http://doi.org/10.35784/iapgos.3217

received: 31.10.2022 | revised: 10.12.2022 | accepted: 15.12.2022 | available online: 30.12.2022

[RETRACTED]: SYSTEM SOFTWARE AT LABORATORY STAND „LIGHTS CONTROL ON THE BASIS OF PROGRAMMABLE LOGIC CONROLLER”

Laura Yesmakhanova, Raushan Dzhanuzakova, Bauyrzhan Myrkalykov
Taraz Regional University named after Dulati M.K., Taraz, Kazakhstan

Abstract. *The article is concerned with the issues of practical utilization of controllers for automation in various technical fields as example Mitsubishi controllers of MELSEC FX1N and FX3U series. Any task, which requires the utilization of electrical control devices, is easy to solve by means of usage of programmable logic controllers. By using software with controllers programming package GX Developer, user has the possibility to program the controller or input changes into existing program. The Laboratory Stand was developed for effective learning of controller programming in LD language.*

Keywords: controller, program, control, relay

[WYCOFANY]: OPROGRAMOWANIE SYSTEMOWE NA STANOWISKU LABORATORYJNYM „STEROWANIE OŚWIETLENIEM W OPARCIU O PROGRAMOWALNY STEROWNIK LOGICZNY”

Streszczenie. *Artykuł dotyczy zagadnień praktycznego wykorzystania sterowników do automatyzacji w różnych dziedzinach technicznych, na przykład sterowników Mitsubishi serii MELSEC FX1N i FX3U. Każde zadanie, które wymaga wykorzystania elektrycznych urządzeń sterujących, można łatwo rozwiązać za pomocą programowalnych sterowników logicznych. Korzystając z oprogramowania z pakietem programowania sterowników GX Developer, użytkownik ma możliwość zaprogramowania sterownika lub wprowadzenia zmian do istniejącego programu. Stanowisko laboratoryjne zostało opracowane z myślą o efektywnej nauce programowania sterowników w języku LD.*

Słowa kluczowe: sterownik, program, sterowanie, przekaźnik

The article has been retracted due to a breach of ethical standards. The editorial board received a report that this work infringes on the copyrights of others. An investigation confirmed the validity of the report, resulting in the decision to retract the work. The evidence in the case is in the possession of the editorial board. The editorial board of the journal also declares that it has followed all procedures for checking the article for unethical practices before its publication in 2022.

Artykuł został wycofany z powodu naruszenia zasad etyki. Redakcja otrzymała zgłoszenie, że niniejsza praca narusza prawa autorskie innych osób. Przeprowadzone postępowanie wyjaśniające potwierdziło prawdziwość zgłoszenia, w wyniku czego podjęto decyzję o wycofaniu pracy. Materiał dowodowy w sprawie jest w posiadaniu redakcji. Redakcja czasopisma oświadcza jednocześnie, że z jej strony zostały zachowane wszelkie procedury sprawdzające artykuł pod kątem nieetycznych praktyk.

http://doi.org/10.35784/iapgos.3217

received: 31.10.2022 | revised: 10.12.2022 | accepted: 15.12.2022 | available online: 30.12.2022

SYSTEM SOFTWARE AT LABORATORY STAND „LIGHTS CONTROL ON THE BASIS OF PROGRAMMABLE LOGIC CONROLLER”

Laura Yesmakhanova, Raushan Dzhanuzakova, Bauyrzhan Myrkalykov
Taraz Regional University named after Dulati M.K., Taraz, Kazakhstan

Abstract. *The article is concerned with the issues of practical utilization of controllers for automation in various technical fields as example Mitsubishi controllers of MELSEC FX1N and FX3U series. Any task, which requires the utilization of electrical control devices, is easy to solve by means of usage of programmable logic controllers. By using software with controllers programming package GX Developer, user has the possibility to program the controller or input changes into existing program. The Laboratory Stand was developed for effective learning of controller programming in LD language.*

Keywords: controller, program, control, relay

OPROGRAMOWANIE SYSTEMOWE NA STANOWISKU LABORATORYJNYM „STEROWANIE OŚWIETLENIEM W OPARCIU O PROGRAMOWALNY STEROWNIK LOGICZNY”

Streszczenie. *Artykuł dotyczy zagadnień praktycznego wykorzystania sterowników do automatyzacji w różnych dziedzinach technicznych, na przykład sterowników Mitsubishi serii MELSEC FX1N i FX3U. Każde zadanie, które wymaga wykorzystania elektrycznych urządzeń sterujących, można łatwo rozwiązać za pomocą programowalnych sterowników logicznych. Korzystając z oprogramowania z pakietem programowania sterowników GX Developer, użytkownik ma możliwość zaprogramowania sterownika lub wprowadzenia zmian do istniejącego programu. Stanowisko laboratoryjne zostało opracowane z myślą o efektywnej nauce programowania sterowników w języku LD.*

Słowa kluczowe: sterownik, program, sterowanie, przekaźnik

Introduction

The contemporary production is based on high technologies that are designed to provide speed, scale, reliability, safety and high quality of task performance. Long-standing need for flexible process control, efficient use of production capacity, remote control in real time determined the need to move from bulky Ladder Diagram to reprogrammable logic controllers. Contemporary controllers swiftly process data and automatically control the processes. The ability to change the program allows quickly change the technological process depending on the current task.

Programmable logic controller (PLC) is a specialized microprocessor device with embedded hardware and software that is utilized to perform the functions of controlling process equipment. PLC is a device accessible for programming by a non-specialist in IT and designed to control sequential logical processes in an industrial environment in real time. The PLC repeatedly scans the inputs to which switches, sensors, etc. are connected, and depending on their state (”on” – 1, ”off” – 0), turns the outputs on and off, and hence the actuators connected to the outputs. The functional diagram of the control system (CS) based on the controller is shown in Figure 1. By utilizing the software, the user has possibility to program the controller or input changes to an existing program [1].

The programmable logic controller mainly consists of a central processing unit (CPU), a memory area, and I/O signal processing functions (i.e., inputs and outputs). Contingently, such a controller can be called the essential or base unit (module). It can be considered the PLC as hundreds or thousands of individual relays, counters, timers, and memory. All these counters, timers are simulated by the CPU and carry out the logic of work, according to the embedded program. The block diagram of the controller is shown in the figure 2.

- INPUTS provide communication with external devices. Physically exist and receive signals from switches, sensors, etc. There are analog and digital inputs designed to work with analog and digital signals, respectively.
- The CPU is the ”brain” of the PLC, carrying out the logic of the system. This is a processor that processes program commands and controls all internal elements of the controller: inputs, outputs, counters, timers, internal relays, registers, etc. In figure 2, the counters, timers, and internal relays are not shown separately; they are part of the CPU chip. I.e. each controller has a fixed set of such elements, which are given in the specification.
- INTERNAL RELAYS (MARKERS) are designed to ensure the operation of the program, because are a kind of information storage units. Along with ordinary markers, there are also service markers that carry a special semantic and functional load (for example, setting an enable flag to start high-speed counters). The purpose of each specific service marker is given in the documentation for the controller.

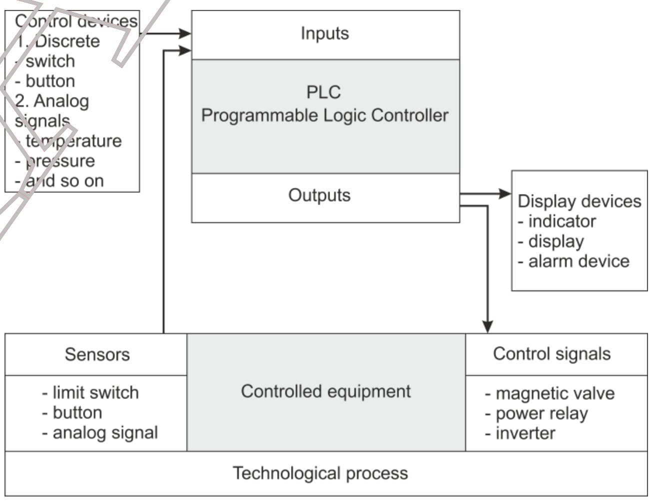


Fig. 1. Functional diagram CS on PLC base

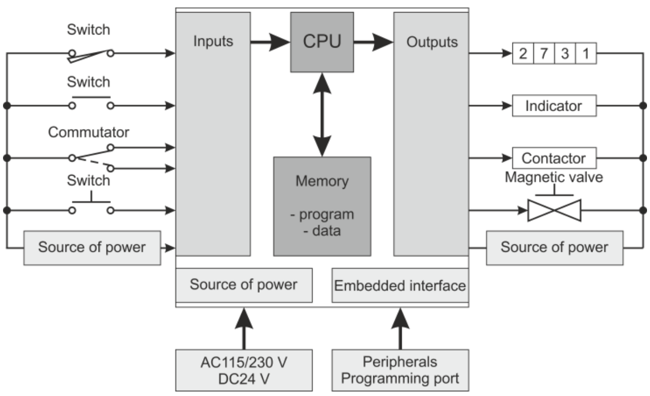


Fig. 2. Block diagram of the controller

- COUNTERS are designed for various types of counting. Separately allocate high-speed counters. Usually, there are restrictions on the count rate and the value to which the count is kept, for which it is necessary to refer to the documentation of a particular controller.
- TIMERS are designed, as a rule, to set the on / off delay time, etc. They differ mainly in the accuracy of timing and, as a result, in their purpose.
- MEMORY – the controller has a certain amount of memory, which may have a different organization in different controllers. Most commonly, memory is divided into a work area (RAM), where the program is loaded directly during the operation of the controller, and a data area (EEPROM, MMC, etc.), where the program and various data are stored. Often the size of the work area is measured in kilobytes, and the size of the data area is measured in the number of program steps.
- BUILT-IN INTERFACE provides connection of the PLC to a computer or programmer for data exchange, including for reprogramming the controller. Basically, these are RS-232C (COM-port), RS-422, RS-485, etc.
- OUTPUTS provide communication with external devices, i.e. provide switching on/off of actuators. There are two versions: relay, semiconductor (transistor and triac). There are analog and digital outputs designed to work with digital and analog signals, respectively.
- POWER SUPPLY is designed to ensure the operation of the controller. External power supplies can be used, both DC +12/24 V and AC – ~110/220 V. Many controllers have built-in service power supplies (usually +12/24V) that are used to power sensors or other devices connected to the controller to simplify input and output circuits [1].

1. Conditions and methods of research

Recently, there has been a tendency to expand the functionality of controllers via the implementation of built-in PID controllers, real-time clocks, networking controllers and using the ability to connect expansion units. In any case, the structure of the controller remains unchanged, and the choice of model is determined only by the requirements of the process, and a wide range of models allows you to choose a controller with an optimal price / performance ratio.

To understand the operation of the controller, figure 3 shows the algorithm of its operation.

During operation, the PLC continuously scans the current state of the inputs X1, X2…Xn and, in accordance with the requirements of the production process, changes the state of the outputs Y1, Y2…Yn (on/off). This cycle can be divided into four main steps.

The first step is to initialize the system. It must be remembered that in the event of a power failure or when the controller is turned off, the system must return to its original state. The importance of this part of the program code should not be underestimated; otherwise it can lead to failures and equipment breakdowns.

The second step is to check the current state of the inputs. The PLC checks the current state of the inputs and, depending on their state ("on" or "off"), performs sequential actions specified in the program. The state of any of the inputs is stored in memory (in the data area) and can later be used when processing the third step of the program [3].

The third step is execution of the program. We will assume that during the process, the input (X1) switched from "off" to "on", and in accordance with the process, we need to change the current state of the output (Y1) from "off" to "on". Since the CPU scanned the current state of all inputs and stores their current state in memory, the choice of the next action is determined only by operation process.

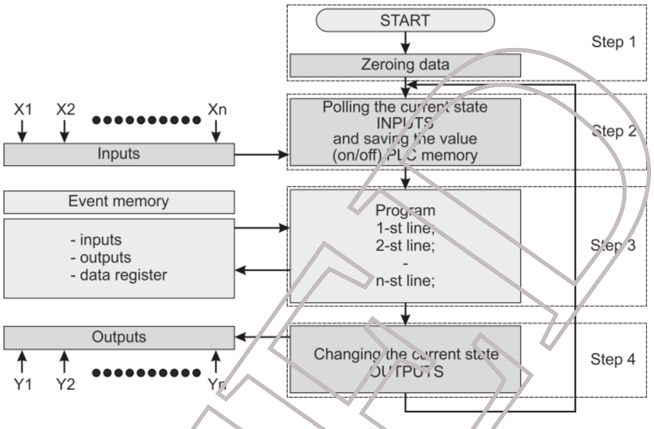


Fig. 3. Algorithmic diagram of controller operation

Step four is to change the current state of the output. The PLC changes the current state of the outputs depending on which inputs are disabled and which are enabled, based on the algorithm of the program stored in the memory, which was processed in the third step. I.e., the controller physically switched the output (Y1) and the actuators turned on: a light bulb, an engine, etc. This is followed by a return to the second step.

While using a DC power supply, it is necessary to connect the "+" terminal of the power supply to the "+" terminal of the controller, and the "-" terminal of the power supply to the "-" terminal of the controller. No other connection is allowed.

When using AC power supply:

- the phase line must be connected to the L terminal and the 0 line to the N terminal. To avoid electric shock, do not connect the L phase line to the N terminal,
- ground connection is required. Ground resistance must be $R < 100\ \Omega$.

An example of connecting a power supply is shown in Fig. 4.

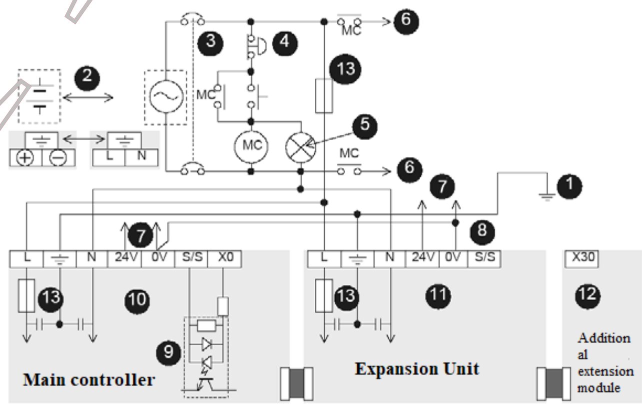


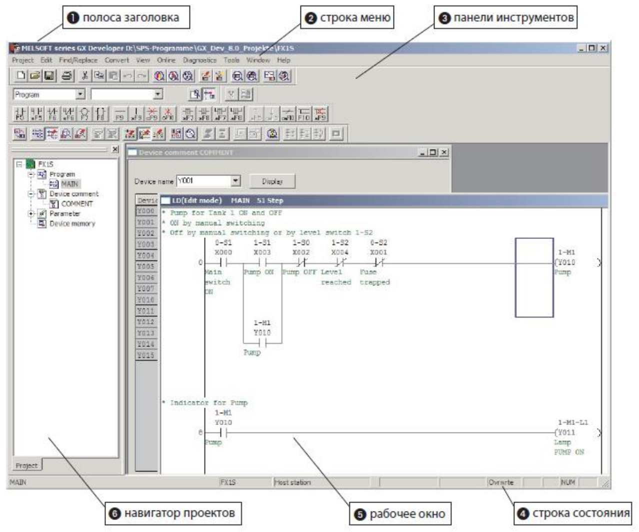
Fig. 4. Example of connecting a power supply to PLC: 1 – grounding: class 3; 2 – power supply; 3 – protective device; 4 – emergency stop; 5 – power indicator; 6 – load supply; 7 – built-in power supply; 8 – switch for the type of input switching; 9 – optocoupler; 10 – base block; 11 – extension block; 12 – extension module (ex. analog input); 13 – fuse (3 A)

If the system uses the built-in power supply of both the base unit and the expansion unit, then the "0" terminals must be connected. At the same time, DO NOT CONNECT the "24 V" terminals of the base unit and the expansion unit to each other; NEVER connect an external power supply to the "24V" terminals.

2. Launch GX Developer, shell program overview

After installing the GX Developer programming environment on your computer, launch it by double-clicking the button in the START menu > Programs > MELSEC Application > GX Developer.

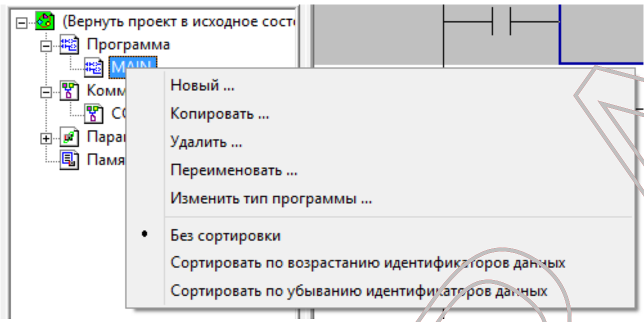
After launching GX Developed, the user software interface appears.



On this figure, for better visibility, the project is already open. After launching GX Developer, you only need to open an existing project or create a new project.

The MELSEC controllers of the FX family can only process one program (default name: MAIN).

To rename the MAIN program, click MAIN and press the right mouse button. Then select the action you want to take.

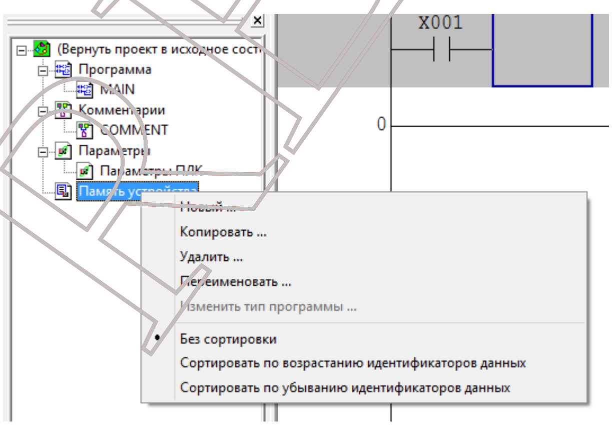


Comments can be added to each controller operand (inputs, outputs, markers, etc.), which can be seen on the screen in the program.

By opening a file with an operand comment in the project navigator, you can enter or edit comments. But input of comments is also possible directly during programming.

Double-clicking **PLC Parameters** opens a dialog where you can set all the parameters necessary for the controller operation. Controller parameters are transferred to the CPU along with the program [4].

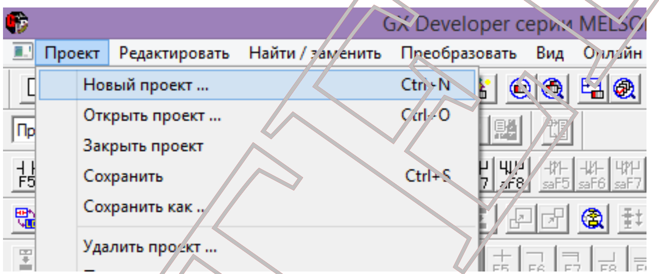
In file in the directory **Device Memory**, you can already enter a value for any data register (D) of the CPU during programming. After loading this file into the controller together with the program, these initial values are taken into account already at the first start of the program.



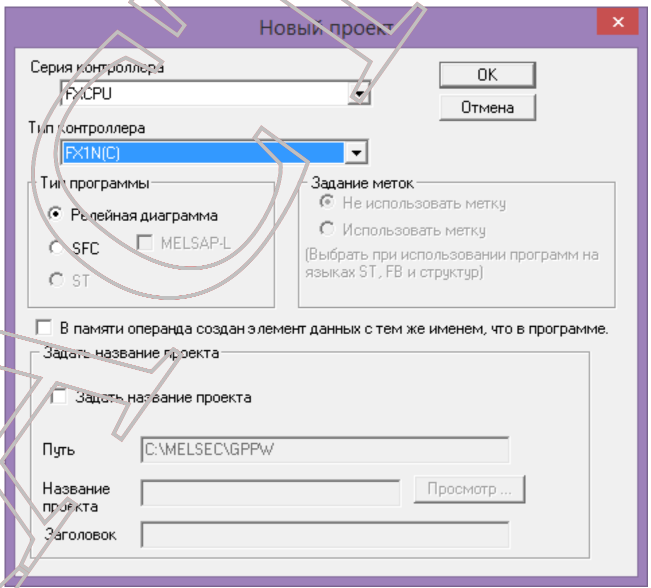
To do this, click on **Device Memory** and press the right mouse button. Then click on **New**... and name the file.

There are volatile and non-volatile (fixed) areas in the memory of operands of the central processor. If the data must be saved even when the controller is powered off until it is turned on again, they must be entered in fixed areas.

To create a new project, click on the menu Project on the line New project or click on the toolbar on this screen button .



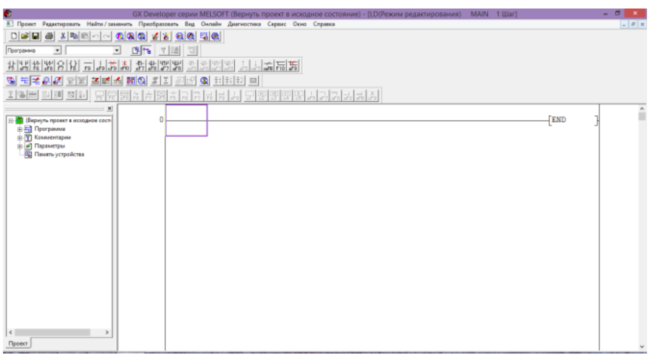
After that, the following dialog box appears:



The **Controller series** entry field in the GX Developer FX software is pre-populated with the FXCPU series, since only MELSEC controllers of the FX family can be programmed with this software.

The base unit type is selected in the **Controller Type** field. Click the arrow at the right edge of the input field. A selection of all available types of processors of the FX family appears. If you click on the controller symbol, this selection is confirmed in the input field. For laboratory stand, choose FX1N(C) or FX3U(C). Next, select the type of program **Ladder Diagram** or **SFC**, set the name of the project.

After clicking OK, a new, as yet empty, MAIN program appears in the GX Developer work window.



The stand ”Lighting control based on a programmable logic controller” consists of a differential automatic switch, a voltage and current relay; 24V power supply; PLC FX3U 24MR; control buttons; signal lamps.

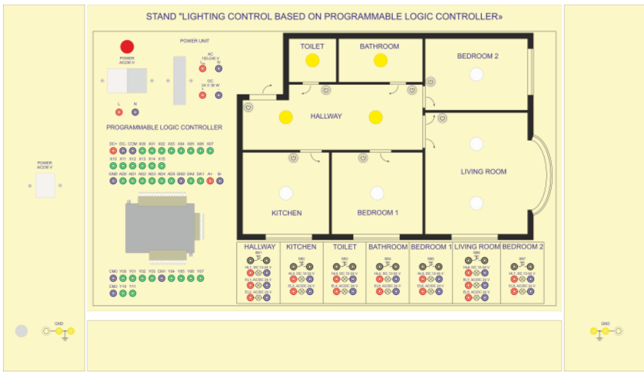


Fig. 5. General view of Stand ”Lighting control based on a programmable logic controller”

The front panel and body of the stand contain: – an AC power source, which includes a two-pole circuit breaker, a differential current circuit breaker of the AVDT32 series and a voltage relay; – a power source – UNO-PS / 1AC / 24DC / 24W; – control buttons; – programmable logic controller FX3U 24MRMitsubishi; – contact sockets. The stand also includes a set of connecting wires.

The GX Developer FX program is required to work with the stand. Connection to the PLC is made using an RS232 cable.

To create a control circuit for lighting operation using control buttons on the stand ”Lighting control based on a programmable logic controller”, assemble a control circuit with status indicators for PLC inputs and outputs, buttons that set lighting control signals.

The following must be done to write and run the program: 1. Create a new project, specifying the type of PLC on which the work will be performed. 2. Create a ladder diagram for this task. 3. Convert ladder diagram. 4. Establish communication with the PLC FX3U24MR at the stand ”Lighting control based on a programmable logic controller”. For that reason connect the RS-232 cable with driver installation, set transmission parameters. 5. Write the diagram to the PLC. 6. Collect the diagram at the stand. 7. Enable monitor mode in GX Developer FX. 8. Put the PLC in ”RUN” mode.

3. Example

On the example of the program ”Lighting control based on a programmable logic controller” let's consider the first main elements of the toolbar. Switching on will be done using the SB1 button, and switching off using the SB2 button. Light bulb EL1 will be an indicator of work.

The ladder diagram of this circuit is shown in figure 6.

On the diagram X000 – controller input from the SB1 button; X001 – controller input from the SB2 button; Y000 – controller output to the EL1 light bulb.

This ladder diagram utilizes a make and break contact, a coil, a shunt contact. The principle of operation is based on closing and opening the circuit.

Further you need to collect the following diagram on the stand – Fig. 7.

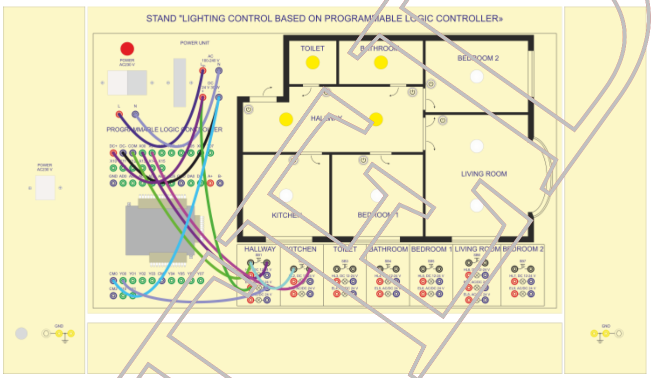


Fig. 7. Connecting patterns for ladder diagram at figure 6

The corridor lamps are turned on and off according to the following algorithm. One press of the SB1 button (input X000) turns on the output Y000 (lamp EL1), the next press turns off the output Y000 (lamp EL1). Double pressing the button SB1, turns on the output Y001 (lamp EL2), the next press turns off the output Y001 (lamp EL2). The algorithm also contains the function of simultaneously switching off two outputs Y000 and Y001. For that reason, press and hold the SB1 button for 3 seconds.

Lamp EL1 is connected to output Y000, and EL2 is connected to output Y001.

SB1 HL1 button light is directly connected to DC24V power supply.

The ladder diagram of this circuit is shown at figure 8.

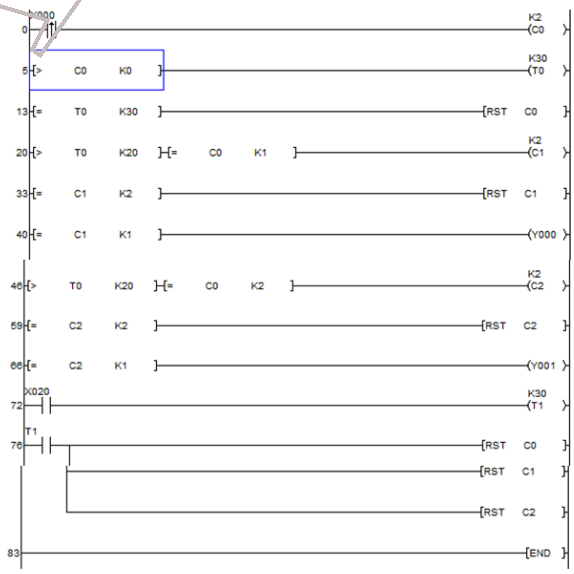


Fig. 8. Ladder diagram for corridor lighting control with one button

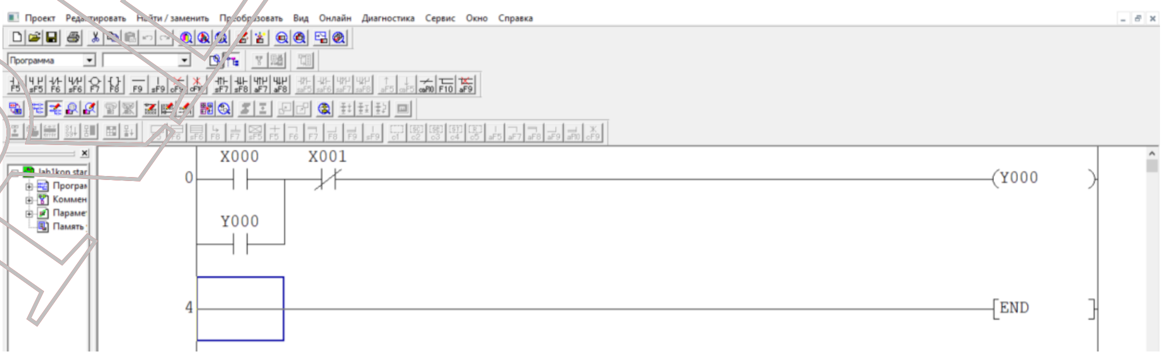


Fig. 6. The ladder diagram of program „Lighting control based on a programmable logic controller”

On the diagram X000 – input to the controller from the button SB1; Y000 – output from the controller to the bulb EL1; Y001 – output from the controller to the bulb EL2.

This ladder diagram utilizes 3 counters, 2 timers, LDP command (rising edge control), comparison functions, etc.

Further you need to collect the following diagram on the stand – Fig. 9.

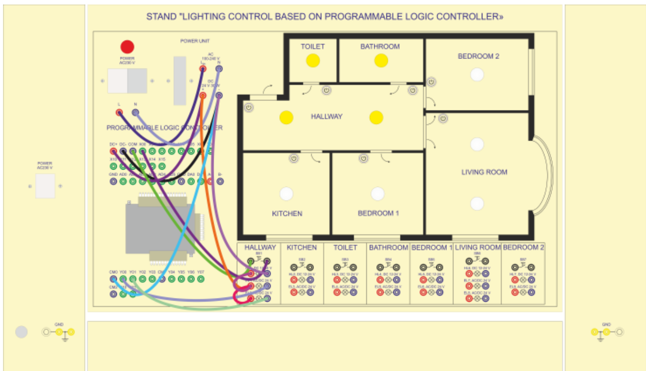


Fig. 9. Connecting patterns for ladder diagram at figure 8

The PLC specifications give typical operating cycle times. While measuring the user program must contain 1K (1024) simple logic instructions (in the IEC 1131-3 instruction list (IL) language). PLCs today have typical runtimes measured in units of milliseconds or less. Events that require a quick response are allocated to separate tasks, the priority and execution period of which can be changed.

4. Summary

So this work considers the issues of practical use of controllers for automation in various fields of technology on the example of Mitsubishi controllers of the MELSEC FX1N and FX3U series, as typical representatives of low-priced and widely used controllers.

Reference

[1] Introduction to PLC: What is a Programmable Logic Controller https://www.compel.ru/lib/95591
[2] Mitsubishi Melsec https://ru.wikipedia.org/wiki/Mitsubishi_Melsec
[3] Programmable Controller MELSEC https://ru3a.mitsubishielectric.com/fa/ru/products/cnt/plc/
[4] Programmable Logic Controllers Mitsubishi Electric. http://esspb.ru/dev.html
[5] GX Developer FX. Programming and Documentation System: Beginner's Guide. 2007. http://www.mitsubishi-automation.ru
[6] MELSEC FX Family. Programmable Controllers. Analog Input Modules, Analog Output Modules, Combined Analog I/O Modules, Temperature Recording Modules: Instruction Manual. 2012. http://www.mitsubishi-automation.ru

Ph.D. Laura Yesmakhanova
e-mail: laura060780@mail.ru

Graduate of the doctoral program of the Lublin University of Technology. Currently, he works as an associate professor at Taraz Regional University at the Department of Automation and Telecommunications. He is the author and co-author of several scientific publications in the field of the use of optoelectronic components in automated control systems and fiber-optic communication lines.

http://orcid.org/0000-0002-3308-9676

Ph.D. Raushan Dzhanuzakova
e-mail: rdd-taraz@mail.ru

Candidate of technical sciences in the Republic of Kazakhstan. He has copyrights for inventions, an innovative patent „Electromagnetic method for measuring stresses in ferromagnetic products"; copyrights for inventions, an innovative patent „Atopological method for measuring stresses in elements of products". He is the author and co-author of several scientific publications in the field of the use of optoelectronic components in automated control systems and fiber-optic communication lines.

http://orcid.org/0000-0001-7168-2676

Ph.D. Bauyrzhan Myrkalykov
e-mail: Baur-phd@mail.ru

Associate Director office of the Registra. He completed scientific internships at Lazar University in Warsaw (Poland) and at the University of Food Technologies in Plovdiv (Bulgaria). He is the author of several scientific publications in the field of innovative technologies. Studies the functions related to the organization of the process of high-quality and timely formation and effective use of the fund of modern control and measuring materials, based on the continuous improvement of methods and technologies of their development and expertise.

http://orcid.org/0000-0001-7224-4168