Non-Profit Joint Stock Company

ALMATY UNIVERSITY OF POWER ENGINEERING AND TELECOMMUNICATIOS NAMED AFTER GUMARBEK DAUKEEV

Electronics and robotics» department

INDUSTRIAL CONTROLLERS IN ROBOTICS PROCESSES

Laboratory work guide for students enrolled in the educational program 6B07110 – Control and measurement systems and instrumentations in robotics

Almaty 2022

AUTHORS: Kalkabekova T. Zh. Industrial controllers in robotics processes. Laboratory work guide for students enrolled in the educational program 6B07109 – Control and measurement systems and instrumentations in robotics. – Almaty: AUPET named after G.Daukeev, 2022. – 33p.

Provided guidelines include instructions, which are devoted to the study of technological process's automation in the robotics field.

Detailed, step-by-step, all instructions to perform the laboratory assignments given to students are reviewed and explained. Starting from creation of the simplest connection between controllers up to creation of a complete industrial network carrying out data exchange between controllers. Also, the topic of creating a web page to monitor controller's current values is touched upon.

In addition, rational methods of building and editing the industrial network are explained.

Guidelines are made to get necessary skills of work with the TIA Portal software and are intended for the students of the specialty 5B071600 – «Instrument Engineering». Two hours for each laboratory work are needed.

Ill.- 39, bibliography.- 6

Reviewer: Associate Professor of Electrical Engineering department
A. S. Baimaganov

Published according to the publication plan of non-profit joint stock company «Almaty University of Power Engineering and Telecommunications named after G. Daukeyev» in 2022.

Introduction

Controller is a software device necessary for control and management of various devices, processes, complexes. The controller can be called the basic element of automation.

There are different number of vendors, engaged in the production, maintenance of controllers, such as: Siemens, Schneider Electric, ABB. In this tutorial all the works are made with controller and TIA Portal software both are created by Siemens.

The purpose of these laboratory works is to develop the skills of designing industrial networks, mastering the basic properties of microprocessor systems, studying the structural and program construction of industrial controllers, consolidating the skills of programming controllers.

Laboratory work №1. Creating a PLC-PLC connection using a TCON block

The goal is to get knowledge about the blocks, allowing to organize connection between controllers, to create an industrial network between two S7-300 controllers

1.1 A brief of theory

The following commands control the data exchange in the program such as: TCON, TSEND and TRCV, TDISCON.

This laboratory work will allow you to work and be familiar with the TCON block, which establishes the connection.

Both communication partners execute the TCON command to create and establish a communication connection. You use parameters to specify the active and passive communication endpoint. Once the connection is created and established, it is automatically maintained and monitored by the CPU.

The TCON unit has the following inputs:

1) REQ – control parameter that triggers the connection's establishment specified in the ID. The task is performed once a rising edge is sent to its input;

2) ID – reference to the connection to be established with the remote partner or between the user program and the communication layer of the operating system. The ID must be identical to the corresponding parameter ID in the local connection description. The range of values is W#16#0001 to W#16#0FFF;

3) CONNECT – link description pointer.

The TCON unit has the following outputs:

1) DONE – parameter that informs whether the task is still in progress or has already been executed;

2) BUSY – output which informs whether the task is finished or not;

3) ERROR – parameter which reports the error status: whether or not there

was an error when processing the task;

4) STATUS – parameter which provides detailed information about the error type and describes the connection status in the form of a code.

1.2 Work order

1.2.1 Create a project in the TIA Portal software, add the following devices to the network: two CPU 1516-3PN/DP controllers. Then, in «Program Blocks» field add one TCON unit from the «Communication» section (→ «Open User Communication» → «Others») (figure 1.1).

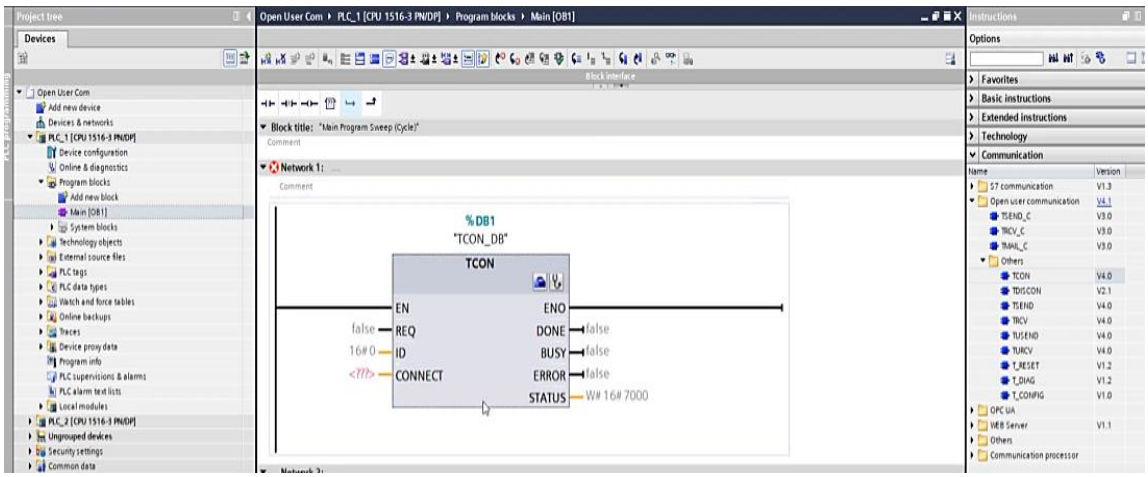


Figure 1.1 – Adding a TCON unit to the network

1.2.2 The added unit must first be configured. To do this, click on the «Start configuration» icon. Make the connection settings in the tab «Connection Parameter».

1.2.3 Select «Unspecified» type in the window named «Partner». Selected «Unspecified» mode means that the controller will be given the port number of his partner. «Broadcast» - means that the controller communicates with all devices in the network. «Multicast» - creating multiple connections with selective controllers. Later, in the tab «Connection Data» we have to create new data. And then, in addition, there is a need to write the IP address of the controller with which the connection is established (figure 1.2).

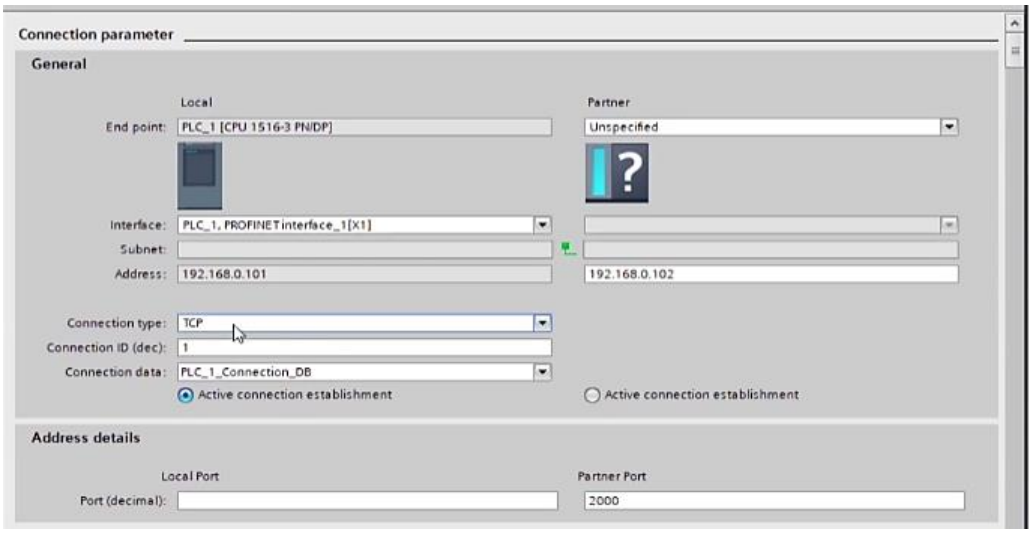


Figure 1.2 –1st controller's connection configuration

1.2.4 Next, we carry out a similar configuration of the 2nd controller, the only difference is that you need to designate the partner as passive (figure 1.3).

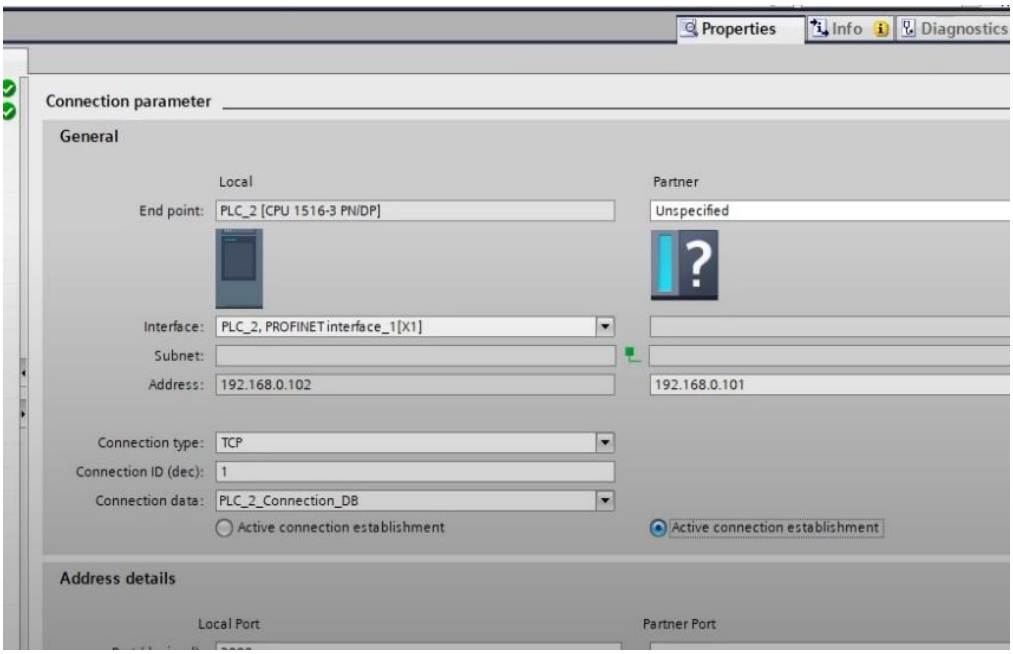


Figure 1.3 – 2nd controller's connection configuration

1.2.5 Following that, go to the configuration tab of the 1st controller and activate the «Block Parameter» tab. It remains to create a request - when the connection occurs. To do this, add a simple open contact with the address m0.0 and with the name «Connect» (figure 1.4).

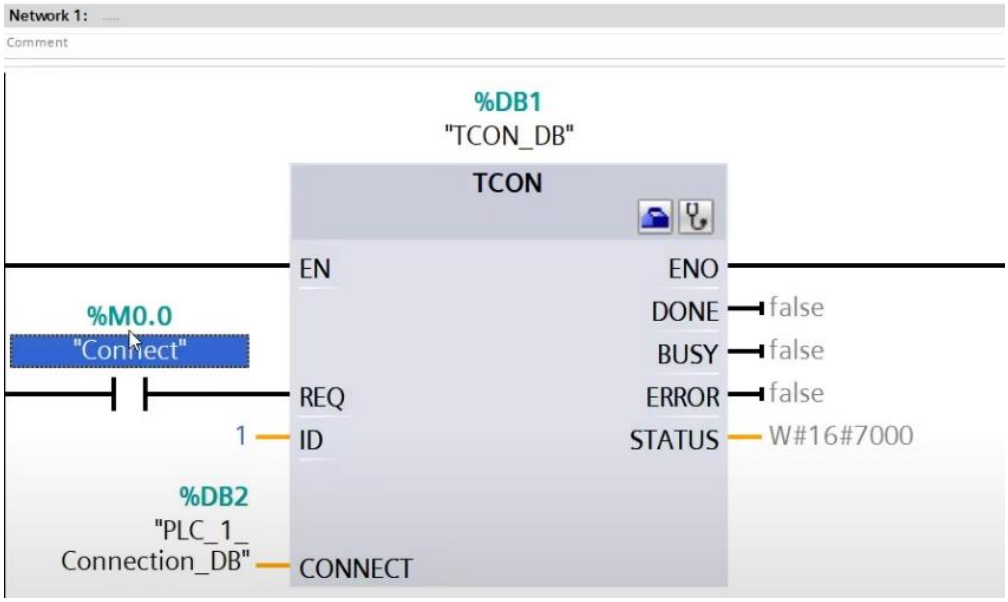


Figure 1.4 – Adding a button to launch the TCON block

1.2.6 After that, do the same for the 2nd controller, keeping the m0.0 tag. Now by clicking on the «Connect» contact the connection should be established.

1.2.7 Download the program to both PLCs. Go into simulation mode and see if anything changes if you change the normally open contact of the 1st controller (figure 1.5). If not, then why?

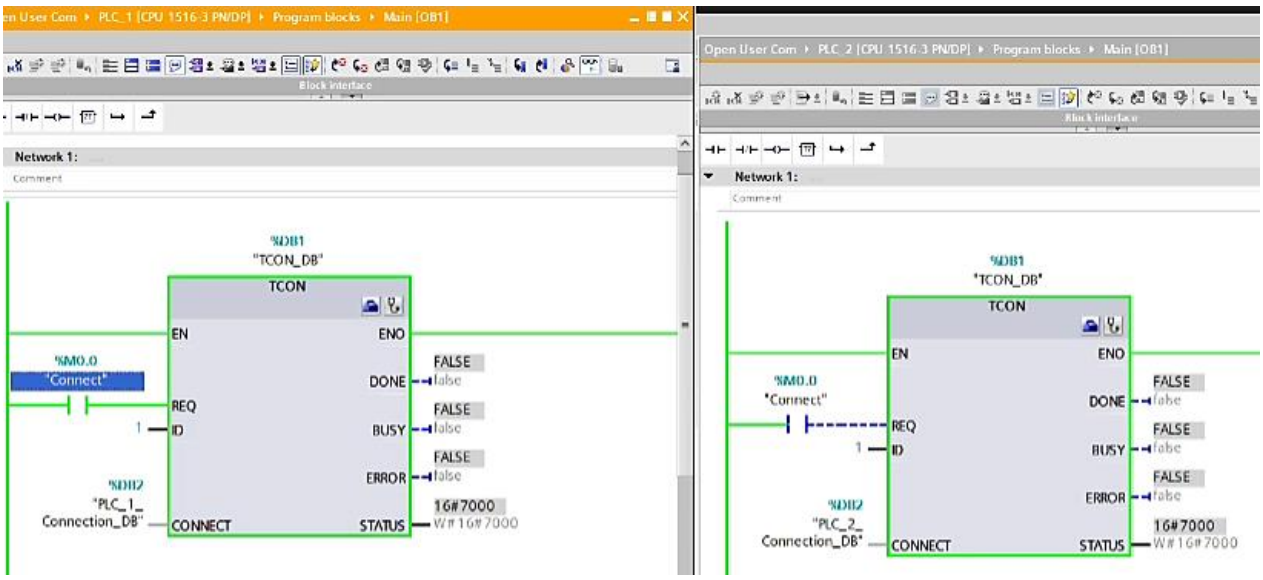


Figure 1.5 – Connection check between two controllers

1.2.8 The reason is that the TCON block on the 2nd controller must first be activated. You will see that the controller has changed its «BUSY» status from «FALSE» to «TRUE», and the status has also changed its number to 7002 - which

means that the controller is waiting for its partner. Then, activate the first controller. You will see that a connection has been established between the two controllers (figure 1.6).

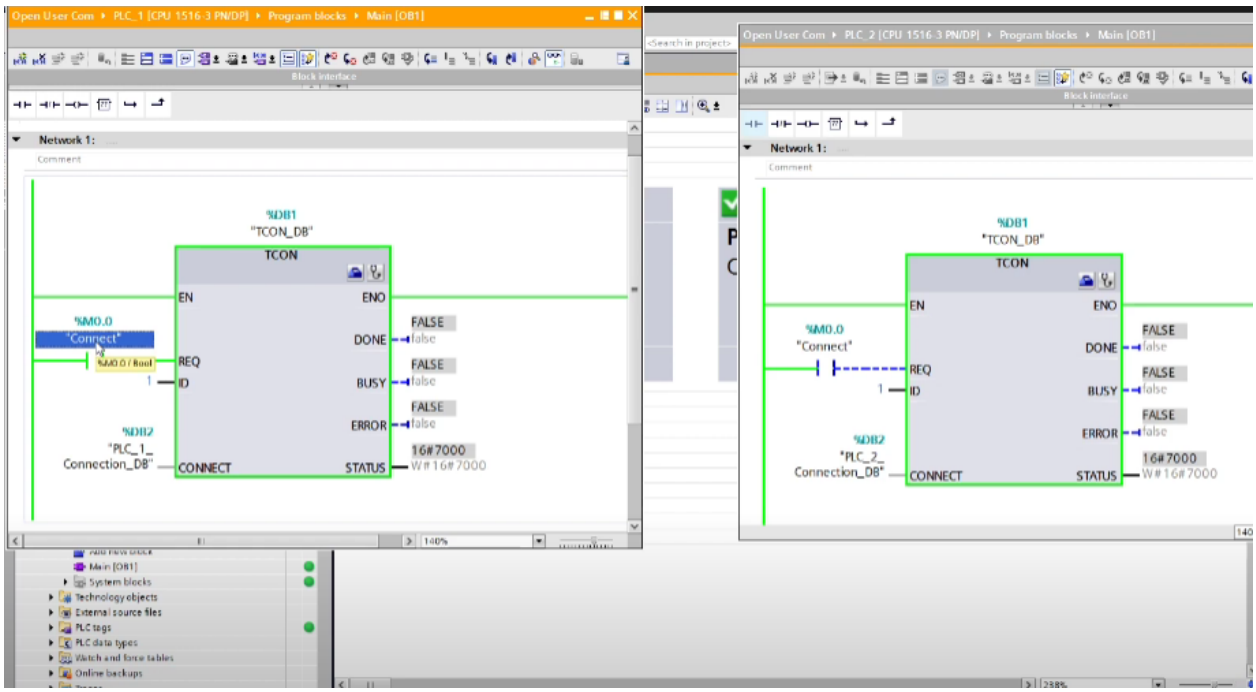


Figure 1.6 – Connection between controllers is set

1.2.9 In order to visualize how the connection goes, add an open contact, and a set contact for each controller. The connection can be observed (figure 1.7).

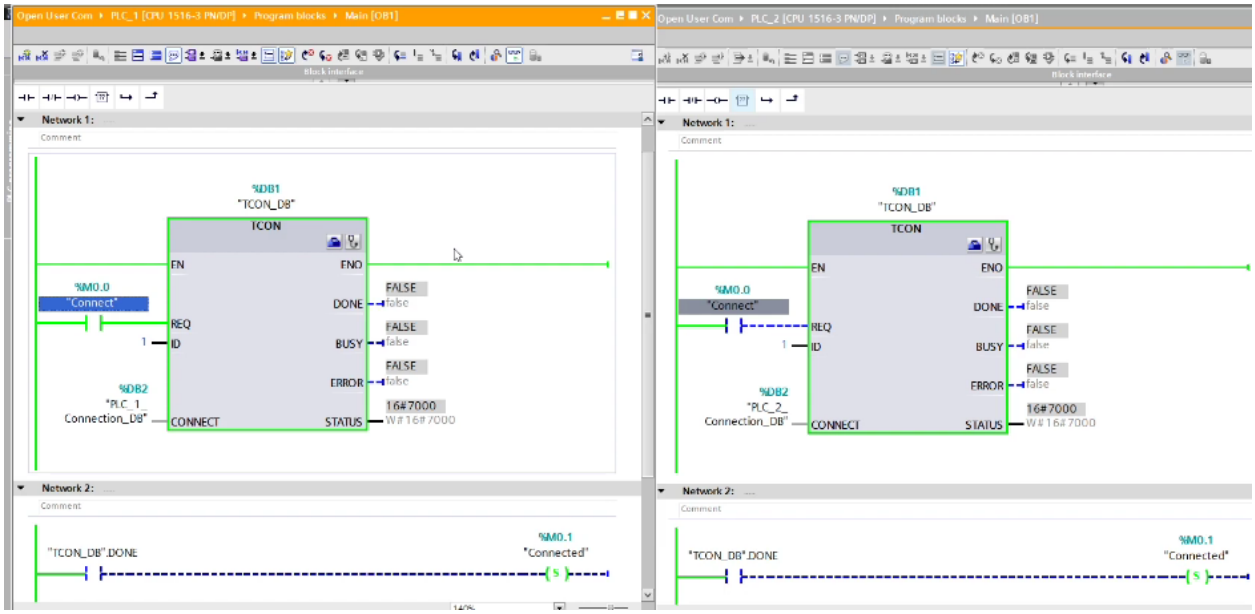


Figure 1.7 – Checking TCON unit's working process

1.3 Content of the report

- the goal of the work;
- description of the program with screenshots and photos;
- summary.

1.4 Questions for checking

1) Describe the TCON-TCP block: its purpose, functions, inputs and outputs;
2) Define the function of each input and output of the TCON function block.
3) What is the difference between passive and active controller? Why should we assign such roles?

Laboratory work №2. Breaking the PLC-PLC connection using the TDISCON block

The goal is to get acquainted with the block that allows you to break the connection between controllers, break the industrial network between two S7-300 controllers, create a configuration in the TIA Portal.

2.1 A brief of theory

Data exchange is controlled in the program by the following commands: TCON, TSEND and TRCV, TDISCON.

This laboratory work will introduce you to the TDISCON block, which terminates the communication connection from the CPU to the communication partner.

The significant advantage is the following: if the connection is terminated, for example due to a wire break or a remote communication partner, the active partner attempts to re-establish the configured connection. You do not have to issue the TCON command again.

If a TDISCON instruction is being executed or the CPU has entered the «STOP» state, then the existing connection is terminated and the created connection is deleted. To create and restore a connection, you have to issue the TCON command once again.

The inputs of the TDISCON block are: 1) REQ; 2) ID. Outputs: 1) DONE; 2) BUSY; 3) ERROR; 4) STATUS.

2.2 Work order

2.2.1 Firstly, open the project created in 1st laboratory work. Delete the elements in the 2nd network and add the TDISCON block from the «Communication» section (figure 2.1).

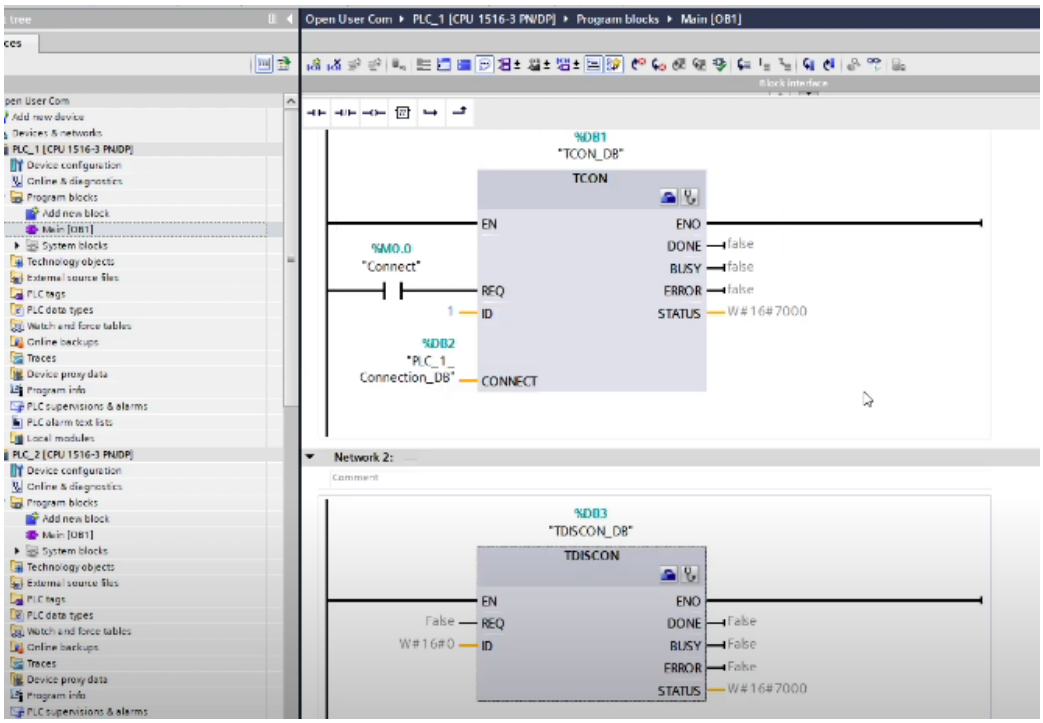


Figure 2.1 – TDISCON block is added to the network

2.2.2 Since ID = 1 is written in the TCON block, therefore, we write the same ID = 1 in the DISCON block, thereby showing the controller which connection to terminate. To the REQ input (the condition under which the interrupt will occur), we add the previously created m0.0 tag. Rename the tag from «Connected» to «Disconnected».

Therefore, according to the tag with the address m0.0, the controllers will have a connection with each other, according to the tag with the address m0.1, they will interrupt the existing connection (figure 2.2).

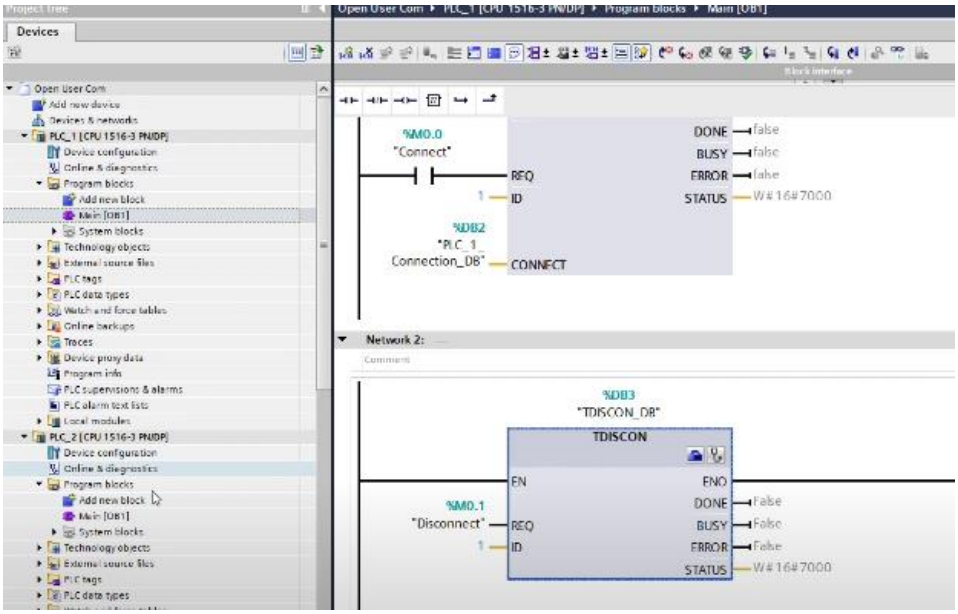


Figure 2.2 – 1st controller's disconnect configuration

2.2.3 Let us, next, configure the 2nd controller as well (figure 2.3).

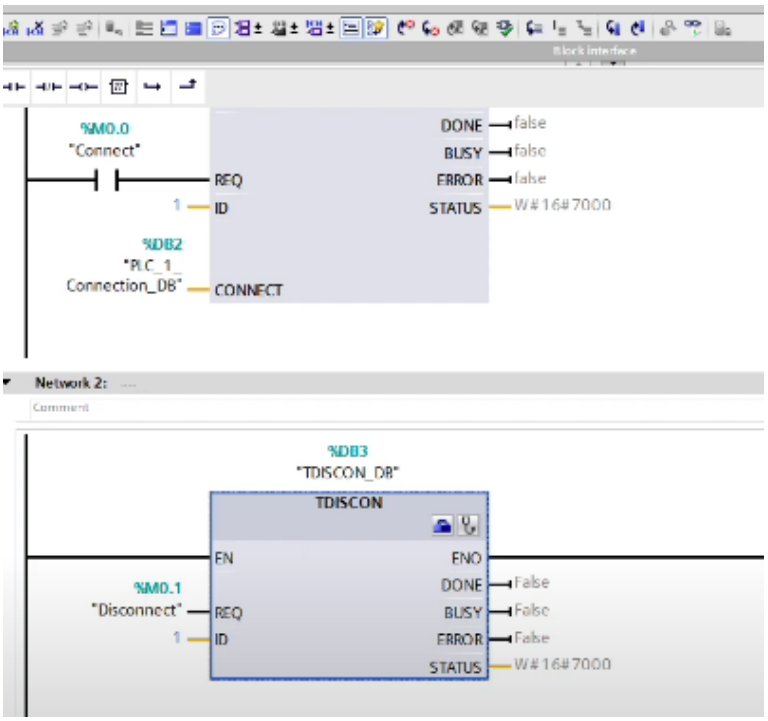


Figure 2.3 – 2nd controller's disconnect configuration

2.2.4 Then, we will download the programs to the controllers and test the network. Switch to the «Go online» mode. Try to analyze the operation of the TCON blocks for both controllers (figure 2.4). Has a connection been made?

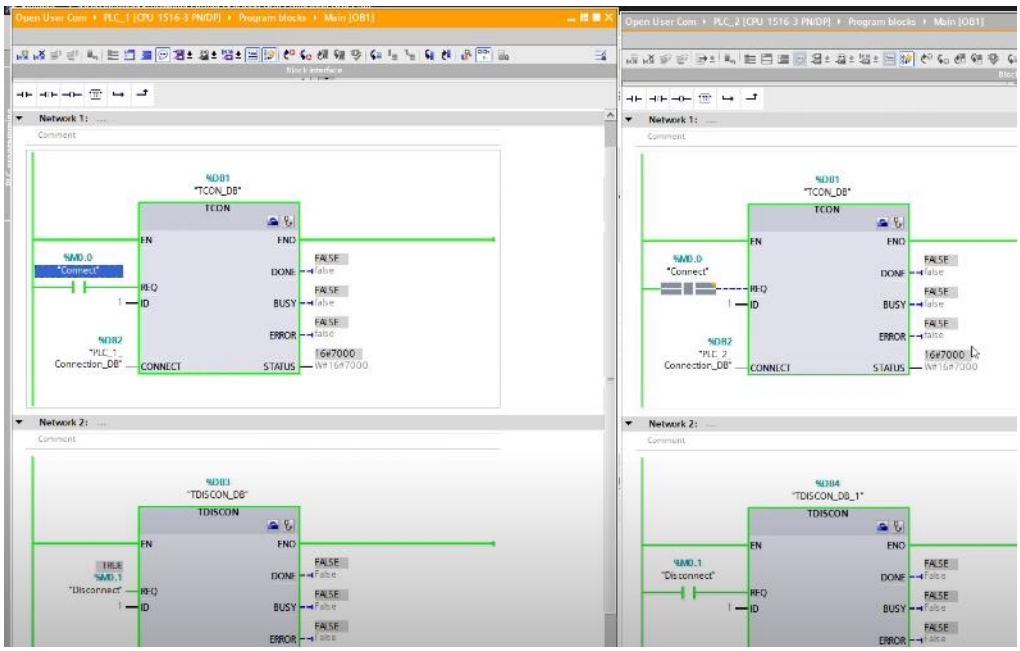


Figure 2.4 – Checking established communication connections

2.2.4 As with the previous laboratory work, the passive partner port (PLC2) has to be opened first. After opening, its status changes to 7002 (waiting for connection with active partner (PLC1). Then, changing the value of the condition (REQ) to PLC1, you will notice that now the statuses of both controllers are the same (7000). But try running TDISCON blocks, will it change anything?

2.2.5 No, nothing will change. Initilly you need to make sure that there is really a connection between the two controllers. To do this, go to the «Devices & networks» → «Connections» mode, where we will notice 4 connections. These are the connections between the controller and the computer (try to «Go offline» mode and check) (figure 2.4).

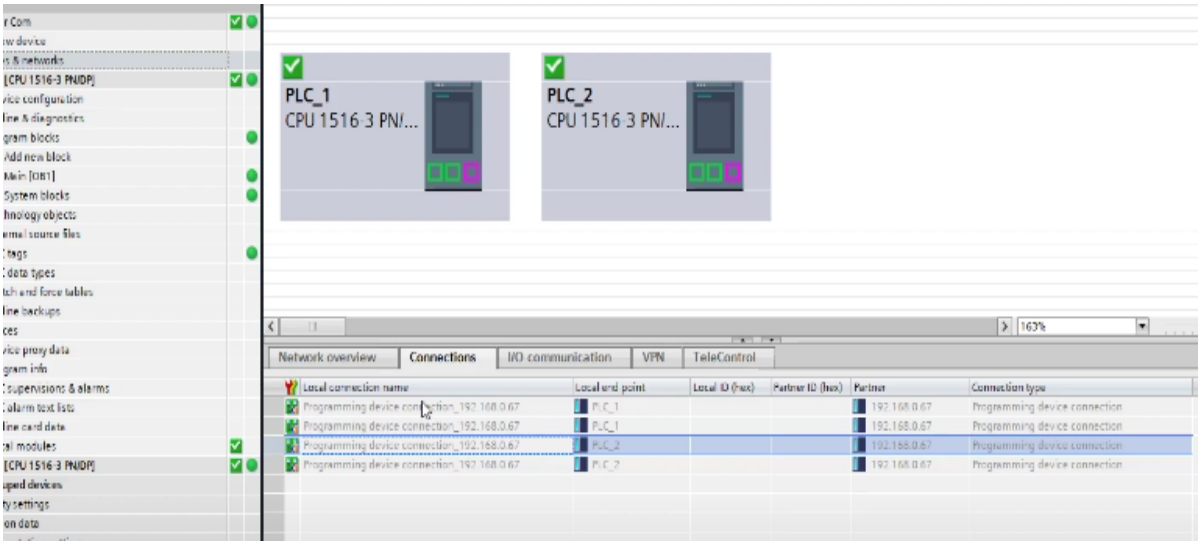


Figure 2.4 – Existing connections review

2.2.6 After that, we will switch back to the «Go online» mode. Then, open the PLC 2 port and see what connection appeared in the list (figure 2.5). What do you think needs to be done to establish a full connection?

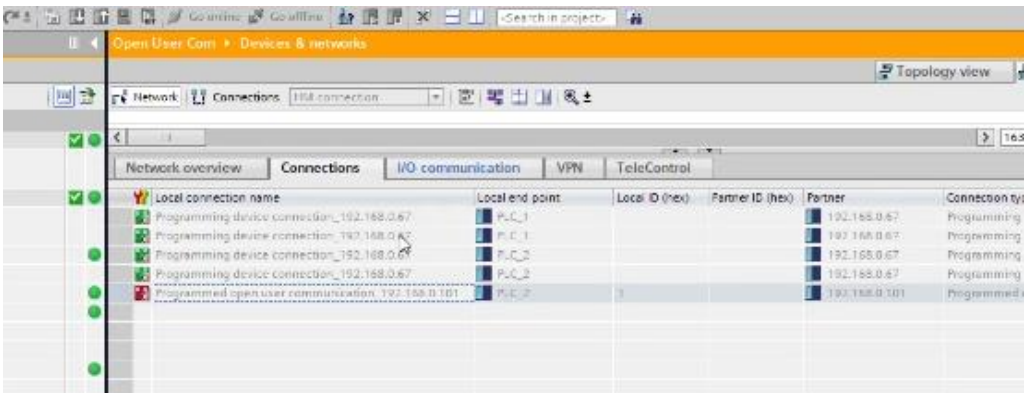


Figure 2.5 – Connection process

2.2.7 You need to open the active controller port firstly. Only then will the

connection be established (figure 2.6).

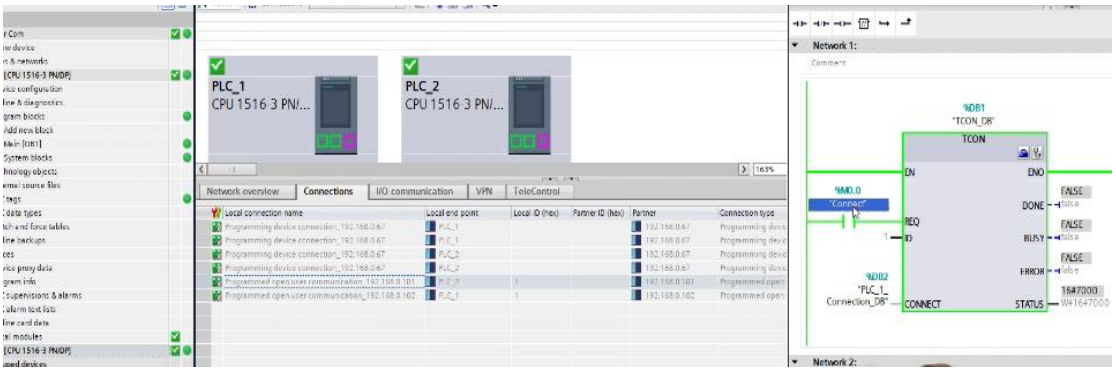


Figure 2.6 – Checking the communication connection for data exchange

2.2.8 Following the next stage, do the same with the TDICSON block and try to correctly interrupt the connection between the controllers yourself (figure 2.7).

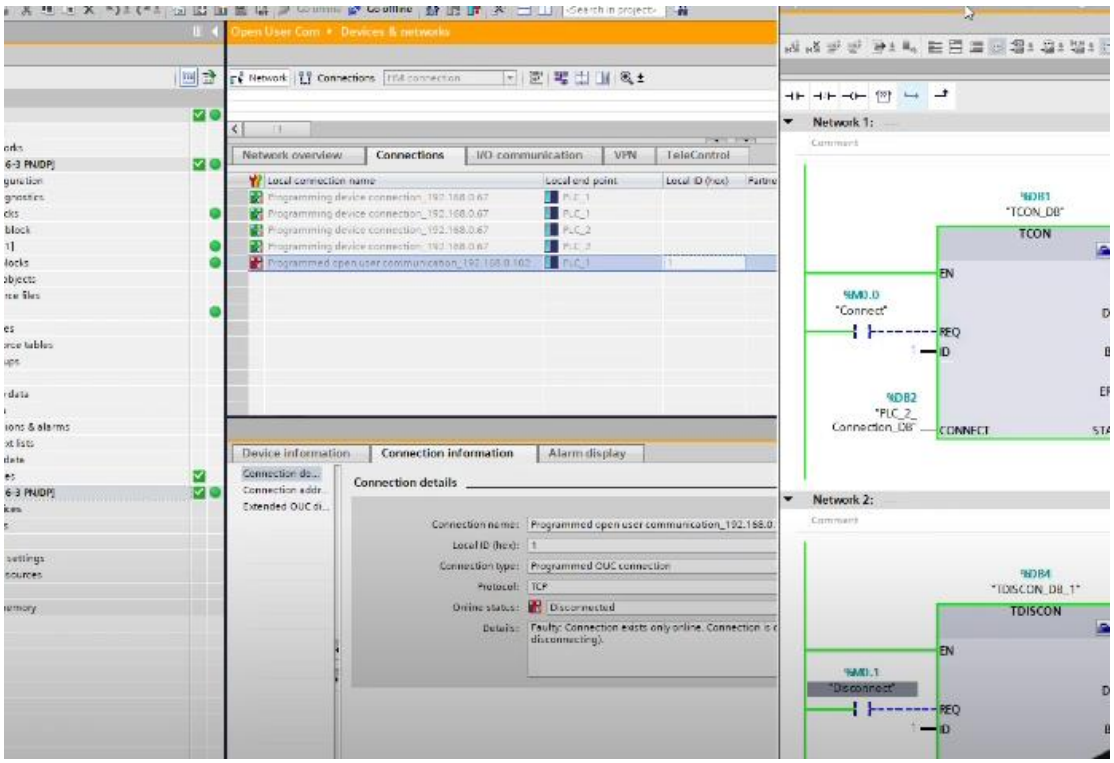


Figure 2.7 – TDISCON unit in progress

-the goal of the work;
- description of the program with screenshots and photos;
- summary.

2.4 Questions for checking

1) Describe the TDISCON block: its purpose, functions.

2) Define the function of each input and output of the TDISCON function block.
3) What is the difference between different controller configuration modes such as «Unspecified», «Multicast», «Broadcast»?

Laboratory work №3. Data exchange between controllers in an industrial network using the PUT block

The goal is to investigate the PUT unit's function for data exchange between controllers in an industrial network. Define the necessary parameters for using the PUT function. Explain how the function works

3.1 A brief of theory

With SFB/FB15 «PUT» you can write data to a remote CPU.

S7-300: Data is sent when a rising edge occurs at control input «REQ». After completing the job, you can assign new values to the parameters ID, ADDR_1 and SD_1

S7-400: The SFB is triggered when there is a rising edge at the control input «REQ». During the transfer, pointers to write destination areas («ADDR_i») and data («SD_i») are sent to the CPU of the communication partner.

The remote partner stores the required data at the addresses provided with the data and returns a confirmation of completion.

If there were no errors, this is indicated by the value 1 in the «DONE» status parameter on the next SFB/FB call. A new task can be activated again after the previous task has been completed.

The remote CPU can be in «RUN» or «STOP» mode. Errors and warnings are output using the «ERROR» and «STATUS» parameters if access errors occur while writing data, or if validation results in an error.

3.2 Work order

3.2.1 First of all create a project in TIA Portal, add devices: to the network as: two CPU 414F-3 controllers and two HMI panels, and connect all Ethernet ports of the devices to one bus (figure 3.1).

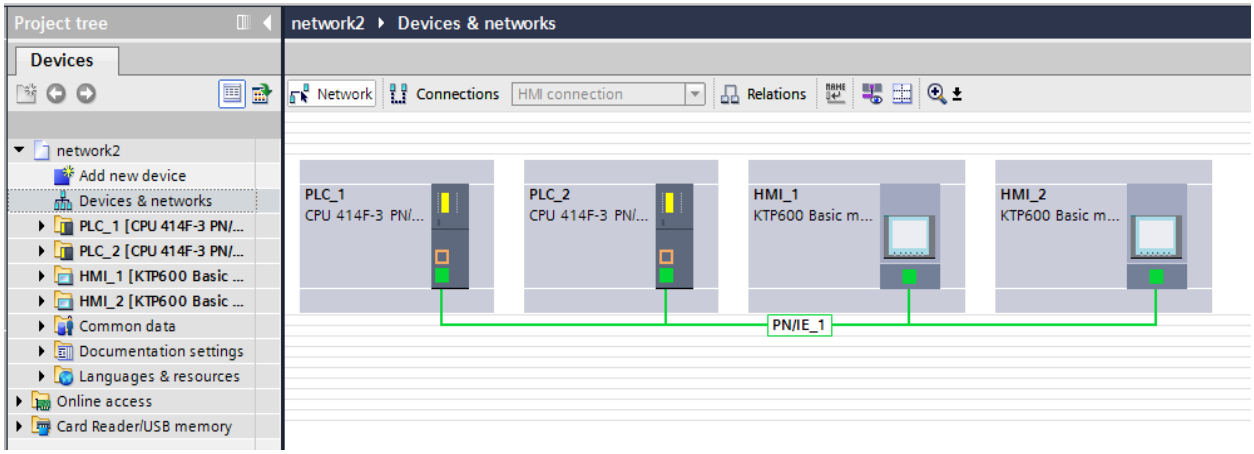


Figure 3.1 – Devices in the PROFINET network

3.2.2 Next, add the PUT function from the «Communication» → «S7 Communication» section to the «MAIN» block of the PLC_1 controller. In the properties of the block, you must specify the name of the partner controller, as shown on the figure 3.2, and the program will automatically configure the connection (figure 3.2).

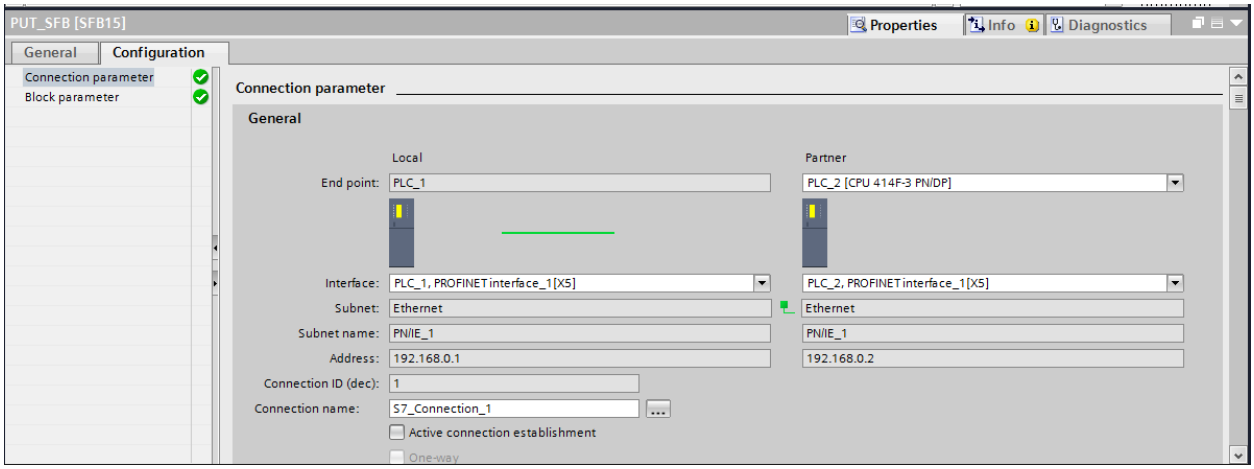


Figure 3.2–PUT unit’s properties

3.2.3 In addition, there is a need to specify the input parameters of the PUT block (figure 3.3). «REQ» - activates data exchange when a signal edge appears,» «ID» is set automatically, «ADDR_1» - pointer to the area in the partner's CPU where data will be written, «SD_1» - pointer to the area in the local CPU containing the data to be transferred.

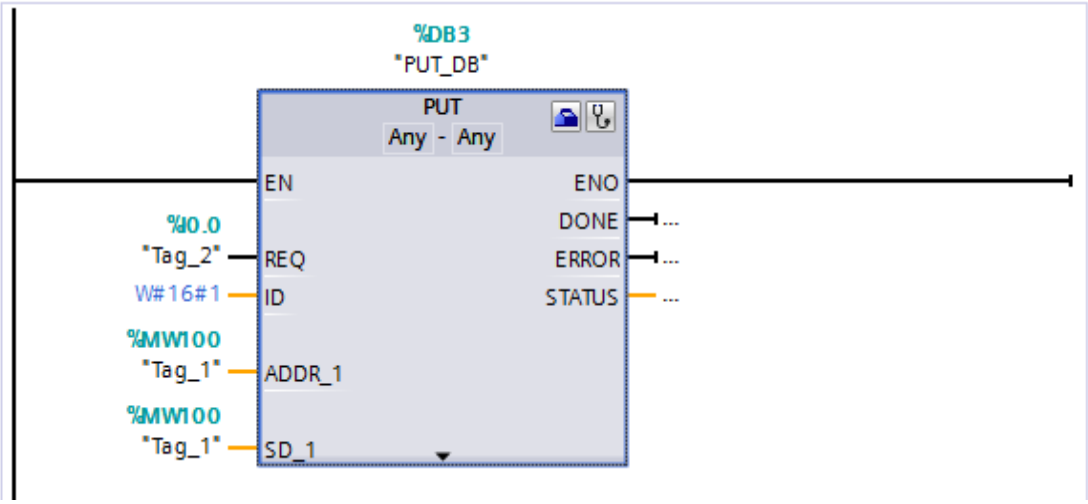


Figure 3.3 – Input settings in the PUT unit's

3.2.4 For the convenience of reading information, we set up the field for the output of the MW100 variable on both controllers and the «REQ» activation button on PLC_1 (figure 3.4).

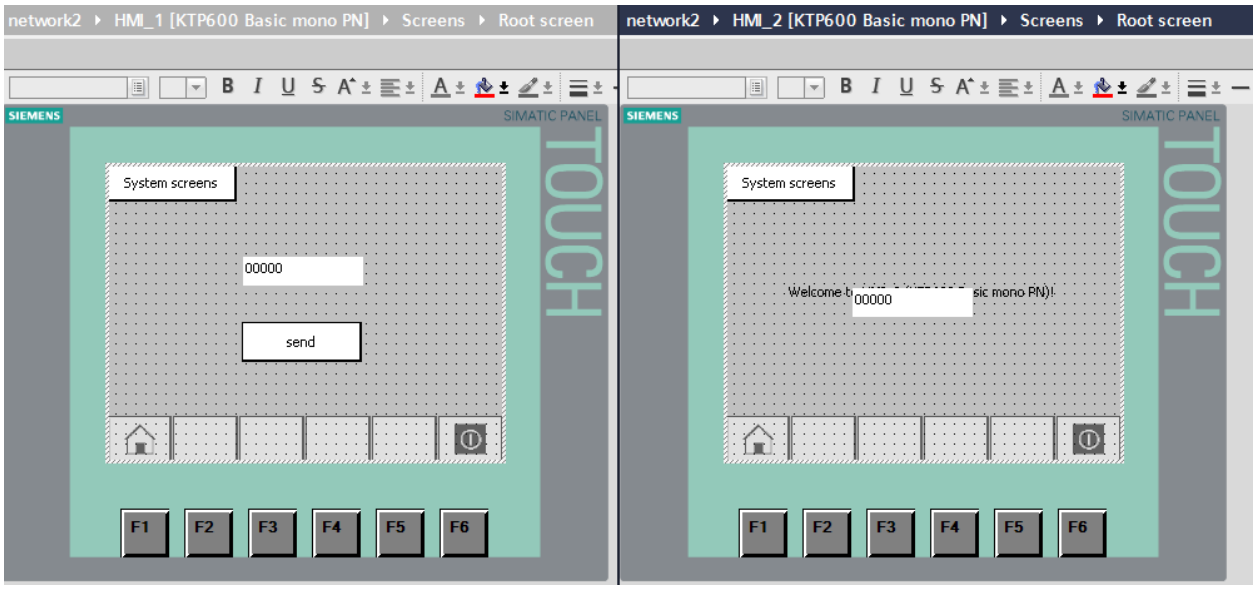


Figure 3.4 – HMI panel's view

3.2.5 To start the simulation in the «Devices & Networks» window, select PLC_1, and click the «Start Simulation» button (figure 3.5). The same must be repeated for PLC_2.

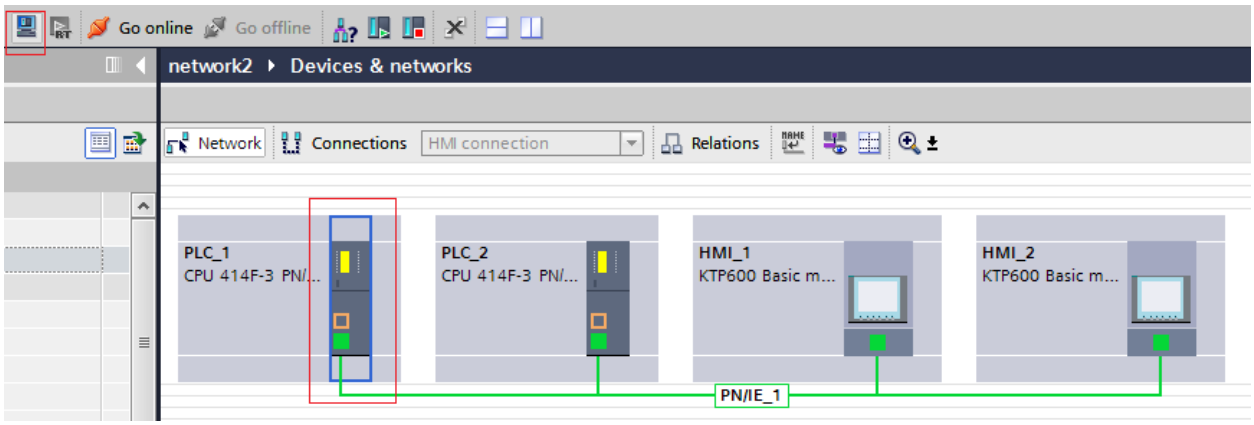


Figure 3.5 – Starting simulation process

3.2.6 To transfer data from controller to controller, you need to set the value of the variable at address MW100 on PLC_1 and press I0.0 or the «Send» button on HMI. As a result, the same value will be written to the PLC_2 controller at the address MW10 (figure 3.6).

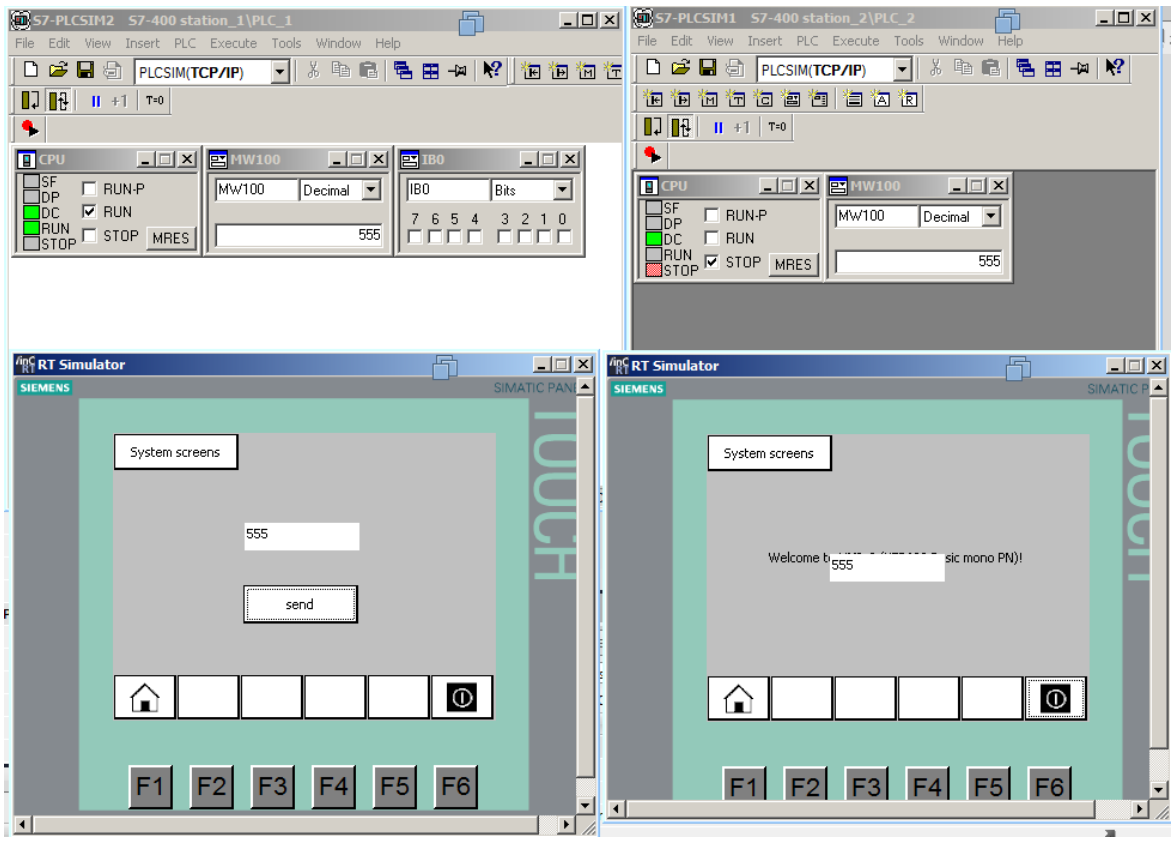


Figure 3.6 – The result of the PUT block's work

3.3 Content of the report

- the goal of the work;
- description of the program with screenshots and photos;

- summary.

3.4 Questions for checking

1) Make a conclusion on the work done;
2) Explain the PUT command and block, what is it for?
3) What input parameters of the PUT block did you use, explain what their purpose is?

Laboratory work №4. Creating an industrial network on S7-300 controllers using PUT, GET blocks

The goal is to connect controllers in the PROFINET network, create a configuration with a connection in the TIA Portal, implement data exchange between test benches using the PUT and GET functions

4.1 A brief of theory

S7 connections used for information exchange between S7-300, 400, 1200, 1500 series devices are configurable. This means that you must explicitly tell devices who is talking to whom. In general, these connections are bidirectional, i.e. the addition of a configurable connection occurs for both PLCs, which results in the need to download both PLCs. This does not cause any particular problems if both PLCs are in the department of the process control system of the enterprise, and both PLCs have up-to-date application software on hand. But there are also special cases - it is necessary to establish an exchange between the «old» S7-300 and the «new» S7-1500, moreover, the source codes of the old program have been lost.

In such cases, it is possible to set up a one-way connection, make it active (that is, «our» PLC will initiate the connection), add PUT/GET communication POUs, and download only one PLC.

GET is for reading data. To transfer information (records) to a remote controller, we use the PUT call.

4.2 Work order

4.2.1 At the beginning, we have to create a project in TIA Portal, add network devices: two CPU 314C-2 PN / DP controllers and a KTP600 Basic Color PN HMI panel, and connect all Ethernet ports of the devices to one bus (figure 4.1).

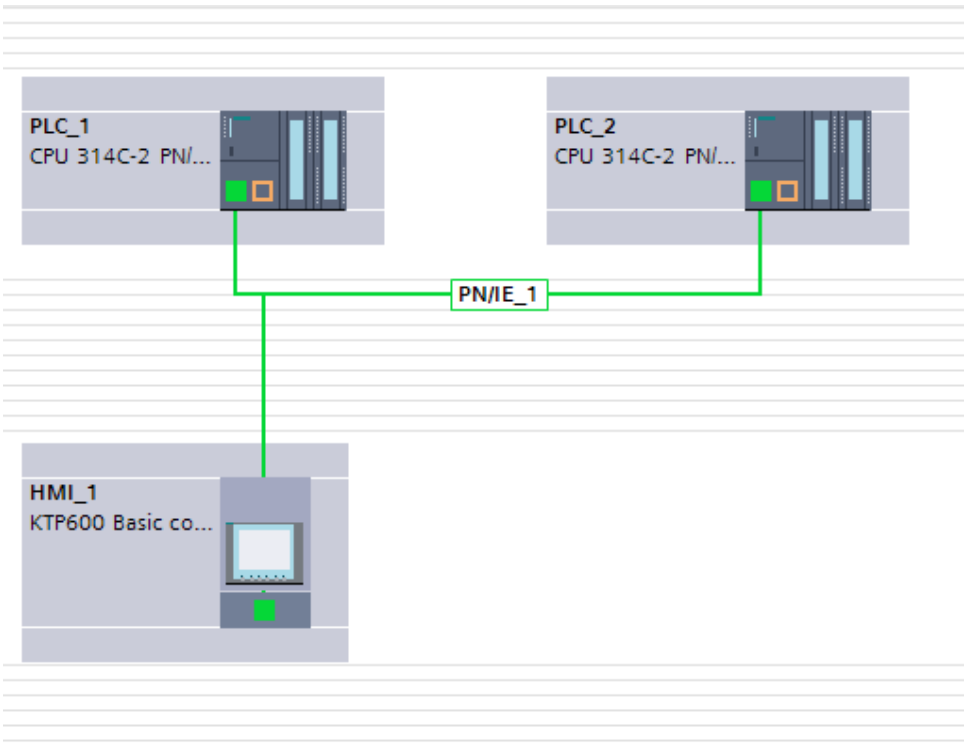
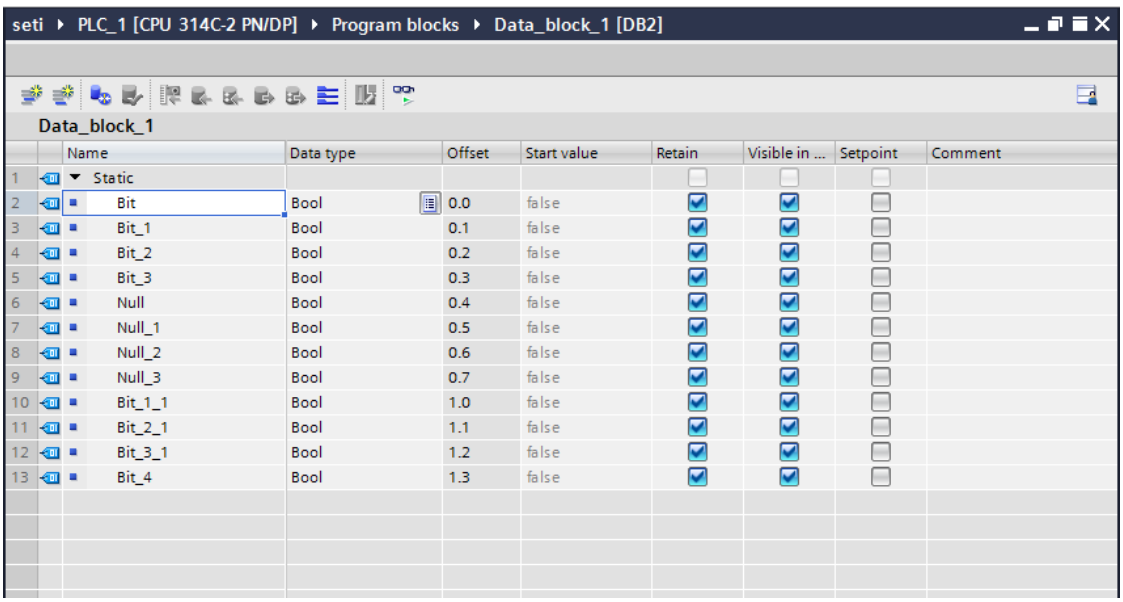


Figure 4.1 – Devices included in to the PROFINET industrial network

4.2.2 The data from the buttons will be written to a data block that will be transferred from one controller to another. To do this, you need to create a data block in each of the controllers (figure 4.2). The first 4 bits of the zero byte store the states of the buttons on the first controller, which will be written to the same addresses in the second, and the first 4 bits of the first byte store the states of the buttons on the second controller, which will be written to the same addresses in the first controller. Thus, the data blocks in both controllers look identical.



	Name	Data type	Offset	Start value	Retain	Visible in ...	Setpoint	Comment
1	Static							
2	Bit	Bool	0.0	false	☑	☑	☐	
3	Bit_1	Bool	0.1	false	☑	☑	☐	
4	Bit_2	Bool	0.2	false	☑	☑	☐	
5	Bit_3	Bool	0.3	false	☑	☑	☐	
6	Null	Bool	0.4	false	☑	☑	☐	
7	Null_1	Bool	0.5	false	☑	☑	☐	
8	Null_2	Bool	0.6	false	☑	☑	☐	
9	Null_3	Bool	0.7	false	☑	☑	☐	
10	Bit_1_1	Bool	1.0	false	☑	☑	☐	
11	Bit_2_1	Bool	1.1	false	☑	☑	☐	
12	Bit_3_1	Bool	1.2	false	☑	☑	☐	
13	Bit_4	Bool	1.3	false	☑	☑	☐	

Figure 4.2 – Data block

4.2.3 Next, add the PUT function from the «Communication -> S7 Communication» section to the «MAIN» block of the PLC_1 controller. In the properties of the block, you must specify the name of the partner controller, as shown below (figure 4.3,) and the program will automatically configure the connection. The same must be done for the GET function.

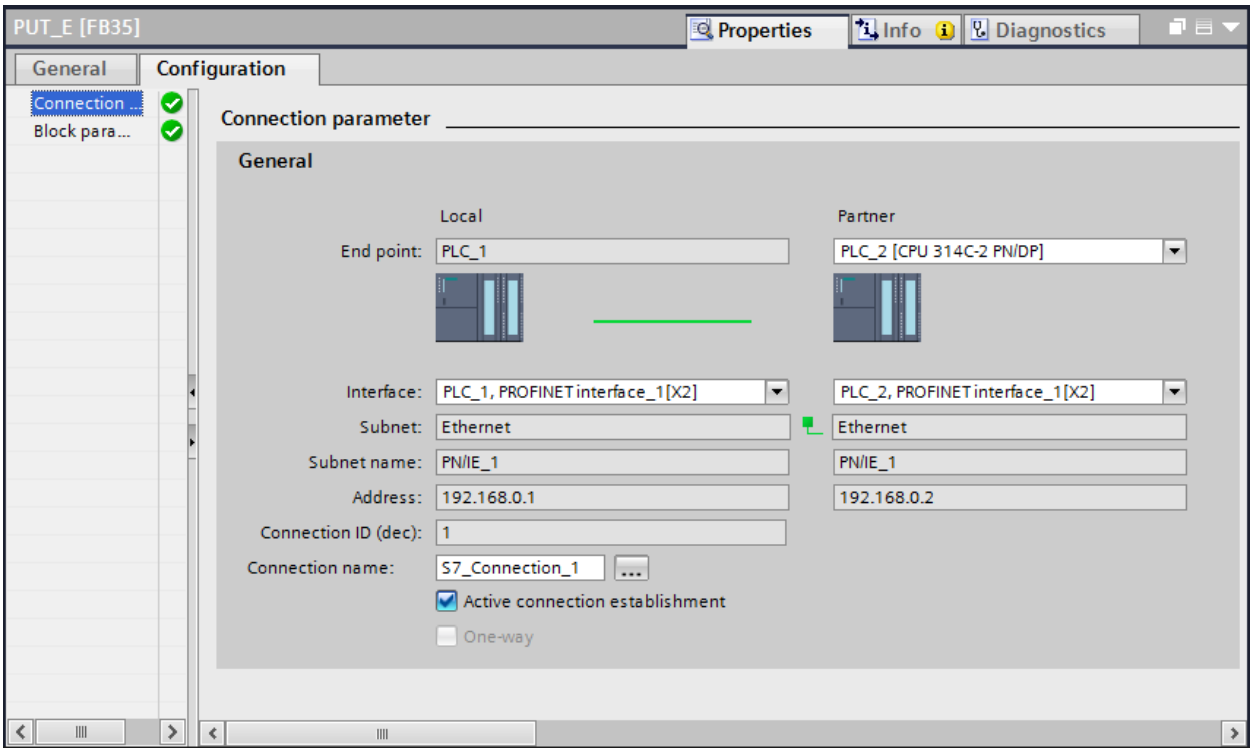


Figure 4.3 –PUT unit's configuration

4.2.4 In addition, we have to specify the input parameters of the PUT and GET blocks (figure 4.4). «REQ» - activates data exchange when the edge of the signal appears, «ID» is set automatically, «ADDR_1» - pointer to the area in the partner's CPU to which data will be written (from where the data will be transferred in case of GET), «SD_1» - pointer to the area in the local CPU containing data, to be transferred, «RD_1» is a pointer to the area in the local CPU where the data will be written.

The «REQ» signal is generated in OB35 at regular intervals and is reset in OB1 (Main) after the PUT and GET functions have been executed.

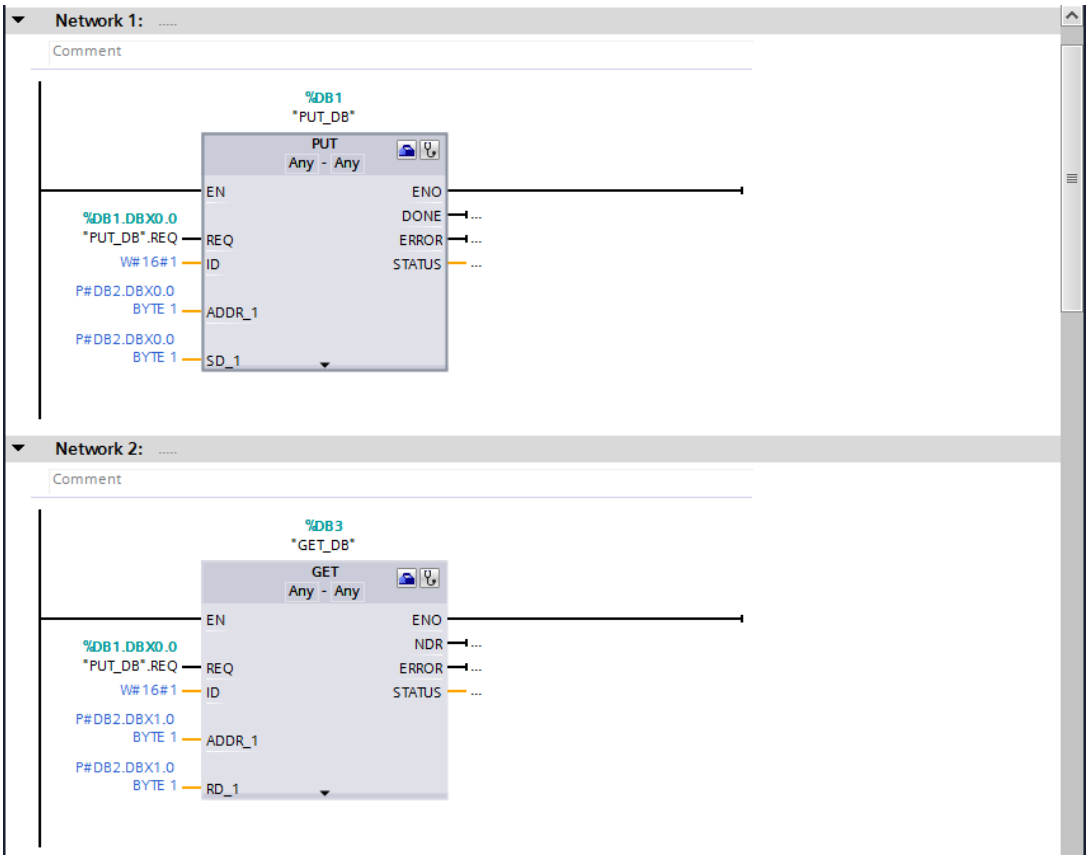


Figure 4.4 – PUT and GET functions

4.2.5 For a better understanding, let us look at the characters in the «ADDR_1» field of the PUT block. In our case, this is P#DB2.DBX0.0.BYTE 1. So, DB2 - we read the data block with number 2; DBX0.0 - offset in the data block 0 bytes and 0 bits; BYTE 1 - read bytes in the amount of 1 piece

Thus, the pointer P#DB2.DBX0.0BYTE1 corresponds to the absolute address of DB2.DBD0. Reminding you that we are talking about the memory addresses of the remote communication partner, about which there is no data in our local TIA Portal project.

4.2.6 In order to write and output data to data blocks, you need to set a simple program in «Main» section (figure 4.5). For PLC_2, the program is the same, except that the bits from the I area are written to the bits of the first byte, and the bits from the first byte are written to the Q area.

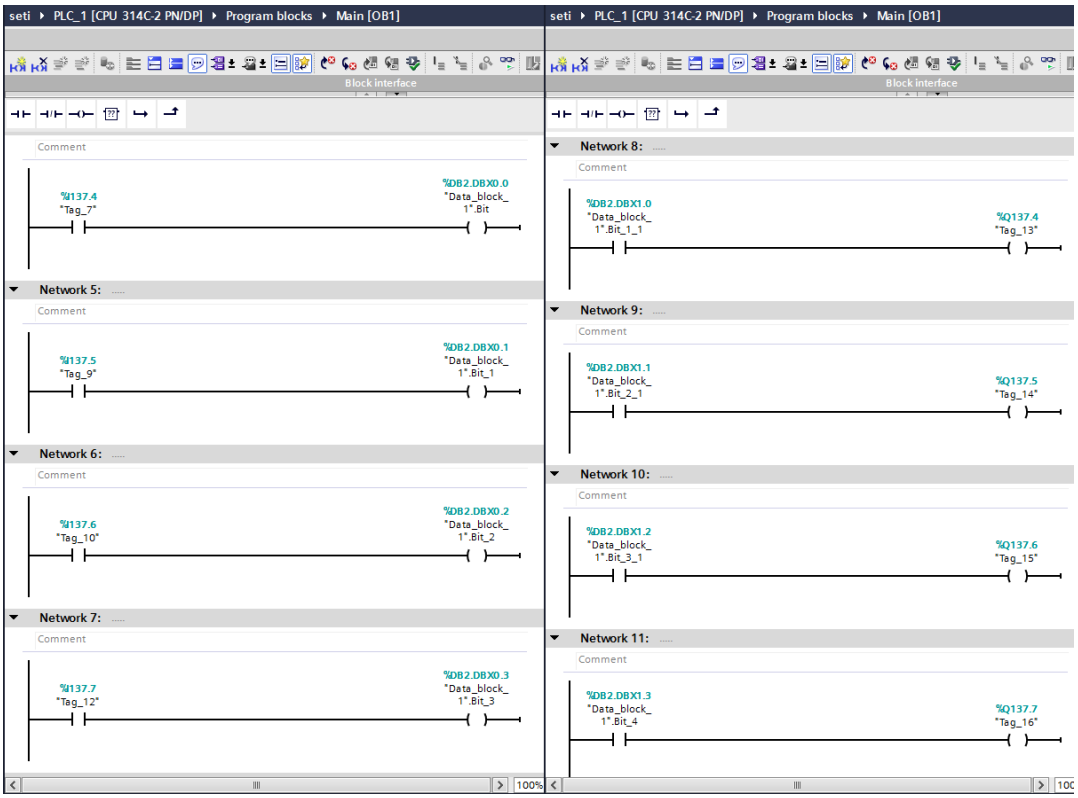


Figure 4.5 – «Main» PLC_1 unit

4.2.6 Next, you need to download the configurations to the controllers.

4.3 Content of the report

- the goal of the work;
- description of the program with screenshots and photos;
- summary.

4.4 Questions for checking

1) Explain the characters P#DB2.DBX0.0.BYTE 1 in the ADDR_1 field of the PUT block.
2) What is the main function of PUT, GET blocks? Show an example of using these commands.
3) Describe the functions of all inputs and outputs of the communication programming block GET.

Laboratory work №5. Organizing an industrial network for data exchange using TSEND_C, TRCV_C blocks

The goal is to implement data exchange between industrial network controllers using the TSEND_C + TRCV_C functions

5.1 A brief of theory

The TSEND_C command establishes a connection with a partner controller via TCP or ISO on TCP, sends data, and may terminate the connection. Once a connection has been established and created, it is automatically maintained and controlled by the CPU. The TSEND_C command combines the functions of the TCON, TDISCON, and TSEND commands.

The minimum size of data that you can send with the TSEND_C command is one byte.

The TRCV_C instruction establishes a connection with the partner CPU via TCP or ISO on TCP, receives data, and can terminate the connection. Once a connection has been established and created, it is automatically maintained and controlled by the CPU. The TRCV_C command combines the functions of the TCON, TDISCON, and TRCV commands.

The minimum data size that you can receive with the TRCV_C command is one byte. The TRCV_C instruction does not support the transfer of boolean data or boolean arrays.

5.2 Work order

5.2.1 First of all, create a network consisting of 2 controllers. Then, add a data block to the first controller for transmission, and to the second controller - for reception. And also for the data block, deactivate the «Optimized Block Access» property in the «Properties» tab (figures 5.1, 5.2).

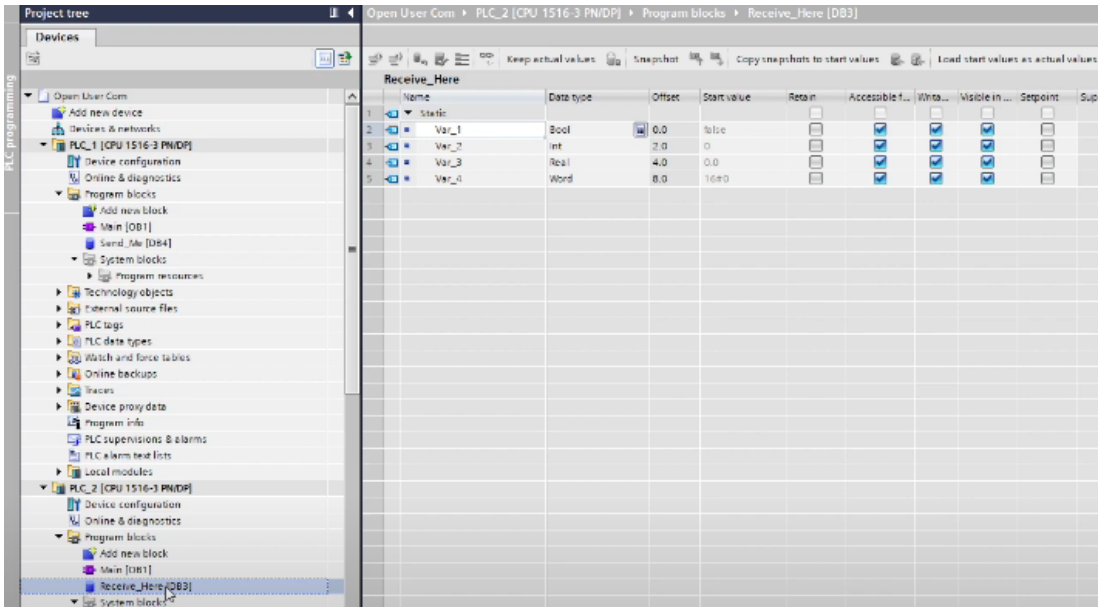


Figure 5.1 – Data received by the 1st controller

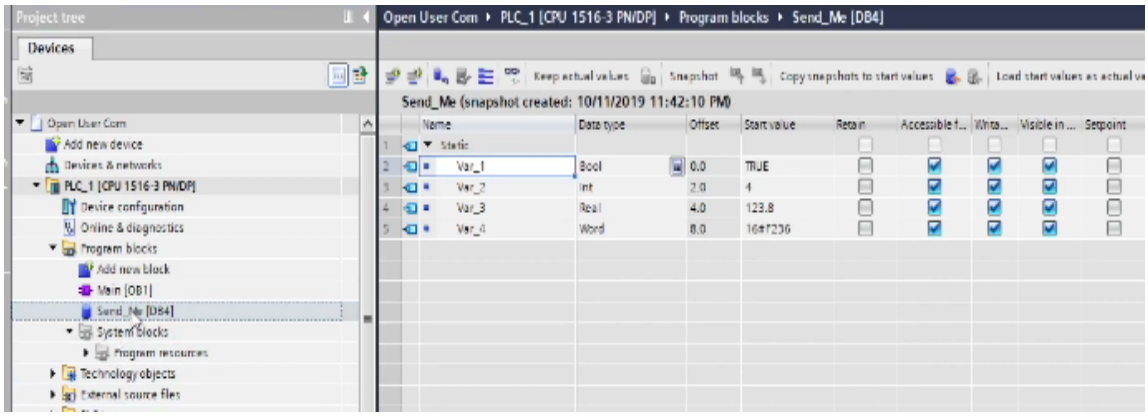


Figure 5.2 – Data sent by the 2nd controller

5.2.2 Afterwards, add the TSEND_C block to the main programming block. To the «DATA» input we will transfer the list of variables that you created in the first paragraph. Add a button to the «REQ» input, pressing which activates the action of the communication program block (figure 5.3).

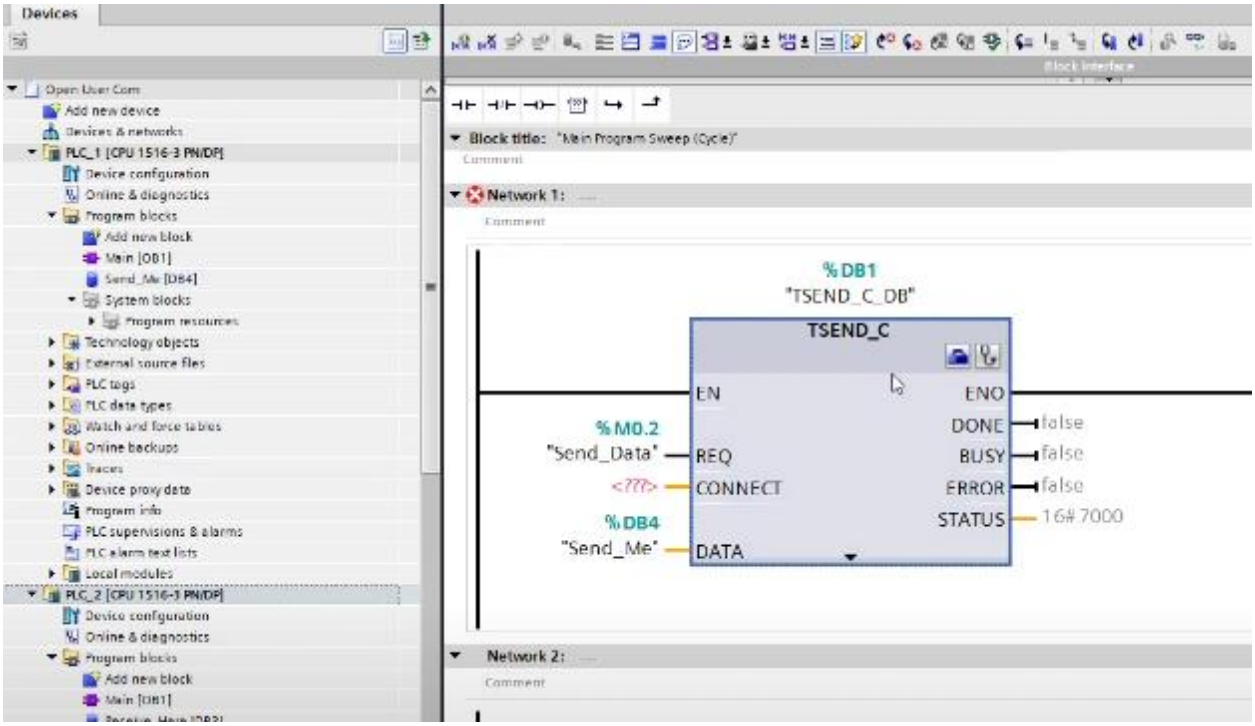


Figure 5.3–TSEND_C unit

5.2.3 It remains to configure the outputs of this block. To do this, go to the «Start Configuration» tab (blue briefcase). Partner view as «Unspecified» means that it is possible to connect any device that has an IP address. After setting, click on «Block Parameter» to activate it (figure 5.4)

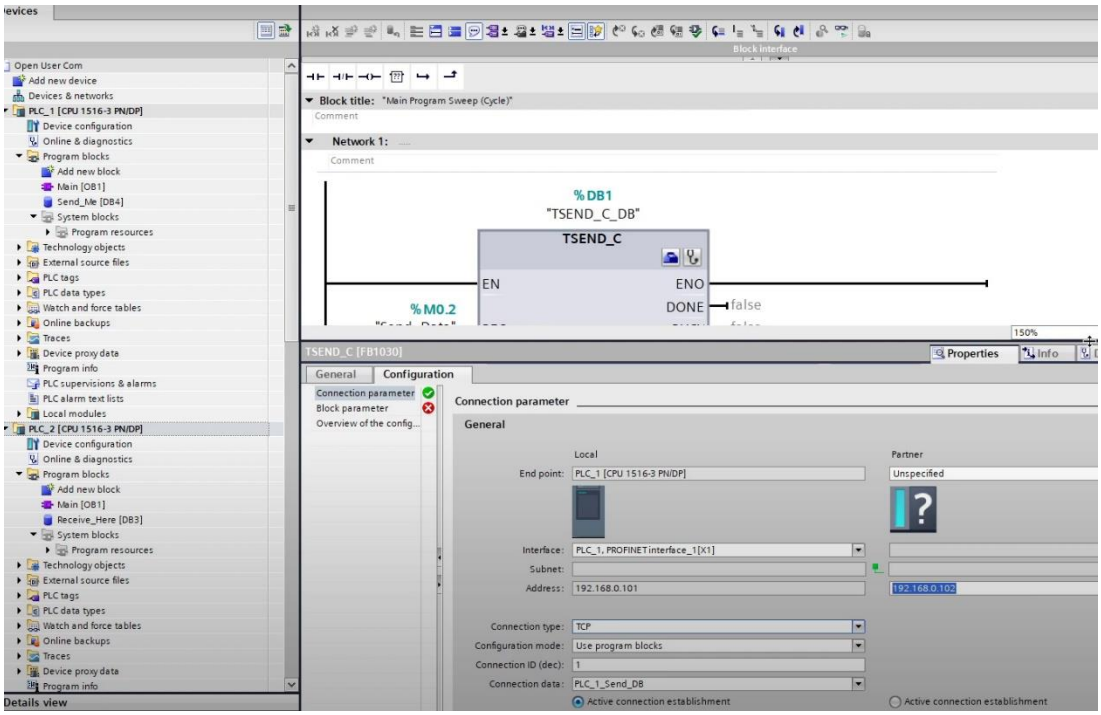


Figure 5.4 – Setting up controllers and setting the operating mode of the 1st controller

5.2.4 Download the program. Go to the «Go online» tab. What do you see? Explain why this is happening? (figure 5.5).

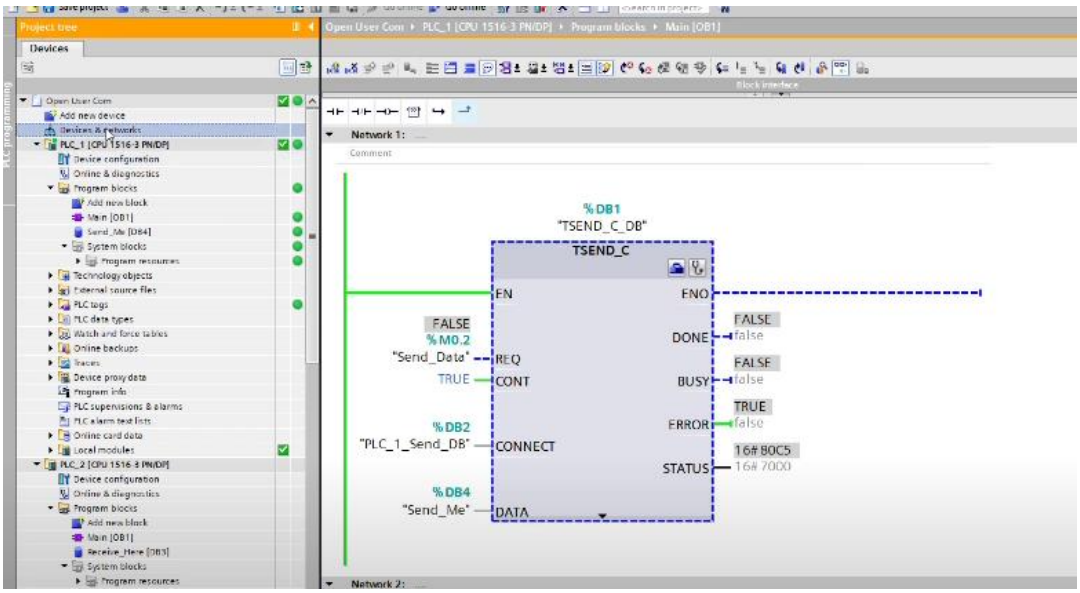


Figure 5.5 – Simulation start for the 1st controller

5.2.5 Go to the main program block of the 2nd controller. Since we have already used the TSEND_C block - send data, therefore the 2nd controller must have a TRCV_C function block - receive data. In order to do this, drag to the «DATA» input - a list of tags created in step 1, where the received information will be written.

Then, configure it the same way. Since the 1st controller is the transmitting one, it will be necessary to check the item «Active connection establishment» and activate the «Block Parameter» tab (figure 5.6).

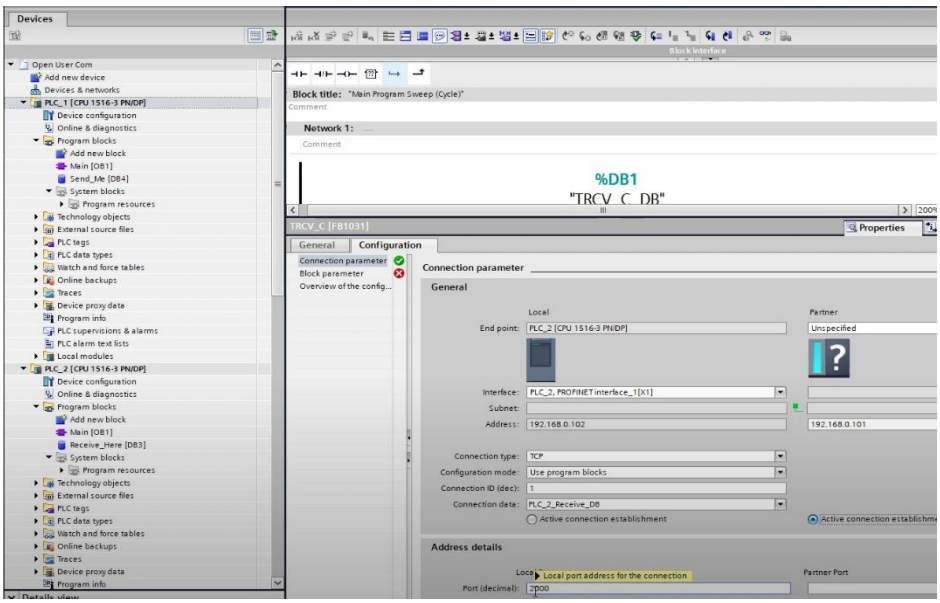


Figure 5.6 – 2nd controller's configuration

5.3.5 Change the EN-R input from «FALSE» to «TRUE» to always receive data. In general, this input means under what condition the data will be accepted. Download the program to the PLC(figure 5.7).

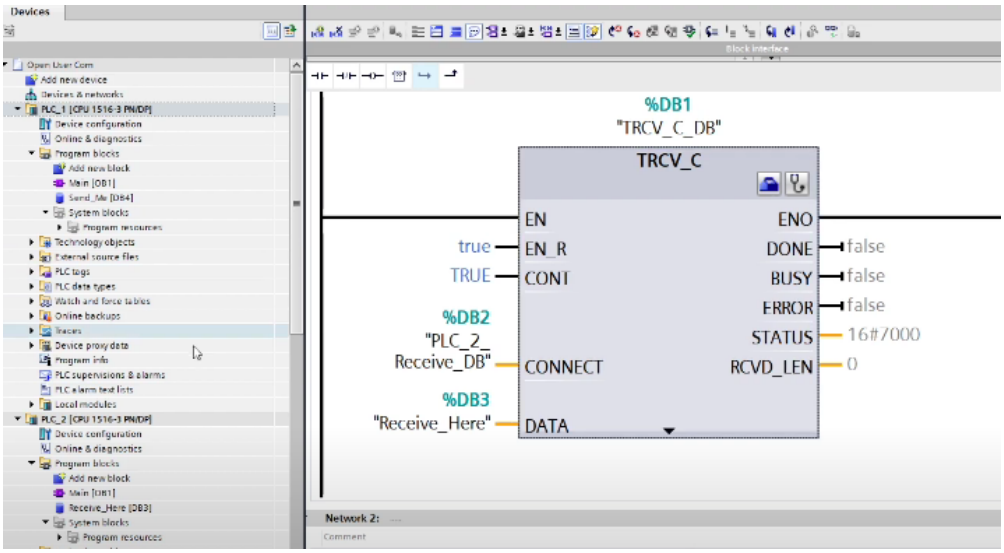


Figure 5.7– Setting the data receiving unit

5.3.6 Start the process by switching to the «Go online» mode and check if the process of transferring data from one controller to another has been completed (figure 5.8).

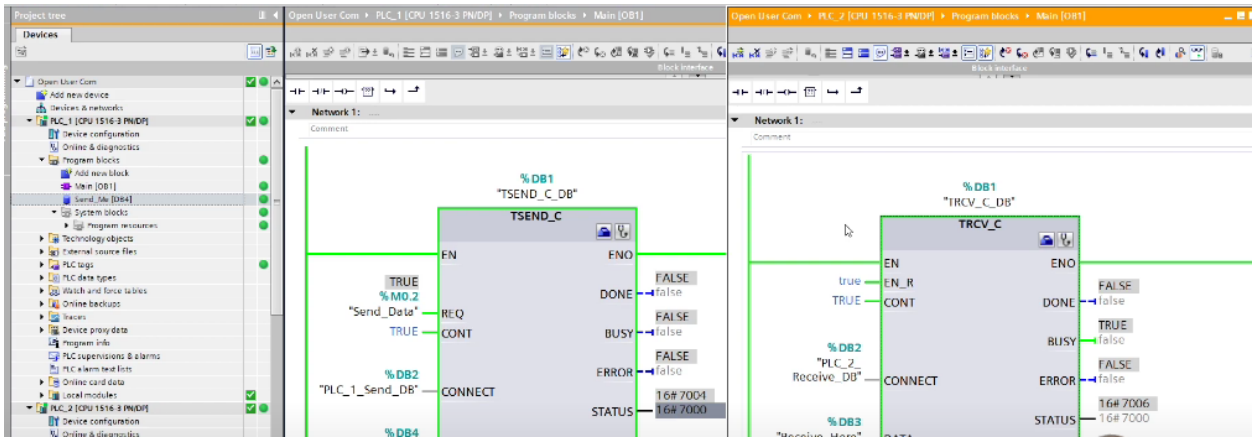


Figure 5.8 – Connection establishment

5.3 Content of the report

- the goal of the work;
- description of the program with screenshots and photos;
- summary.

5.4 Questions for checking

1) Explain how the connection between the controllers is made.
2) Describe the functions of each input and output of the TSEND_C, TRCV_C function block.
3) What are the types of data in the Tia Portal environment?

Laboratory work №6. Creating a user interface for a web page based on the S7-1200 controller

The goal is to learn how to create a web page user interface for viewing and changing variables in the S7-1200 controller program

6.1 A brief of theory

The widespread development of the production of programmable logic controllers and, in parallel, the impact on all areas of the global Internet have led to the emergence of an embedded web server in the controllers, designed to monitor and control the operation of the controller.

The introduction of the Internet made it possible to use a web server to access web pages hosted on a server defined by controllers, access them, and view or change variables in the program.

The embedded web page mechanism allows any user to remotely access the controller without the need for additional software.

6.2 Work order

6.2.1 The procedure for creating a custom web page is shown below. (figure 6.1).

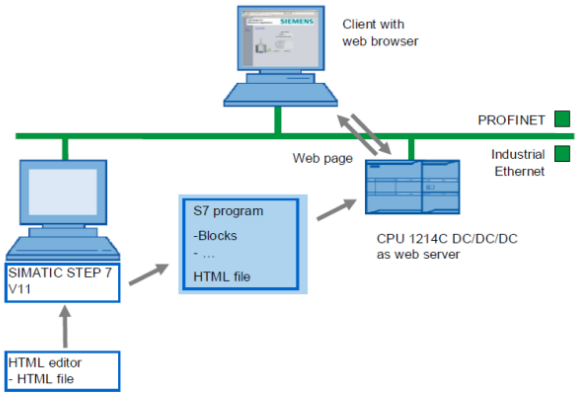


Figure 6.1 – Procedure for creating a web page

6.2.2 The whole process presented in the figure 6.1 can be described as follows. First, an HTML file is created using the «Notepad» program or the «Notepad» editor. This file in the browser is a custom web page with the necessary input and output fields.

Then, in the TIA Portal environment, a project is created for the S7-1200 controller, which, when configured, specifies a folder that contains a custom web page. The commands of the HTML file get into data blocks, and then the entire project is loaded into the controller.

After starting the project, the web server program, working together with the controller hardware and the user program, transfers the current data to the web page. By accessing this web page with a browser, you can also change this data.

6.2.3 Let us create a custom web page showing the current state of the variables from the controller program - Tag_1, Tag_2. Using the «Start», «Stop» buttons, it will be possible to change the state of boolean variables.

As a result, we get the following listing (figure 6.2).

```
<!-- AWP_In_Variable Name="Start" -->          (эти команды необходимы
<!-- AWP_In_Variable Name="Stop" -->           для записи переменных в контроллер)
<html>
    <head>
        <title>Start Stop</title>
        <meta http-equiv="Content-Language" content="en" >
        <meta http-equiv="Content-Type" content="text/html; charset=utf-8" >
            <meta http-equiv="refresh" content="10; URL=Start_Stop.htm">
                (частота обновления веб-страницы в сек.)
    </head>
    <body bgcolor="#D0D3DA">
        <table border="1" >
            <tr>    (создание поля для вывода Tag_1)
                <td class="static_field">Tag_1:</td>
                <td class="output_field">:="Tag_1":</td>
            </tr>
            <tr>    (создание поля для вывода Tag_3)
                <td class="static_field">Tag_3:</td>
                <td class="output_field">:="Tag_3":</td>
            </tr>
            <tr>    (создание кнопки Start)
            <td class="static_field_button">
                <form method="post" action="">
                <input type="submit" value="Start" style="height: 30px; width: 100px">
                <input type="hidden" name="Start" value="1">
                <input type="hidden" name="Stop" value="0">
                </form>
            </td>
            </tr>
            <tr>    (создание кнопки Stop)
            <td class="static_field_button">
                <form method="post" action="">
                <input type="submit" value="Stop" style="height: 30px; width: 100px">
                <input type="hidden" name="Stop" value="1">
                <input type="hidden" name="Start" value="0">
                </form>
            </td>
            </tr>
        </table>
    </body>
</html>
```

Figure 6.2 – Web-page code

6.2.4 Following that, you need to save this file to your PC. When you open this file in a web browser window, you will receive the following page image (figure 6.3).

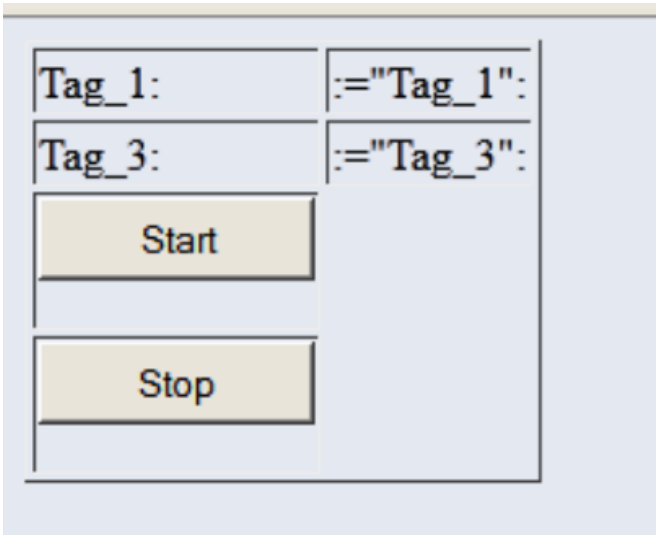


Figure 6.3 – View of a web page in a browser window

6.2.5 Next, in the TIA Portal, we create a project for the S7-1200 controller, set its Ethernet address, for example, 192.168.0.1, activate the Web Server, and also

specify the storage location for the user web page we created earlier. In the CPU properties window, click on «Generate blocks», as a result of which we will see that the commands of the HTML file automatically fall into the data blocks DB333, DB334 of the project. In the «Protection» tab, specify a password to protect information. The result obtained is shown below (figure. 6.4).

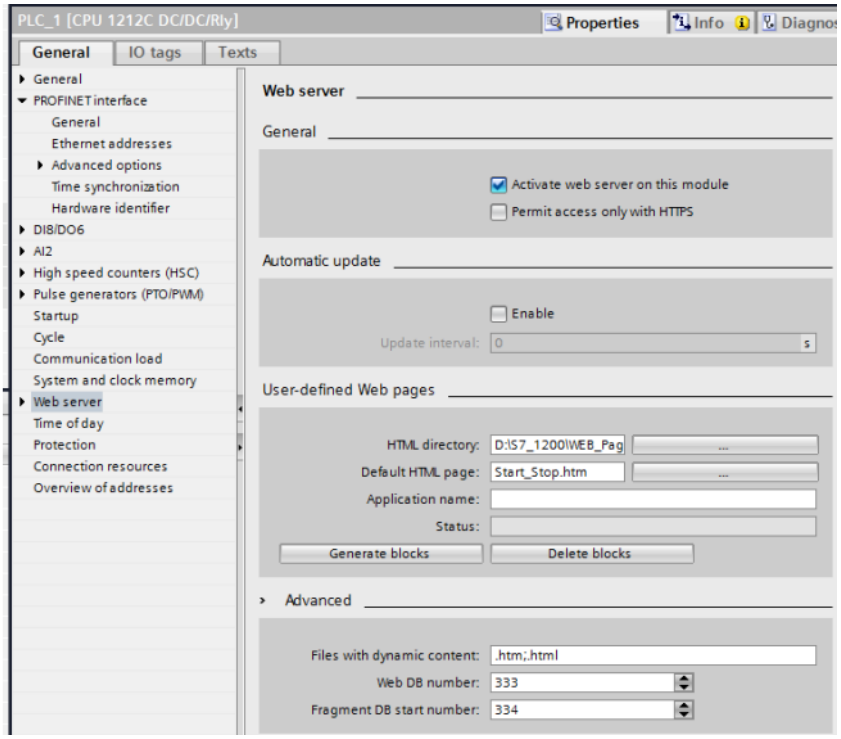


Figure 6.4 – CPU property configuration

6.2.6 Afterwards, we create a table of variables (tags) presented below (figure 6.5).

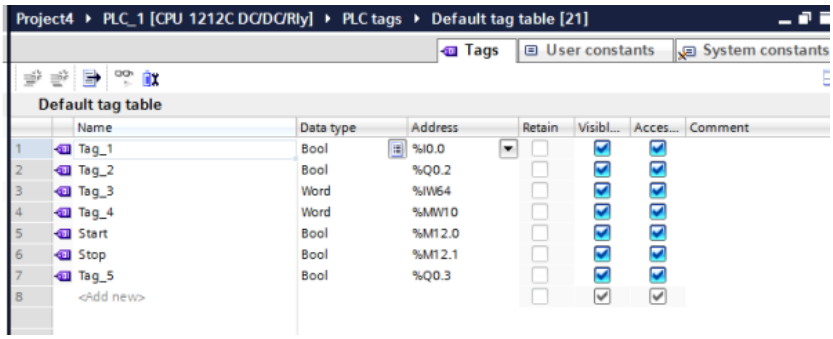


Figure 6.5 – Tag table

6.2.7 In addition, we have to create a program in the LAD language, using the «MOVE», «RS», «WWW» blocks (the function provides interaction between the DB333 and the controller's web server). It is simple and intended to show how

variables are changed, which can then be seen in the browser window (figure 6.6).

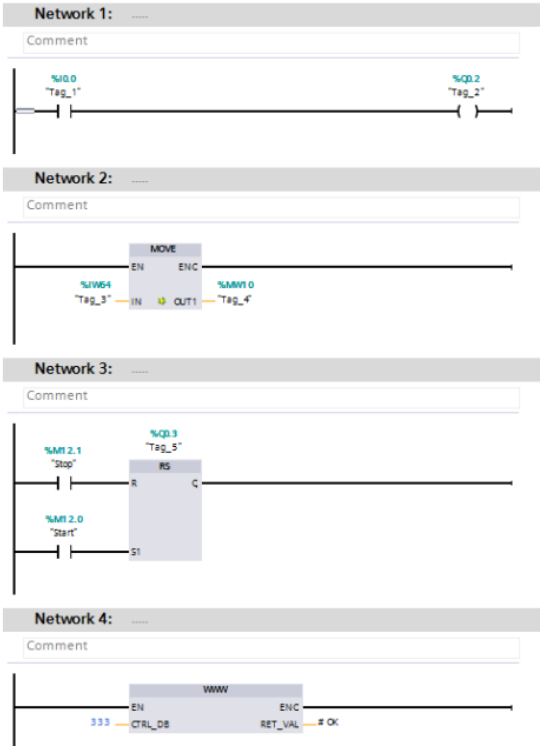


Figure 6.6 – Program in LAD

6.2.8 After checking, using «Download to device» button load the project into the controller and launch the browser with the address you specified (for example, 192.168.0.1). The controller's start web page will be displayed. In order not only to view, but also to change the variables in the program, you will need to go to the page with administrator rights, specifying in «Name» - «Admin» and the password specified earlier in the «Protection» tab. Switching to the «User Pages» tab on the start page, you can see the result (figure 6.6).

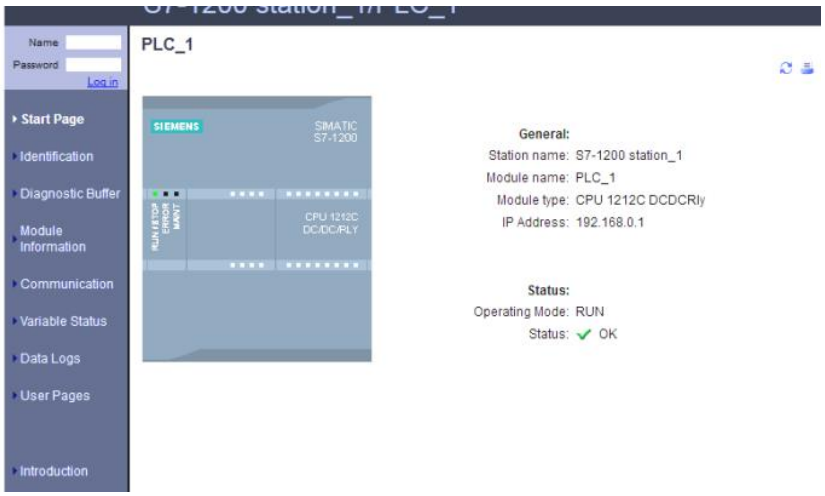


Figure 6.6 – Controller's starting page

6.3 Content of the report

- the goal of the work;
- description of the program with screenshots and photos;
- summary.

6.4 Questions for checking

1) What is the purpose of this work?
2) Explain the web page code listing.
3) Show the result of your work by changing the variables.

Used literature

1. Nathan Clark. PLC Programming Using RSLogix 5000: Understanding Ladder Logic and the Studio 5000 Platform. USA: Kindle Edition, 2020.- 184 p.

2. Frank Lamb, Advanced PLC Hardware & Programming: Hardware and Software Basics, Advanced Techniques & Allen-Bradley and Siemens Platforms— USA : Automation Consulting, LLC, 2019. — 346 p.

3. Архипов, М. В. Промышленные роботы: управление манипуляционными роботами : учебное пособие для среднего профессионального образования / М. В. Архипов, М. В. Вартанов, Р. С. Мищенко. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2022. — 170 с.

4. Creating and Using Own Web Pages for S7-1200 Application. Description March 2012 Version 1.0, Entry ID:58862931. Siemens AG 2012. – 76 с.

5. Hans Berger. Automating with SIMATIC S7-1200. – Germany: Publics Publishing,2018. – 575 с.

6. Hans Berger. Automating with SIMATIC inside TIA Portal. – Germany: Publics Publishing, 2018. – 747с.

Content

Additional plan 2022, pos.13

Kalkabekova Tomiris Zhenisbekkyzy

INDUSTRIAL CONTROLLERS IN ROBOTICS PROCESSES

Guidelines for performing laboratory work for students enrolled in the educational program 6B07109 – Control and measurement systems and instrumentations in robotics

Editor: Zhanabayeva Y.B.
Standardization specialist: Anuarbek Zh.A.

Signed to print
Circulation 50 copies
Volume 2.0 publishing sheets

Format 60x84 1/16
Typographical paper № 1
Order_____Price 1000 tg

Multiple copying Office of
Non-Profit Joint Stock Company
“Almaty University
of Power Engineering and Telecommunications named
after Gumarbek Daukeev”
050013, Almaty, Baytursynov st. 126/1