Laboratory manual

Programmable Logic Controllers Laboratory

UbiLAB

Универзитет "Св. Кирил и Методиј" во Скопје
ФАКУЛТЕТ ЗА ЕЛЕКТРОТЕХНИКА И ИНФОРМАЦИСКИ ТЕХНОЛОГИИ

Hochschule Anhalt
Anhalt University of Applied Sciences

University of Maribor

Erasmus+

NATIONAL AGENCY for European Educational Programmes and Mobility

2 Programmable Logic Controllers Laboratory

2.1 Introduction

This remote laboratory exercise is intended for a course named Programmable Logic Controllers (PLC). The laboratory equipment contains a PLC and a personal computer (PC) that would be connected within the university network. In addition, a conveyor belt process is provided which should be electrically connected with the PLC. This means, that the remote laboratory exercise would require a one-time physical access in order to complete the electrical connection. After the electrical connection has been completed, the students will be required to develop a control program for the process and test its operation afterwards.

The physical access laboratory experiment involves interaction between the students and the conveyor belt. In order for the students to observe their results, they put metallic and non-metallic objects on the conveyor belt and afterwards they observe the output from the sensors. They also interact with the push buttons or the analog potentiometer from the process. In the remote access laboratory, this interaction can be realized by involving augmented reality (AR) with virtualized objects, push buttons, potentiometer conveyed through a remote live stream of the conveyor.

For this occasion, students will be remotely connecting through Remote Desktop Protocol (RDP), facilitated through the Apache Guacamole framework, to each workstation placed within the laboratory. This will provide a remote workstation for each student in order to be able to develop a control program which would be uploaded to the PLC through the laboratory network. Afterwards, students can observe and interact with the conveyor belt by a specific web application deployed on the Moodle framework. The web application will live stream the conveyor belt from the laboratory and will concurrently utilize AR in order to enable the interaction with the virtualized objects. Each outcome of the interaction will be communicated to the PLC through the university network. The PLC program will act according to the received messages and students will be able to observe the results from their developed control program through the live video stream.

2.2 Laboratory equipment

The laboratory contains a total of eight workstations. Each workstation provides:

- Programmable logic controller: Mitsubishi FX3GE-24MT/DSS [1]
- Variable frequency drive: Mitsubishi D700-SC

- Programmable logic controller: Student made
- Voltage supply: 24VDC
- Conveyor belt process
- Personal computer with appropriate software
- Wires with appropriate safety terminal connectors
- Live stream camera

Each device provides electrical connection to its terminal through safety plug-in connectors shown on the board next to the device. Each connection is designated according to device's datasheet.

Figure 2-1 shows an example of a workstation with the mentioned equipment.

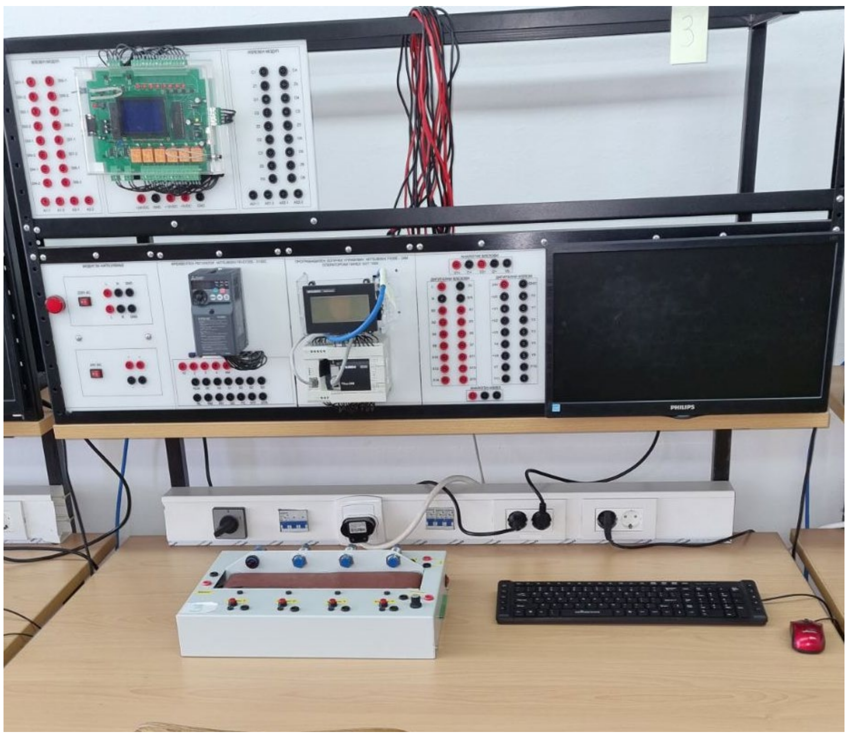


Figure 2-1. A single laboratory workstation.

The conveyor belt process is shown in Figure 2-2. The conveyor belt process includes four digital proximity sensors, of which three are inductive and one is photoelectric, four push buttons, of which one pair are momentary switches and the other are maintained switches, direct current (DC) motor for the conveyor and an analog potentiometer. It can be summarized that the plant in total requires 8 digital inputs, 1 digital output and 1 analog input from the PLC.

Each personal computer and PLC is connected to the university network. The full network topology is shown in Figure 2-3 and the actual network equipment in Figure 2-4.

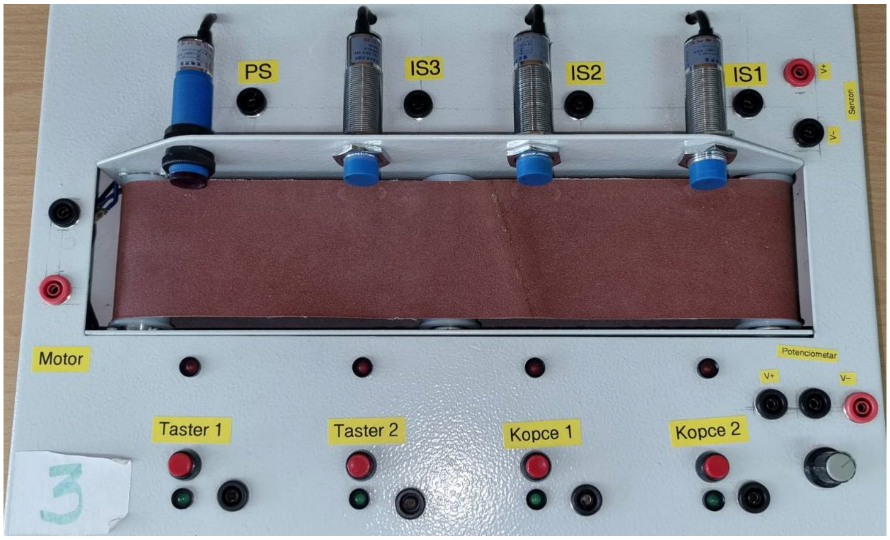


Figure 2-2. The conveyor belt process.

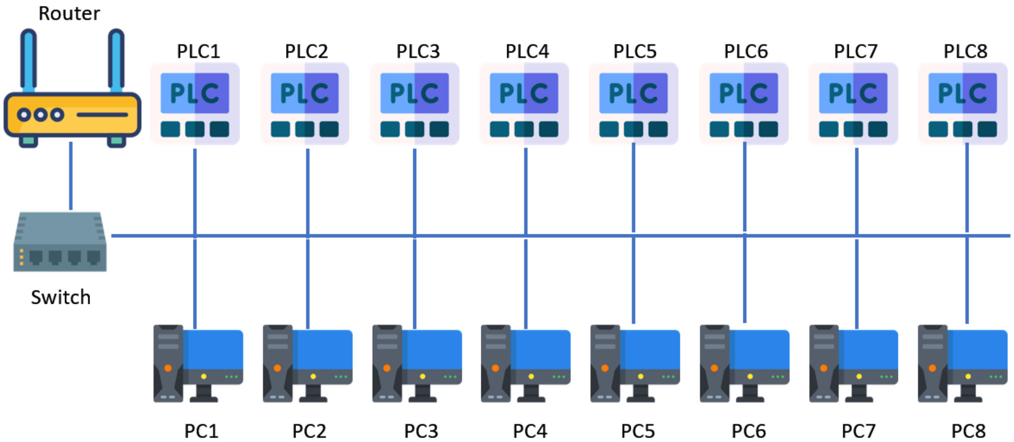


Figure 2-3. Network topology.

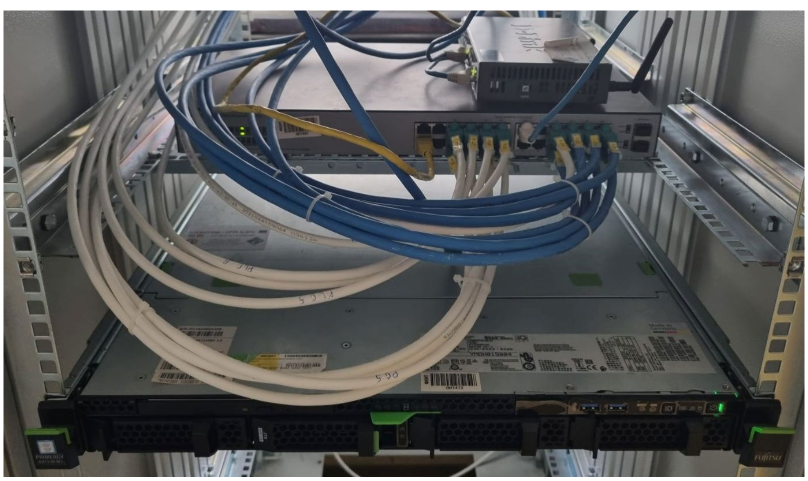


Figure 2-4. Network equipment used for the setup.

2.3 System architecture for remote access

Understanding the system architecture for remote access to the laboratory is important for executing remote laboratory experiments. The student will be required to establish two links in order to be able to access the laboratory:

- PC access
- Process access

The PC access will enable the student to remotely access the personal computer within the laboratory. This eliminates the necessity for students to install appropriate PLC software on their computers, instead they can utilize the installed software on the laboratory computers. This can be realized through RDP, and facilitated by the Apache Guacamole Framework in order to be easily integrated into the UbiLAB framework.

The Moodle LMS implementation enables the student to access the AR application for interaction with the conveyor belt process. Students will log in to the UbiLAB framework and use the appropriate course links to enter the AR application. Accordingly the AR application will provide live stream of the conveyor belt process and virtual interaction with the sensors, push buttons and potentiometer. The complete architecture is shown in Figure 2-5.

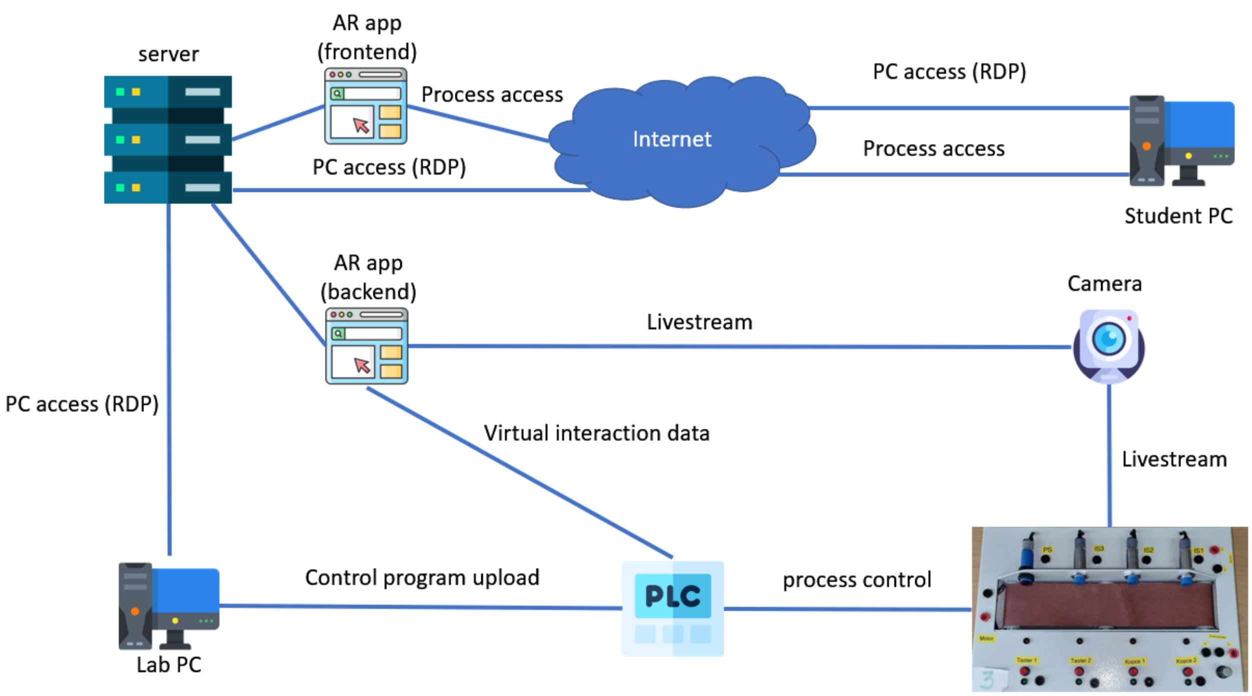


Figure 2-5. System Architecture.

In the physical access laboratory, students had to put real life objects in front of sensors, push real life buttons and interact with the potentiometer. The AR application virtualizes this interaction by adding push buttons and potentiometer. In addition, it also adds virtual objects on the live stream footage which can move with the conveyor belt accordingly. These objects can interact with virtual sensors by evaluating their mutual position through AR processing algorithms. This enables to asses if an object is in front of a sensor and therefore assesses the sensor output accordingly.

The frontend of the AR application is web based, hence easily accessible through any browser. It provides the preprocessed live stream footage along with the virtual push buttons and potentiometer. Figure 2-6 illustrates the interface of this application. Students can use the "Place Object" button to place or retrieve a virtual object on the conveyor. They can also interact with the push buttons and potentiometer. Each function of the interface widgets is shown in Table 2-1.

Table 2-1. Functions of the interface widgets.

Button name	Function
T1	Momentary switch
T2	Momentary switch
B1	Maintained switch
B2	Maintained switch
Place object	Maintained switch for placing and retrieving a virtual object from the conveyor belt
Potentiometer	Simulate a 0-10V on the analog input of the PLC

Figure 2-7 shows a real life use where the virtual object is shown in green along with appropriate sensor states (red – off; green - on).

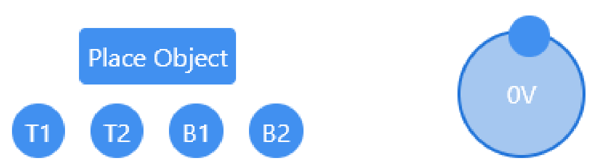


Figure 2-6. Application interface.

The backend of the same AR application retrieves live stream footage from the camera which is preprocessed using AR algorithms to add the objects and evaluate their interaction with the virtual sensors. In addition, it uses Melsec Communication protocol [2], from Mitsubishi, to communicate with the PLC. The states of the push buttons, potentiometer and virtual sensors are sent through modifying appropriate bits and registers in the PLC.

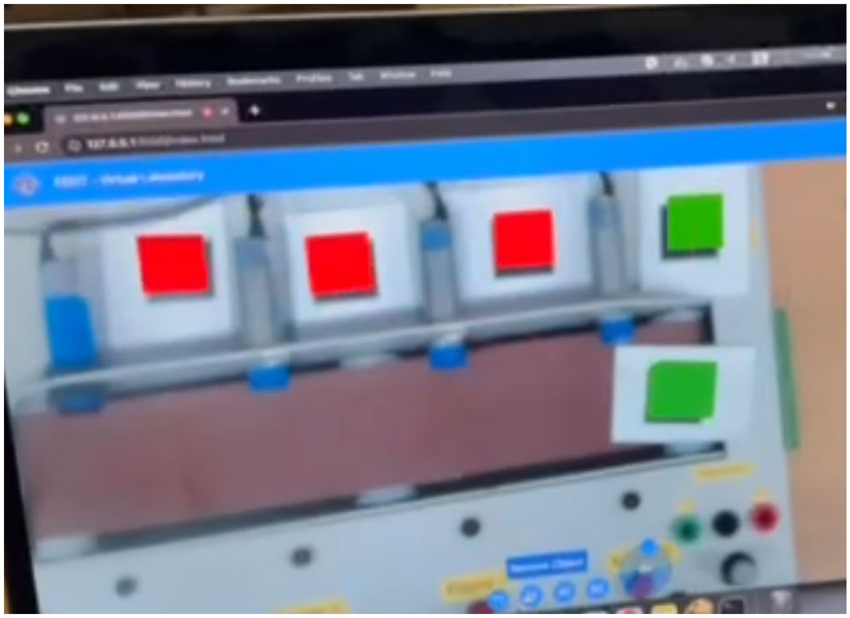


Figure 2-7. A detected virtual object on the conveyor belt.

2.4 Laboratory experiment

2.4.1 Introduction

In this laboratory experiment we are going to develop a control program for transporting objects on a conveyor belt. The objects are placed on the start of the conveyor belt and they should be transported on the other end. When an object arrives at the end, a photoelectric sensor proximity sensor detects its arrival and the conveyor belt should stop moving. As soon as the object is picked up, the conveyor belt should continue moving. The process can be started with a “start” push button and stopped with a “stop” push button. An emergency button should be utilized as well.

2.4.2 Remote access procedure

A general laboratory experiment requires the following steps for a remote access:

1. Student establishes access to laboratory PC through RDP (integrated into the framework)
2. Student logins through his UbiLAB framework account
3. Student uses the appropriate course link to open the AR application
4. Student uses the lab. PC software to create and upload control program to PLC
5. Student uses the AR application to virtually interact with the conveyor belt process in order to evaluate his control program

2.4.3 Goals of the experiment

- Develop a control program for conveyor belt process
- Transport an object from one end to another
- Use the photoelectric sensor to detect if an object has arrived at the end
- Always stop the conveyor when an object arrives at the end

- The conveyor should continue moving after object is picked up at the end
- Use start and stop push buttons for starting and stopping the process
- Use emergency button for emergency states

2.4.4 Experimental setup

2.4.4.1 Electrical connection

In order to convey this experiment the following devices will be used:

- Programmable logic controller: Mitsubishi FX3GE 24MT/DSS
- Conveyor belt process
- Voltage supply: 24VDC

In order to execute the experiment the conveyor belt, shown in Figure 2-2, needs to be connected to the PLC along with appropriate voltage supply. Each PLC and Power supply connection terminal is accessible through safety plug-in connectors mounted on a wiring board, shown in Figure 2-8.

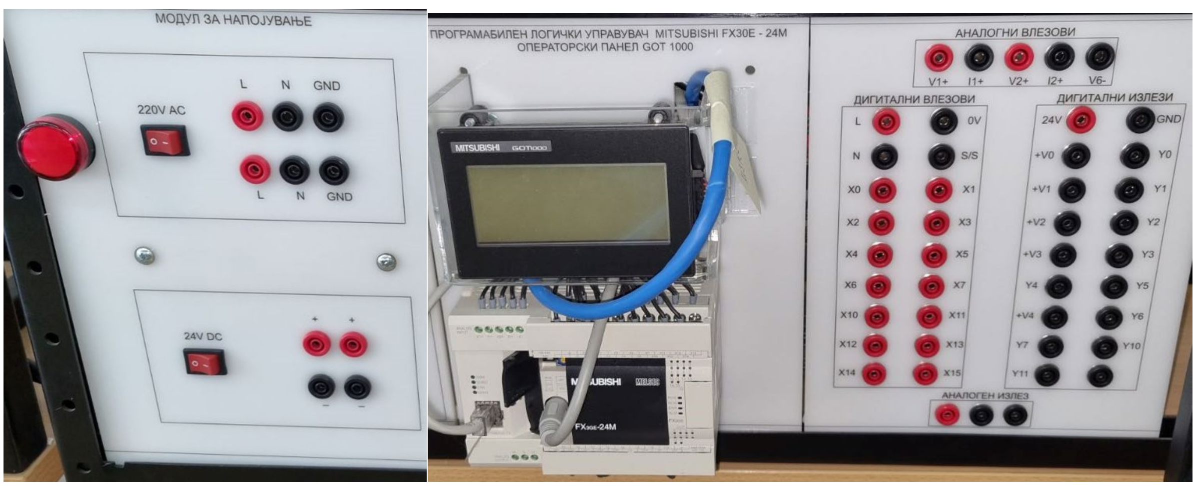


Figure 2-8. Wiring terminals for each PLC and power supply.

Using the schematics, shown in Figure 2-9, the following steps need to be executed in order to electrically connect the devices [3]:

- Connect the PLC to 24VDC power supply
- Connect the PLC digital output Y0 to one end of the DC motor and connect the other and to V- from the 12VDC power supply
- Connect the +V0 terminal, from the PLC, to the V+ terminal from the 12VDC power supply

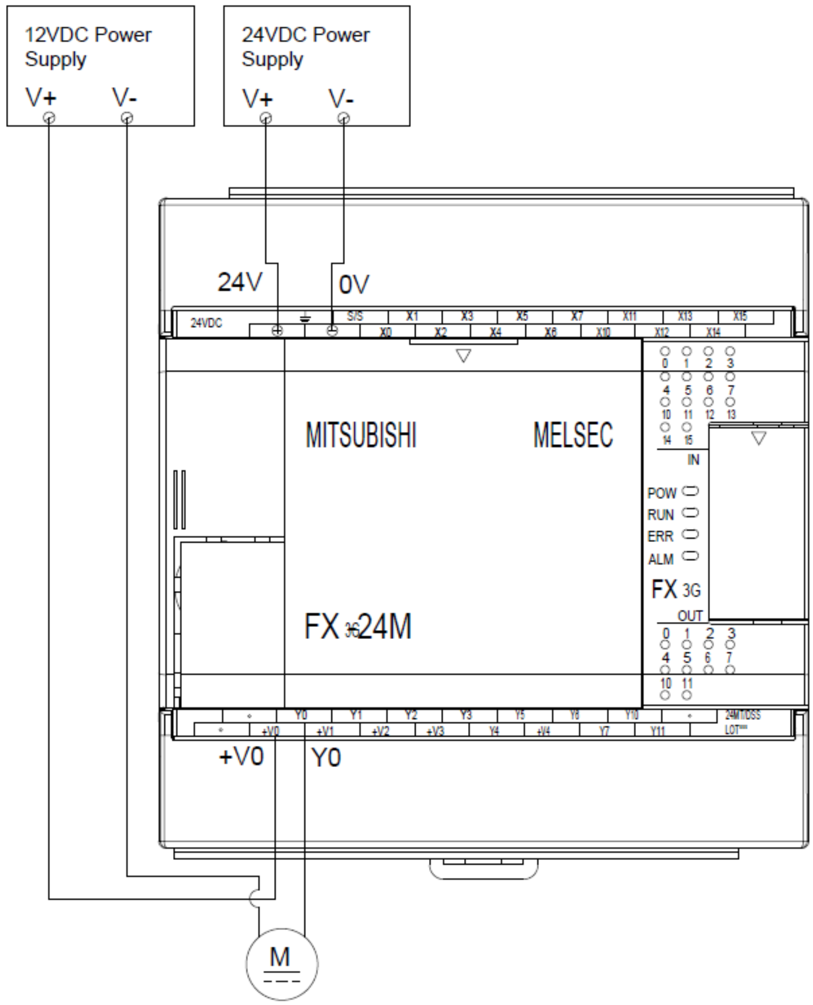
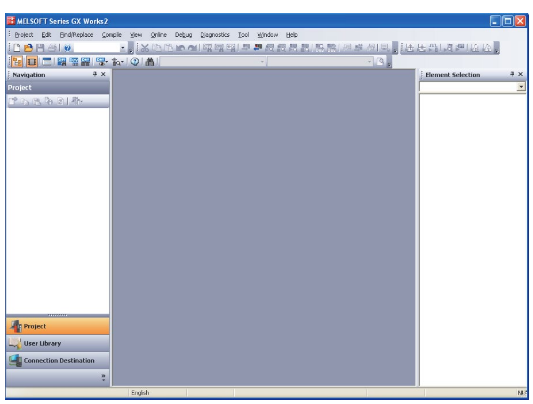


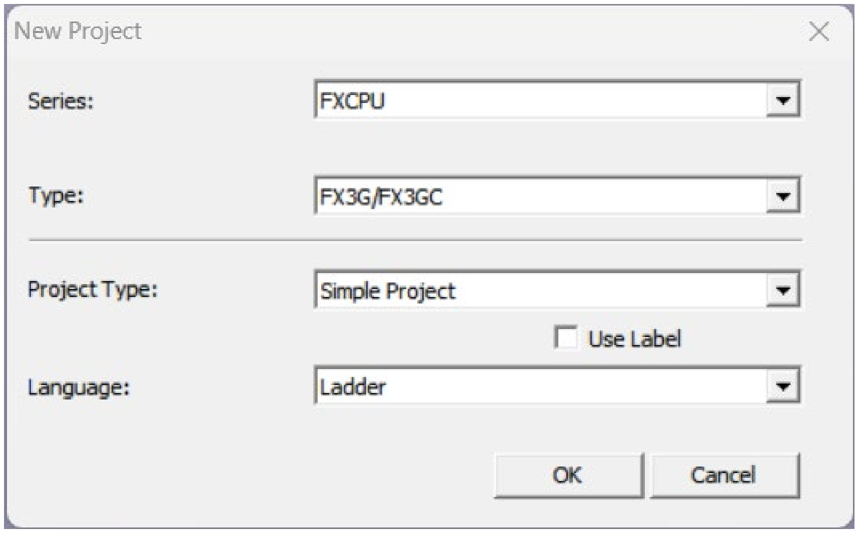
Figure 2-9. Schematic of the power supply connection to the PLC.

2.4.4.2 PLC software and program upload

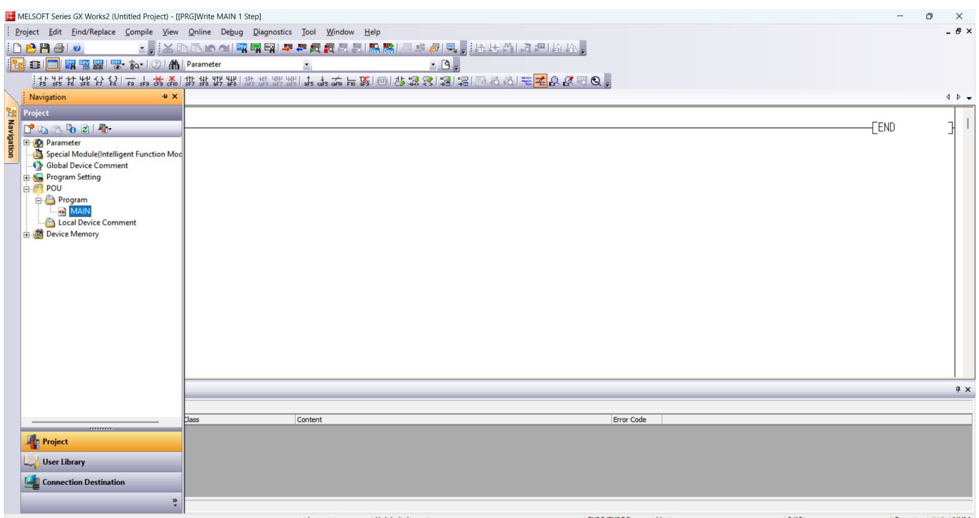
The PLC control program will by developed by using GXWorks2 software from Mitsubishi Electric. The following screen should appear when starting the software:



Create a new project using the menu bar by clicking **Project -> New.** A new window should appear allowing to choose PLC parameters. Choose the following parameters and click **OK**:



A new project should be created, displaying the following screen:



This project will use Ladder Logic programming language. This is a symbolic language which uses symbol instructions being graphically connected in order to program a control logic. Each symbol can be inserted within the program sheet divided by multiple square, each square for one instruction. Use the following symbols in order to graphically construct the ladder program.

Inserting an instruction can be realized by double clicking a square within the program sheet. A new window will appear as shown:

Choose the appropriate instruction and digital input/output address within the field.

The only sensor/actuator electrically connected to the PLC is the DC motor. Considering that we connected the DC motor to address Y0, we will use this address within our control program. Any additonal sensor used within the control program is virtualized therefore we should use internal addresses for each virtualized sensor. The addresses should be used according to Table 2-2.

Table 2-2. Addresses and tags for the used devices.

Device	Tag	Address
Inductive sensor 1	IS1	M0
Inductive sensor 2	IS2	M1
Inductive sensor 3	IS3	M2
Photoelectric sensor	PS	M3
Momentary switch 1	T1	M4
Momentary switch 2	T2	M5
Maintained switch 1	K1	M6
Maintained switch 2	K2	M7
Potentiometer	Potenciomear	D0

Use the push buttons T1, T2 and K1 as start, stop and emergency stop (NO) push button.

A control program solution is shown in Figure 2-10 [5]. We use the first rung in order to define the work state of an auxilary bit with address M10. The conveyor belt process should work if this bit is logic 1 and otherwise if logic 0. The state of this bit depends on the ladder instruction states for the start, stop and emergency push button. The second rung defines whether or not the conveyor belt should be moving according to the work bit and photoelectric sensor state. Use the menu bar to compile the program by clicking **Compile->Build.**

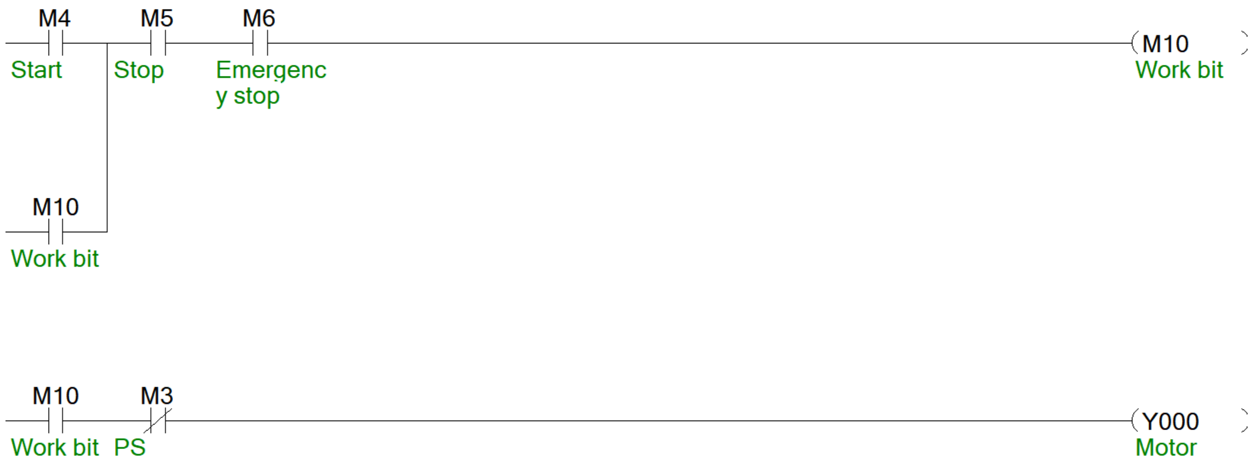
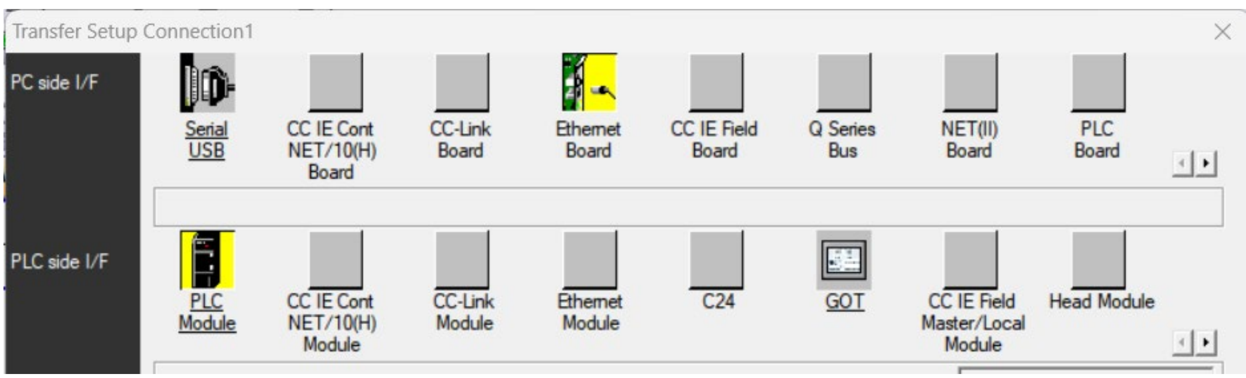


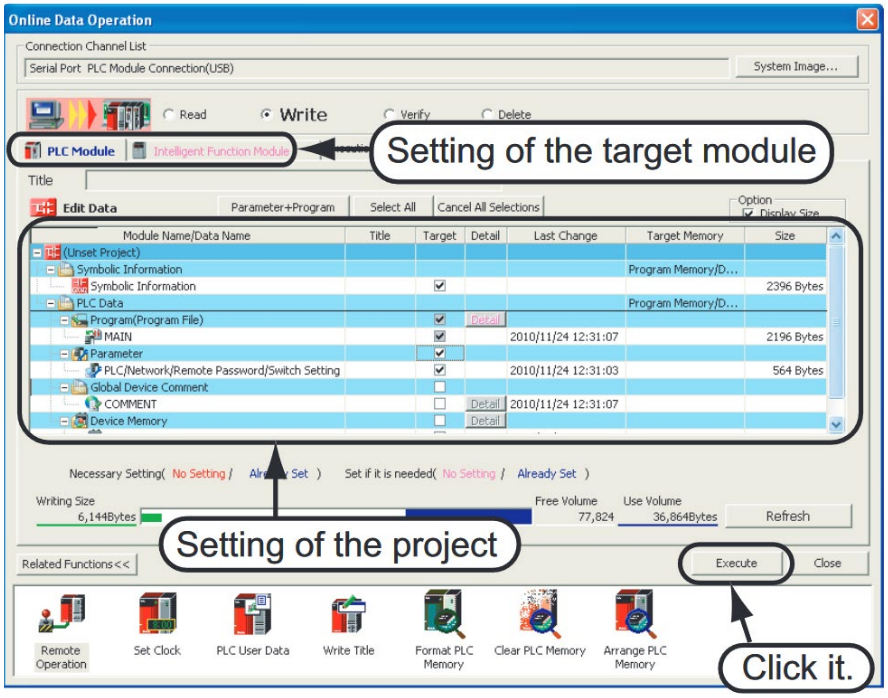
Figure 2-10. Control program solution.

Using the navigation panel go **Connection Destination** and double click **Connection1.** A new window should appear allowing to establish connection with the PLC. Choose **Ethernet Board** for PC side communication and **PLC Module** for the PLC side.



Double click the **PLC Module** to open setup the communication parameters for the PLC. Choose the appropriate IP address for your PLC and click **OK.** Test the connection by clicking **Connection Test** and click **OK** after confirming that the connection was successful.

Now that we have established succesful connection to the PLC we may proceed with uploading the developed control program. Use the menu bar to open the programming windows by clicking **Online->Write to PLC.** Choose the MAIN program file as the only file to upload to the PLC and click **Execute**.



2.4.4.3 AR application

In order to test the uploaded program the student may interact with the process through the AR application by using the interface buttons shown in Figure 2-6. Figure 2-11 illustrates the interface of the AR application.

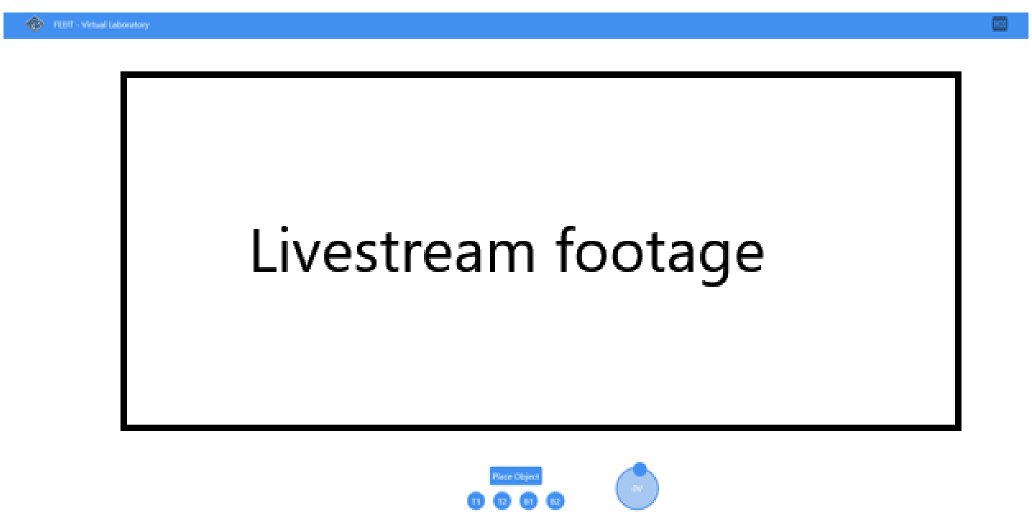


Figure 2-11. AR application interface.

2.4.5 Experimental results

Use the AR application in order to test the program workflow.

- Use the **Place object** to place a virtual object on the conveyor belt. A green square should appear on the conveyor belt.
- Click the **T1** to start the process. If the program upload was successful, the virtual green object should start moving. Observe the triggering of each sensor, as the object passes in front of them.
- Each sensor designates its state by a green or red square right next to it, as shown in Figure 2-7. If the photoelectric sensor is successfully triggered, the conveyor belt should stop.
- Use the **Place object** button to remove the object. The conveyor belt should start moving again

Figure 2-12 shows a series of images that present the workflow of the AR app.

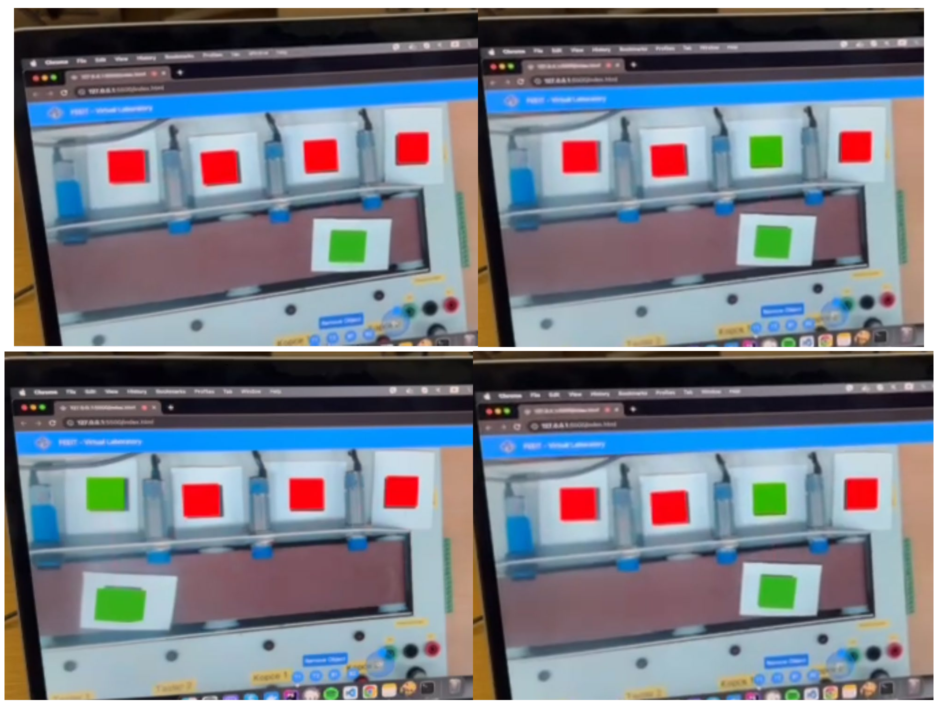


Figure 2-12. AR application workflow.

2.5 Conclusion

A remote access PLC laboratory was presented utilizing AR technology. It was shown that necessary interaction for evaluation PLC programs can be executed remotely through AR virtualized objects and sensors. Furthermore, this was demonstrated through a simple laboratory experiment. A control program was developed and its operation was tested through an AR application enabling remote access to the laboratory.

2.6 References

[1] Mitsubishi Electric, Product Information, FX3GE MELSEC Compact PLC, All-in-one-standard including analog and network features, EBG 243 EN

[2] Mitsubishi Electric, Programmable Controller MELSEC-F, FX3U-ENET User's Manual

[3] Mitsubishi Electric, Programmable Controller MELSEC-F, FX3GE Series Programmable Controller, Hardware Manual, JY997D49401J

[4] Mitsubishi Electric, Engineering Software, GX Works2 Beginner's Manual (Simple Project)

[5] Mitsubishi Electric, Programmable Controllers MELSEC-F, Programming Manual, The FX Series of Programmable Controller

2.7 Acknowledgements

- https://www.flaticon.com/free-icons/plctitle - created by Freepik – Flaticon
- https://www.flaticon.com/free-icons/network-switch - created by Chattapat - Flaticon
- https://www.flaticon.com/free-icons/server - created by Freepik – Flaticon
- https://www.flaticon.com/free-icons/router - created by Vectors Tank – Flaticon
- https://www.flaticon.com/free-icons/server - created by Freepik – Flaticon
- https://www.flaticon.com/free-icons/web-camera - created by Freepik – Flaticon
- https://www.flaticon.com/free-icons/web-pagetitle - created by Talha Dogar – Flaticon