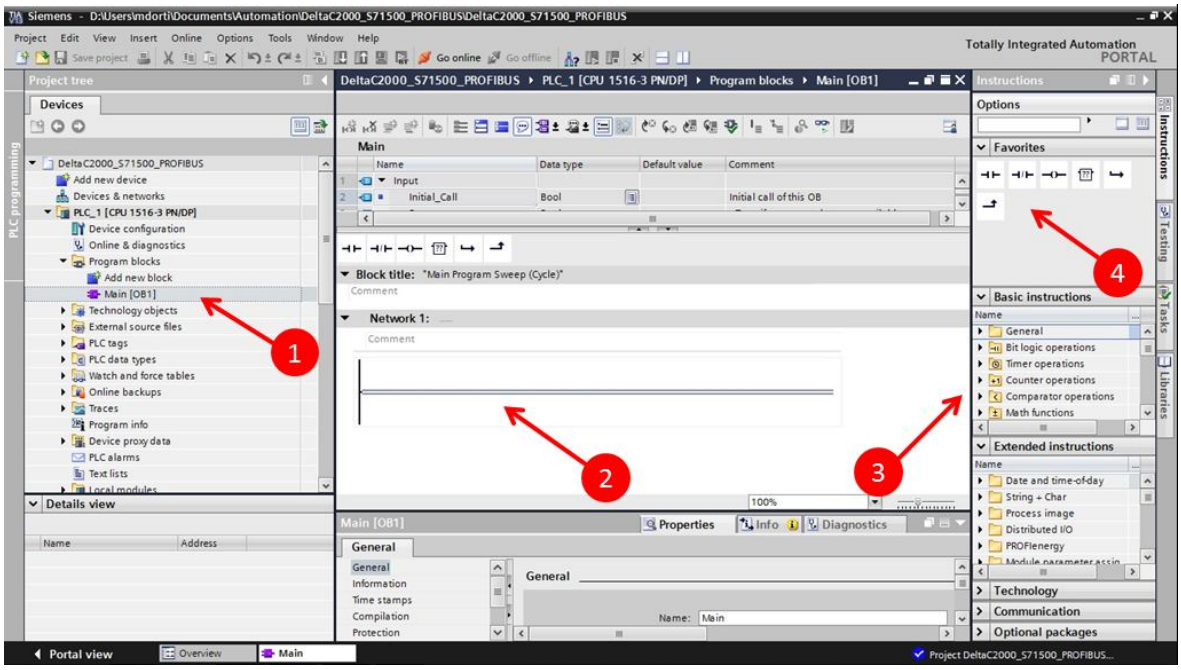


Figure 41: 1 - MAIN OB1 routine, 2 - Networks of Ladder diagram, 3 - General Blocks, 4 – Functions

In this practical example, we are going to use MW (memory words) of the PLC in order to write bits and then send them to C2000. We are going to use **MW20, 22, 24, 26** to send data, and **MW 40, 42, 44 and 46** to receive data from **C2000**, we will use **MW50** and **MW52** for error report of the transmission.

Table 4 - Data transmission example of practical example

Send Data	**1 Word**	**2 Word**	**3 Word**	**4 Word**
Address	MW20	MW22	MW24	MW26
Receive Data	**1 Word**	**2 Word**	**3 Word**	**4 Word**
Address	MW40	MW42	MW44	MW46

MW20 carries 2 bytes, high byte and low byte of the 1st word, and the same thing with the other words. Now let's create a tag table in order to make it easier for us to identify what we are sending and what we are receiving.

Create a Tag table, go to 1 - **PLC tags**, 2 - **Add new tag table**.

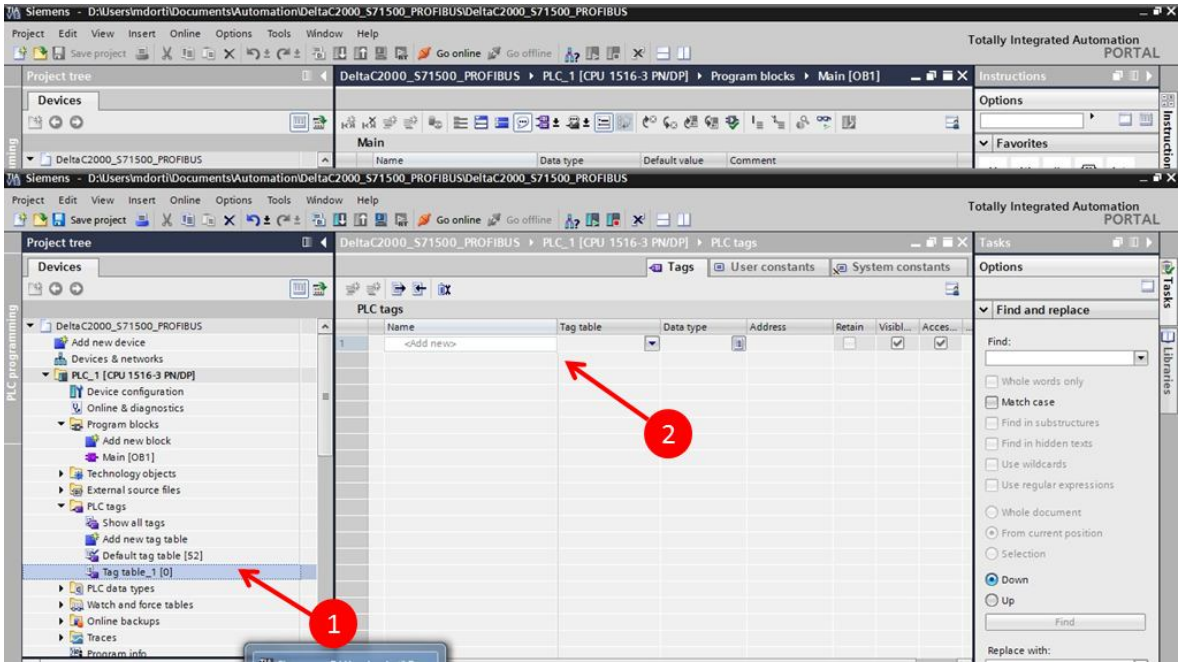


Figure 42 - Inserting a Tag Table

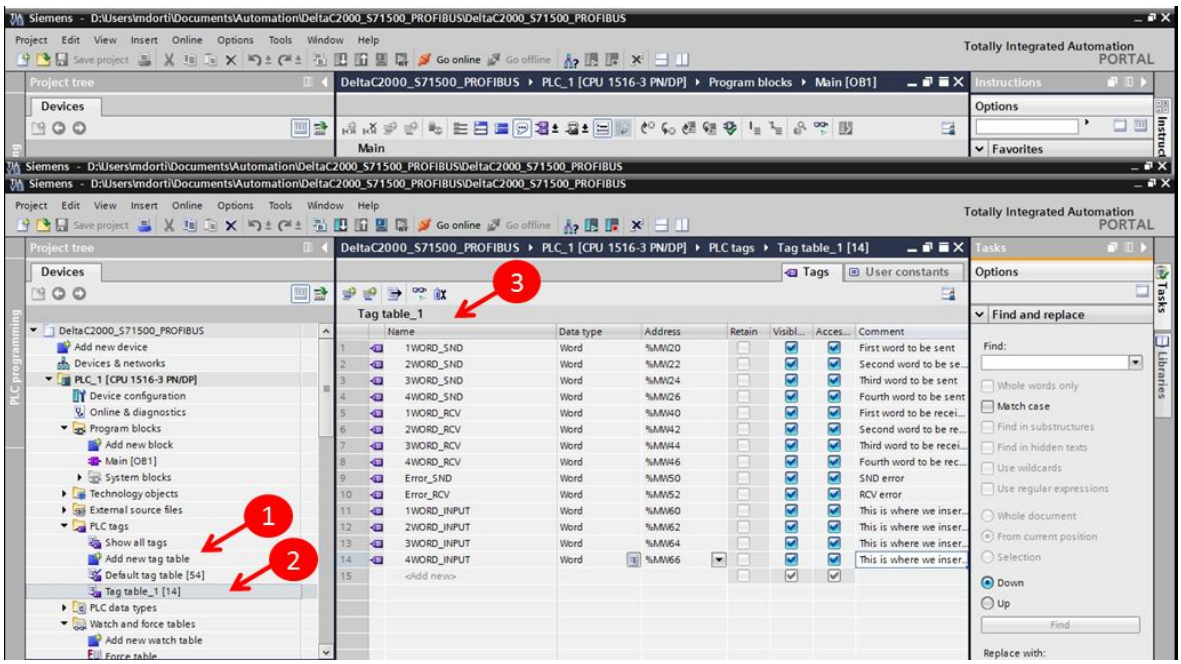
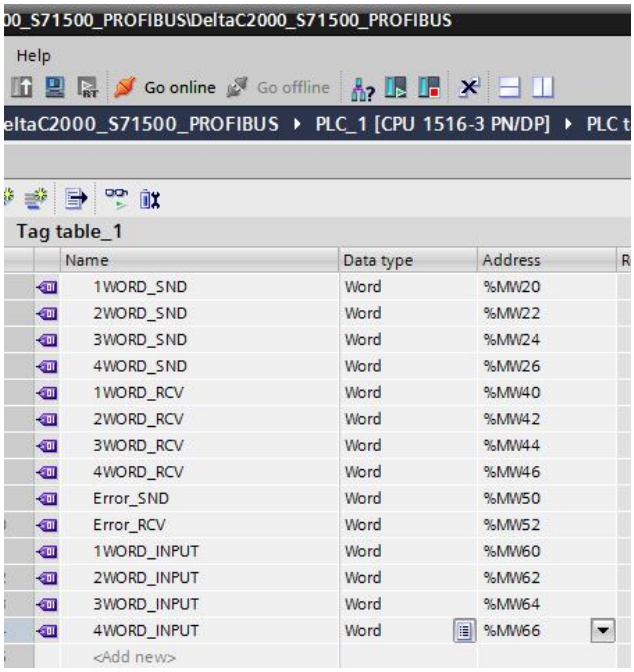


Figure 43 - Tag table with address and variable

Developing a **Tag_table** is not mandatory, but it will help you to understand what you are transmitting and receiving. So 1 – **Go to Add new tag table**, 2 – **Click on the new tag_table**, 3 – declare the Tags like the example listed, this can be found in the project attached.



Name	Data type	Address
1WORD_SND	Word	%MW20
2WORD_SND	Word	%MW22
3WORD_SND	Word	%MW24
4WORD_SND	Word	%MW26
1WORD_RCV	Word	%MW40
2WORD_RCV	Word	%MW42
3WORD_RCV	Word	%MW44
4WORD_RCV	Word	%MW46
Error_SND	Word	%MW50
Error_RCV	Word	%MW52
1WORD_INPUT	Word	%MW60
2WORD_INPUT	Word	%MW62
3WORD_INPUT	Word	%MW64
4WORD_INPUT	Word	%MW66
<Add new>		

Figure 44 - Tag list of the project

This is the tag table, **4 Words** to send, **4 Words** to receive, 2 **Errors words**, **4 input words**, since you have to move memory values on main routine in order to be transferred with consistency.

Now, it is time to program the S7 CPU in order to transmit/receive data, we are going to use two very usual transmission function blocks of S7 family SFC14/SFC15 – DPWR_DAT and DPRD_DAT.

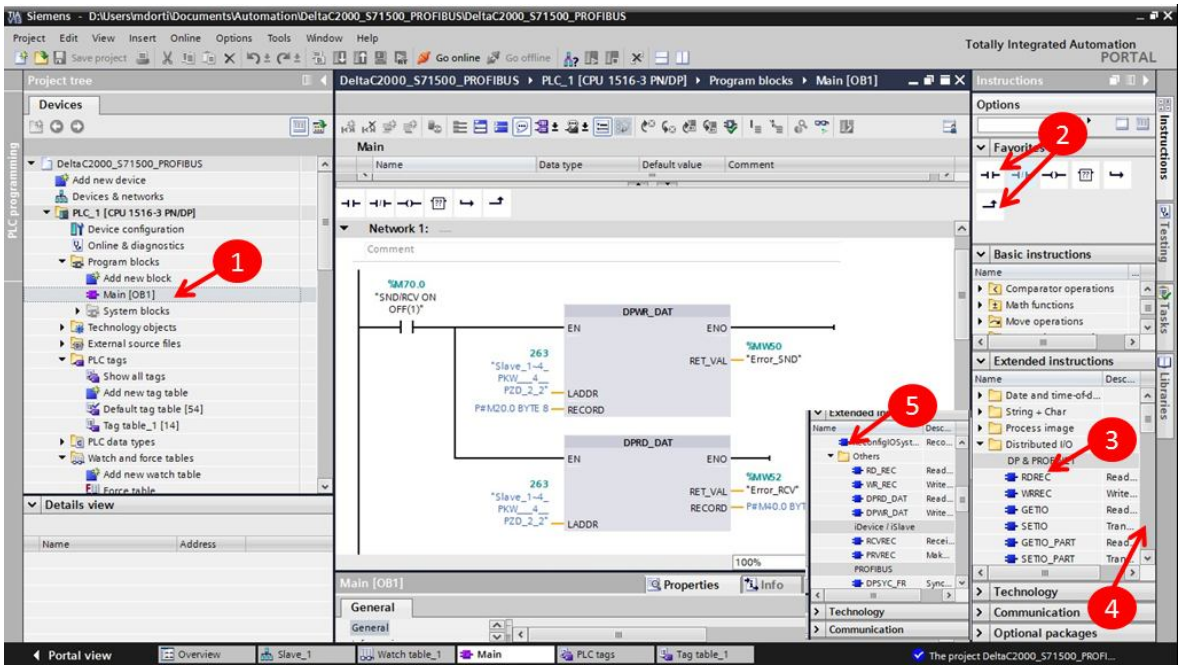


Figure 45 - Main and Extended Instructions

1 – Click on **Main**, then 2 – Click on the Normally open contact to insert it in the network (conventional ladder) and then another branch. 3 – Go to to **Extentend Instructions**, **Distributed IO**, then 4 – Scroll down to **Others** and then **DPWR_DAT** and **DPRD_DAT** blocks will be available.

It is necessary to use those 2 blocks to write and read data in consistent way from Delta C2000 consistent way just means, data will be transmitted all at once, in cycles of number of bytes. Basically, these two blocks require the same information.

1 – Contact to start communication,

2 – You can type the address, or select it from the list,

3 – LADDR, which means, hardware address, select the Slave1-4, PZD2_2 n. 263 in order to select the address of the slave, in this example 263 (a list will show you the available slave addresses).

4 – Pointer for data transmission, copy this format on the block: **P#M20.0 BYTE** 8, which means, send from MW20, 8 bytes.

5 – Error address for RCV block, we set those 2 addresses in the tag list we made.

6 – LADDR, hardware address again, of the slave the same for DPWR_DAT / DPRED_DAT.

7 – Now, another point, copy this information on block **P#M40.0 Byte 8**, which means, put the values read from the slave and insert them from MW40 on the next 8 Bytes.

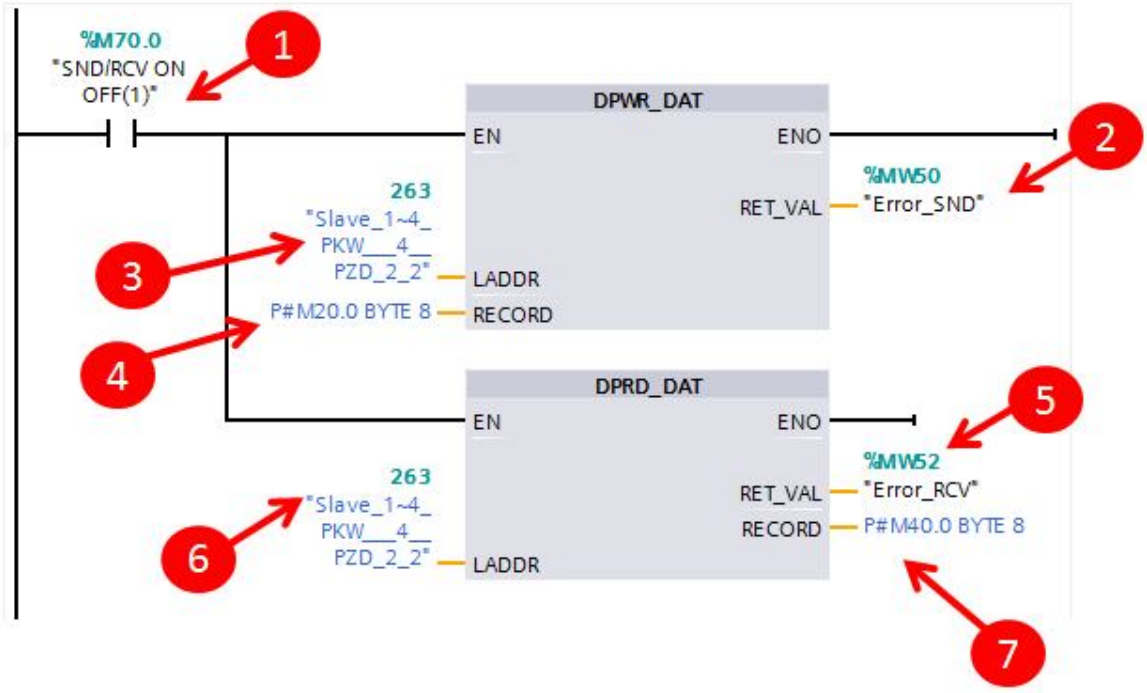


Figure 46 - First Network of OB1 routine.

Consistent communication unfortunately as far as tested in S7 1500 needs to be moved to the address, and cannot be written direct on the Process image like PIW, PQW. So, we still need to MOVE values from one memory address to another memory address which will be sent, that is why we created MW20 – **1Word_SND** and **MW60** – **1Word_INPUT**, we are going to input the words, move them to another memory range, which the blocks will scan, and send to the Delta C2000. The next pictures show how your following networks should be, MW60 value will be moved to MW20, MW 62 will be moved to MW22. Following that, MW64 will be moved to MW24, MW66 will be moved to MW 26.

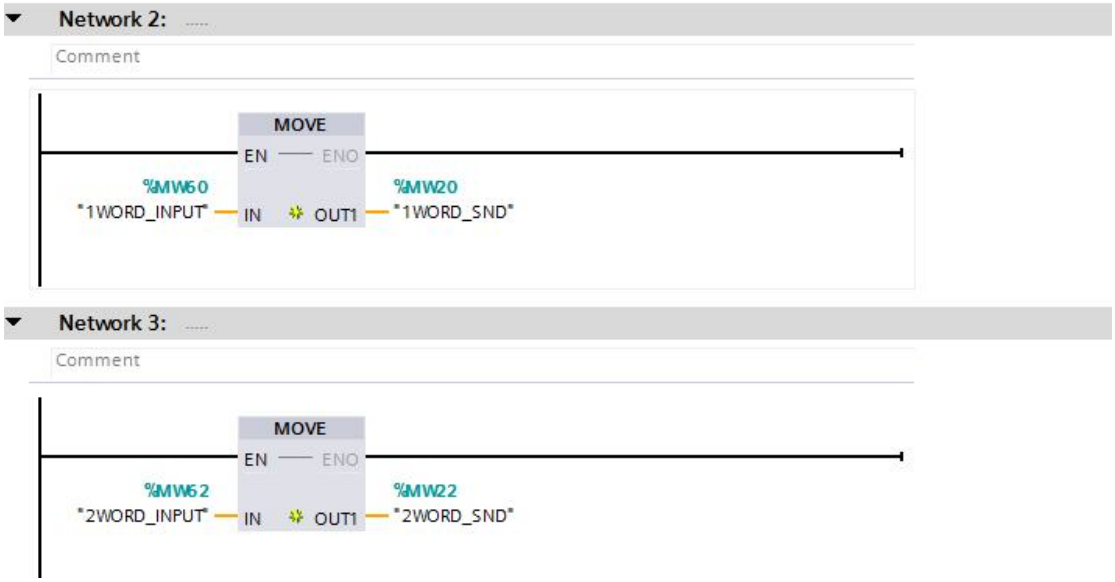


Figure 47 - Network 1 and Network 2

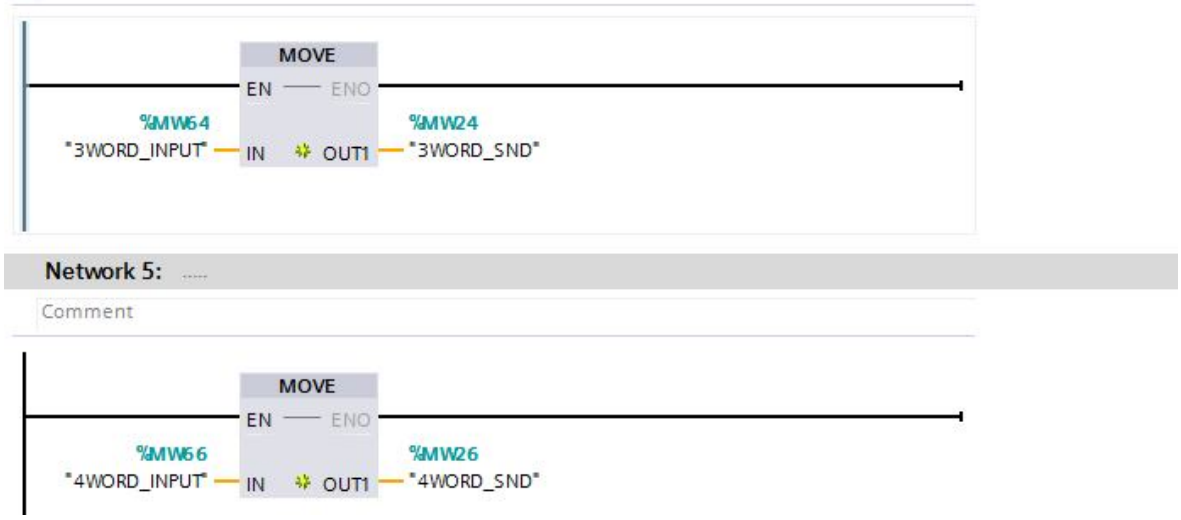


Figure 48 - Network 4 and Network 5

9 Testing communication between S7 1500 and C2000

Once you finished the programming of the chapter before, compile the project, go to the PLC, left-click and then **compile**, **complete hardware and software (only changes),** and then, again, PLC, left-click and **Download, Hardware and software (only changes)** to **S7 CPU**.

If everything was done correctly, you should have no errors on the compilation of the 5 networks we programmed on the previous page. Now it is time to test our communication, sending data to the converter C2000 and receiving it in the PLC. We are going to use a Watch Table, which is similar to the old VAT from S7 Classic, Go to **Watch and Force tables**, **Add new watch table**. Before that, now with the program already downloaded, you can go to the **ONLINE** option in order to monitor what is happening during the Main Routine and on the memory content. 1 – Click on **GO ONLINE**, 2 – Check the Orange bar, if it is orange, it is online. Check the green circles and tick boxes in green, it means they are working properly.

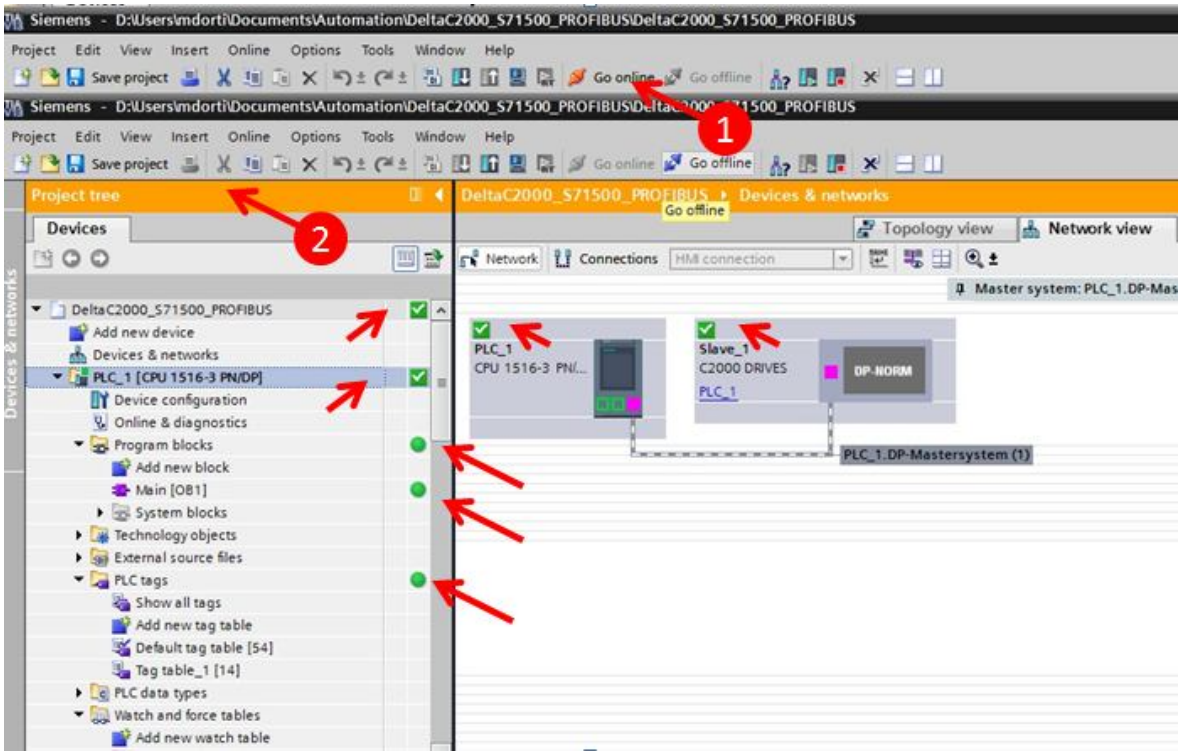


Figure 49 - Online setting of the communication

Now we can create our Watch table, 1 – **Add new Watch Table**, 2 – **Watch_Table_1**, 3 – This is where you modify the values on the memory address, 4 – Here you monitor the values of the memory addresses, 6 – Here you specify the value you want it to be, 7 – **81 hex**, will enable the drive on/ **80 hex** will disable it, 8 – This is where you specify the frequency command, the **second word** = **Control mode**, it must be set to 0 for **speed mode**.

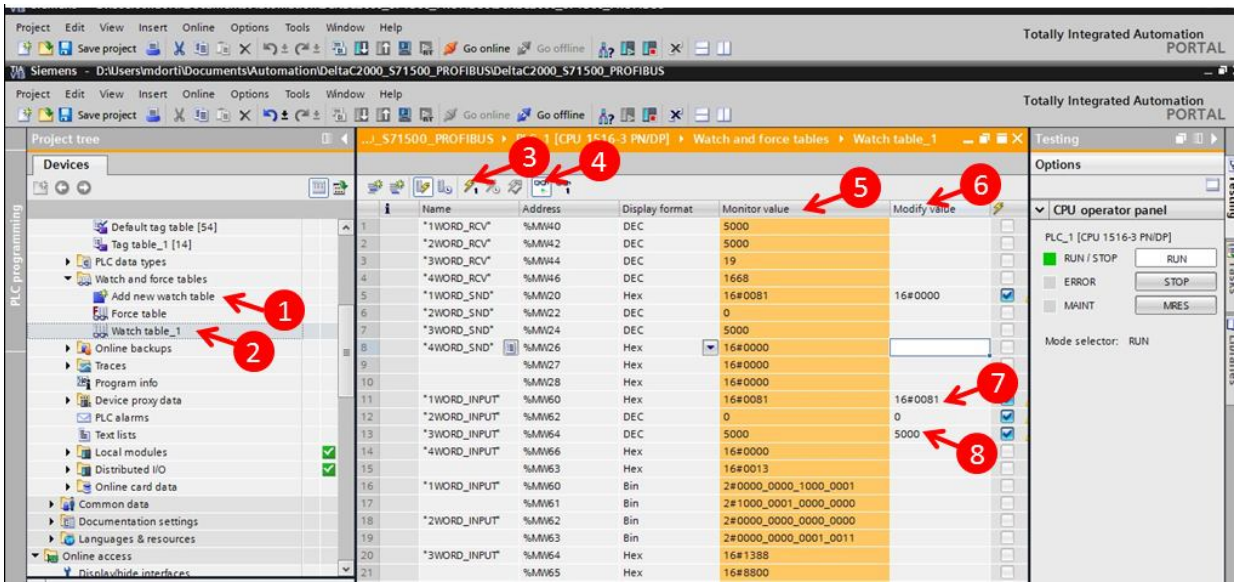


Figure 50 - Usual Watch table for basic drive connection

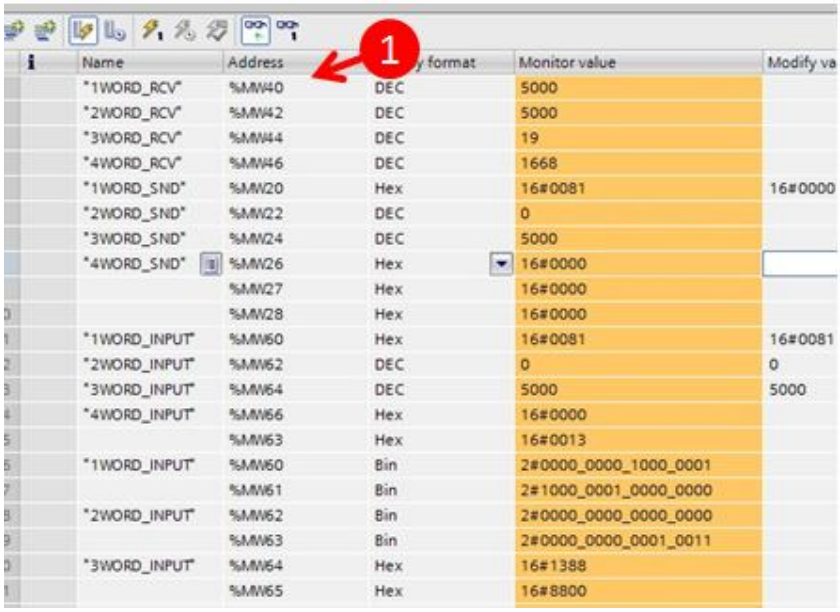


Figure 51 - Basic Watch Table for communication

1 – Here you should type those **address memories** in order to monitor them, everytime you put a MW address declared on the **PLC TAG LIST**, it will automatically fetch its **TAG**.

The orange part will just be seen when you start the communication process on the **Main Routine**.

Now, get back to the main routine, OB1, and monitor the blocks, left-click on that normally open contact we named **SND/RCV ONOFF(1)** set it to **ON**. 1 – Click to monitor, 2 – Left click and then, set to 1. When you do this if you had set the values on the **Watch Table**, drive will start running.

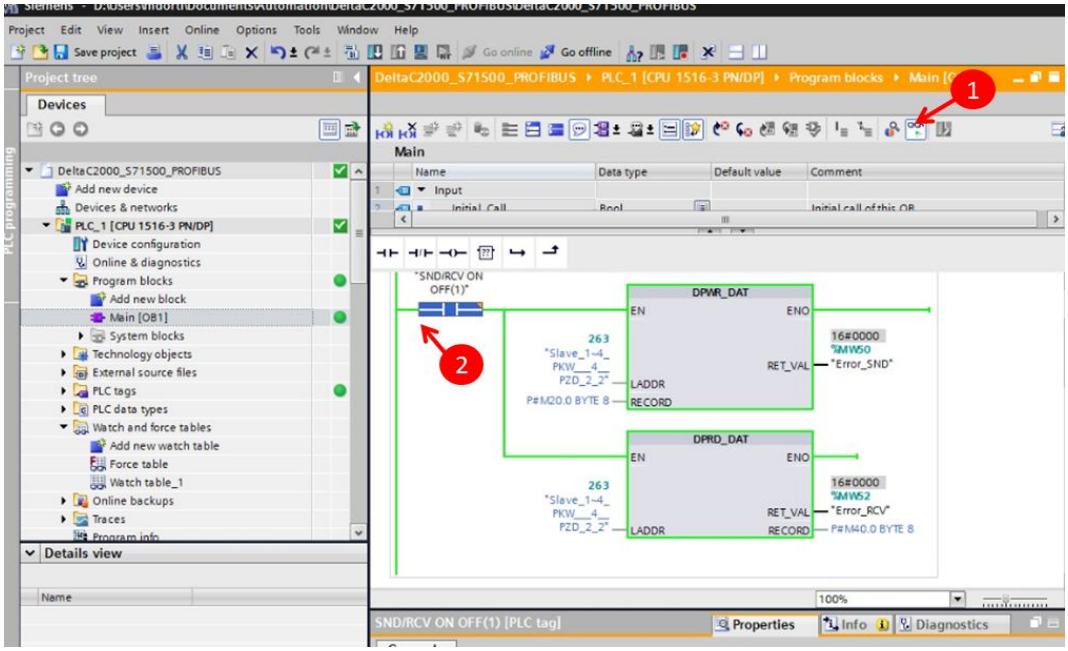


Figure 52 - Enabling communication

Now, you can also change the data to be transmitted on the Watch Table, when you do it, left-click on the line and modify value, or go to the **thunder button** to modify the values instantly. Do like in the example first, we will check what is being done after, when you **modify** those values, drive should start running. We set the Output data on the converter on "**Data specific parameters**."

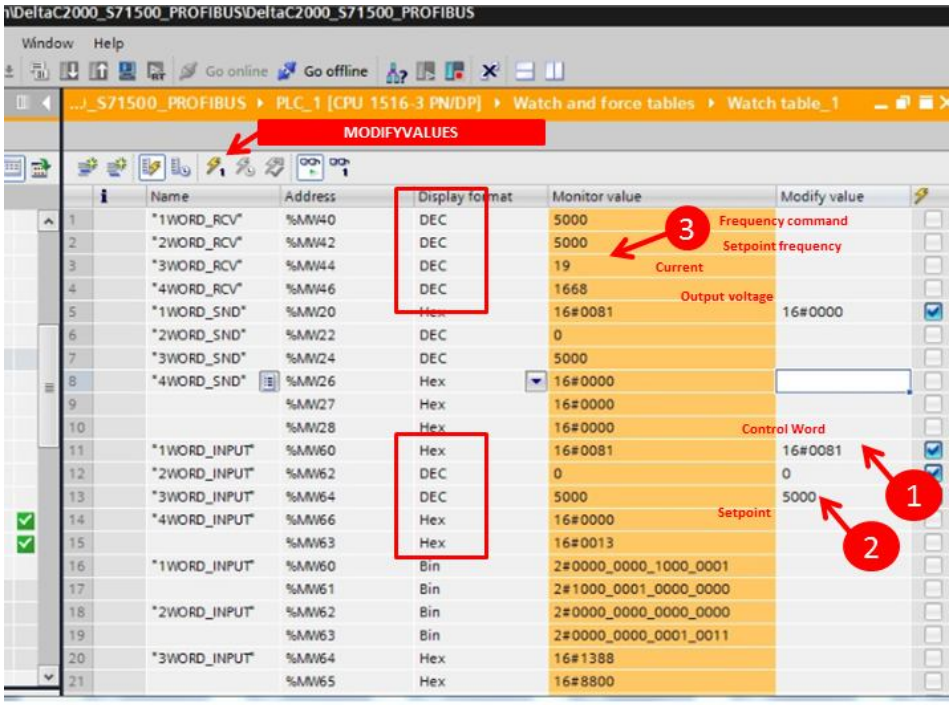


Figure 53 - Watch Table transmitting and receiving data

Figure 54 - Profibus Master sending and receiving, drive runs and current, frequency and voltage are displayed

We have done just a simple communication, sending a control word, control mode and a frequency command to the drive, we acquired data such as frequency command, current frequency, current and output voltage of Delta C2000. Take a look on the register table in **C2000** manual, the same we checked some pages ago, there you can find all information you can send and receive through **PZD data exchange** and **cyclic communication**.

Let's just analyse the data we sent and received, we sent 81hex to enable the drive and 80hex to disable the drive. 5000 on frequency command, we received as word in decimal 5000, 5000 19 and 1668.

81 is a the control word, if we convert it into binary in 16 bits we will have 0000 0000 1000 0001, taking a look at the table of coding 60XX (page 25) on the addresses list we will find the description of each bit in 60XX coding.

You can use a block **WORD to BOOL/BOLL to Word** to make your life easier in order to send data to the drive, and to receive it. You will also have to divide the values which in decimal from the drive by **100**. And to input values, multiply it by **100** sending them in decimal.

Communication with S7 1200 is really the same, you can follow the same procedures, and just change S7 1500 to S7 1200 on the specifications when inserting a CPU.