INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2022, VOL. 68, NO. 4, PP. 709-714
Manuscript received November 5, 2022; revised November 2022. DOI: 10.24425/ijet.2022.141289

Fast Designing Ladder Diagram of Programmable Logic Controller for a Technological Process

Adam Szcześniak, and Zbigniew Szcześniak

***Abstract*—The article presents developed method and general principles of creating ladder diagrams, which are commonly used for systems with programmable logic controllers (PLC). Ladder diagrams are created for sequential control systems of technological processes, which are described by a connection pattern, time diagrams of the executive elements` operation. The executive elements are double-acting pneumatic or hydraulic actuators controlled by bistable electrovalves. A method of designing sequential systems enabling the creation of a ladder electro-pneumatic system is presented. The ladder diagram consists of two parts. One is responsible for controlling the valve coils, the other for the implementation of the memory block. The signals that control the transition to the next state are the signals described on the boundaries of the graph division. The synthesis of control systems and their verification was carried out using the computer aided program FluidSim by Festo.**

***Keywords*—synthesis and verification of sequential systems; ladder diagrams; PLC programming**

I. Introduction

IN designing, an important issue is the analysis and synthesis of the schematic diagram of the device. In the synthesis of digital [1,2,3] and analog [4,5,6,7] electrical systems, a mathematical description is required, which is often complicated. In order to meet the expectations of designers regarding the minimization of the mathematical apparatus in the analysis of circuits, a method of the synthesis of sequential circuits has been presented. The presented material is a continuation of the research problems announced in the article [8], which presents the synthesis of a sequential system with the use of logical elements.

The article presents a quick design of a ladder diagram for a PLC controller on the example of a selected sequence of executive elements of a technological process. The work cycle of the automatic machine is presented with the connection pattern and cyclogram, which show the states of the actuators. On their basis, the automaton graph was created. The following information is obtained from the graph: memory parameters of the chip; system memory control signals; actuator control signals that are used to create an electrical ladder diagram. The memory control functions are shown in the left part of the ladder, while the actuator control functions are shown in the right part of the ladder diagram. The obtained ladder diagram was verified using the computer aided program FluidSim from Festo. The cyclogram showing the state of actuators and the state of the memory of the system for two cycles of the automatic unit was presented, confirming the design assumptions of the sequential control system. The transformation of the electrical ladder diagram in the ladder program for the PLC is shown. The program of the analyzed system was also verified on a real system using the Simens Simatic S7-1200 controller.

The programming language that has gained the most popularity among PLC programmers is the Ladder Diagram language [9,10,11,12,13]. Reason for this is that it is easy to understand due to the similarity with contact-relay diagrams. The problem was that the solutions proposed by the producers differed from each other, which forced programmers and designers to constantly adapt to new requirements. To solve this, it was decided to standardize the programming methods, i.e. IEC 61131-3 standard. This standard distinguishes two groups of languages, i.e. text and graphic languages [14,15].

Text languages are the IL (instruction list) language, which is structured like assembly language. Among its instructions are logical operations, arithmetic operations, relation operations and functions of timers, flip-flops and counters. The second text language is the ST language (Structured Text), which in turn resembles a high-level language, which is the C language, in its syntax [14].

Graphic languages, i.e. the LD language (ladder diagrams) and the FBD language (function block diagrams), in which the program is built through the use of graphic elements. The LD language resembles contact relay circuits, but in its structure it allows the use of blocks containing more advanced functions. The FBD language, on the other hand, is the equivalent of a signal flow diagram, and the circuits built in it take the form of connected function blocks [9]. Ladder logic also allows performing more complex operations such as arithmetic and timing operations.

The control diagram in this language takes the form of symbols that are placed in circuits resembling a ladder of the relay diagram. This language allows to build control systems based on logical dependencies resulting from Boolean algebra [2].

Adam Szcześniak is with the University of Technology In Kielce, Department of Mechatronics and Machine Building, Poland (e-mail: adam_szczesniak@o2.pl).

Zbigniew Szcześniak is with the University of Technology In Kielce, Department of Electrical Engineering, Automatic Control and Computer Science, Poland (e-mail: z.szczesniak@tu.kielce.pl).

The appearance of the control system resembles electrical circuits built in a con-tact-relay system. The program is written as a series of lines containing graphic elements such as coils or contacts, limited on the right and left side by "current busbars", wherein the right line, depending on the manufacturer's preferences, may not be present.

The basic elements we encounter in LD language are contacts and coils. A contact is an element that transfers the state from its left side to its right side, and the variable assigned to that contact. A contact cannot modify the variable associated with it. The coils also pass the state from left to right side, but in their case the value of the variable assigned to them changes depending on the properties of the coil used.

Function blocks and functions in the LD language are rectangular. As IEC61131-3 describes, each function can only have one output, while a function block can have several outputs. The input parameters of blocks are passed through the variables or constants assigned to them. It is also possible to transfer data directly from the output of one block to the input of the next. Each block must contain at least one Boolean input and output so that current can flow through it [15]. The name of the function block is always entered above the symbol of the function block.

Bearing in mind the popularity of programming PLCs using the LD language in designing sequential circuits [16,17,18], a method of algorithmic design of asynchronous control systems has been developed, which allows for the synthesis of virtually any electro-pneumatic systems used in the automation of technological processes, leading to the obtaining of a control system in the form of a contact ladder.

II. Method of designing a sequential circuit for application to ladder diagrams

In the synthesis of circuits, both in analysis and design, the most common methods of externally describing sequential circuits are verbal description and time plots.

A verbal description formulating the principles of the system operation, assigns input signals in the order of their occurrence the corresponding output signals. The time graphs define the mutual correlation of input and output signals, assigning them values 0 or 1, usually ignoring transient processes.

On the basis of the description of the operating principle of the system, a pattern is created that illustrates the operation of the automatic machine, i.e. the sequence of operation of the actuators.

The connection pattern was adopted, on the example of which the synthesis of cir-cuits will be discussed.

S+/- A+ A- B1+ A+ C+ D+ C- D- A- B1/2- A+ A- B-

On the basis of the connection pattern, the cyclogram of the automatic machine operation has been presented in Fig. 1.

The cyclogram shows the states of the actuators during one operating cycle of the machine. The A, C and D actuators are two-position, while the B actuator is a three-position actuator. In the initial state, all actuators are retracted and their states are signaled by position sensors a0, b0, c0, d0, respectively. The extended position is signaled by sensors a1, b1, c1, d1, respectively, while the intermediate position of the actuator B is signaled by the sensor b1/2.

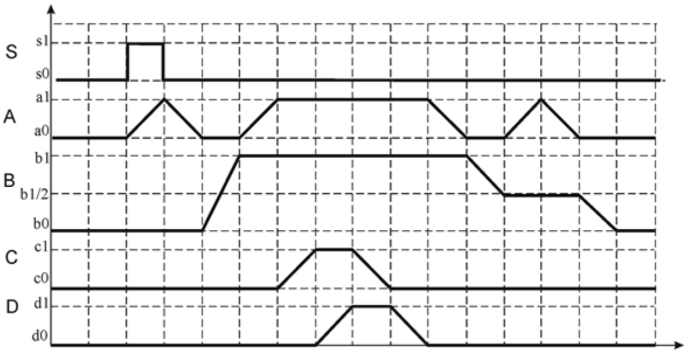


Fig. 1. System work cyclogram.

After giving the start S signal, the actuator A is moved to position a1 and then it returns to position a0. Retracting actuator A extends actuator B to position b1, and then extends actuator A again. After extending actuator A, actuator C extends etc. according to the cyclogram. The sequence ends with actuator B which moves from position b1/2 to position b0. After giving the start signal again, the machine cycle is repeated. Based on the above-mentioned description, we create the layout graph shown in Fig. 2.

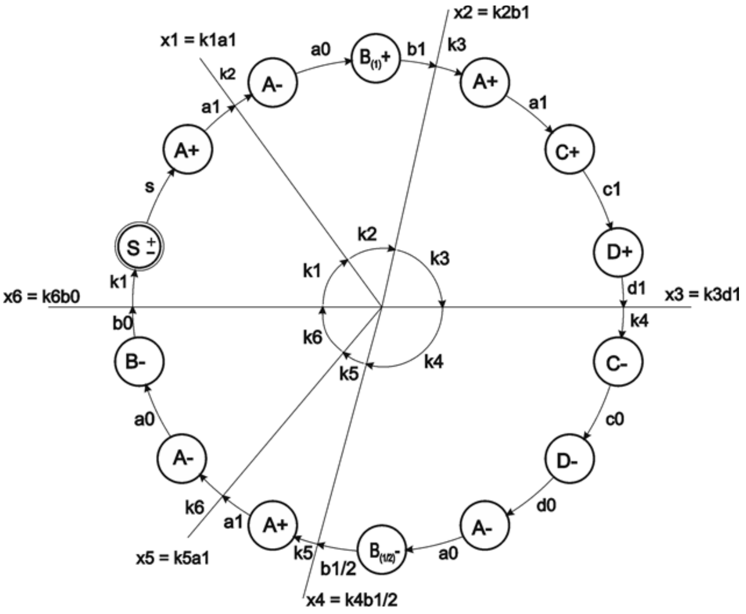


Fig. 2. Graph partition into states.

We create the graph by placing smaller circles on its main circle, the number of which is adequate to the number of actuator states occurring in the connection pattern.

The signal *S* ± that is Start of the system is a stable state, the remaining states are unstable states. Then we divide the graph into groups (circle segments) in such a way that in each group the state of individual elements occurs only once. The groups are represented graphically by dividing lines that come from the center of the circle. We describe each dividing line with an equation in which we multiply the signals from the arc directed to the dividing line (i.e. the state of the actuator) with the value of the group state in front of the dividing line (i.e. the state of the memory of the circuit).

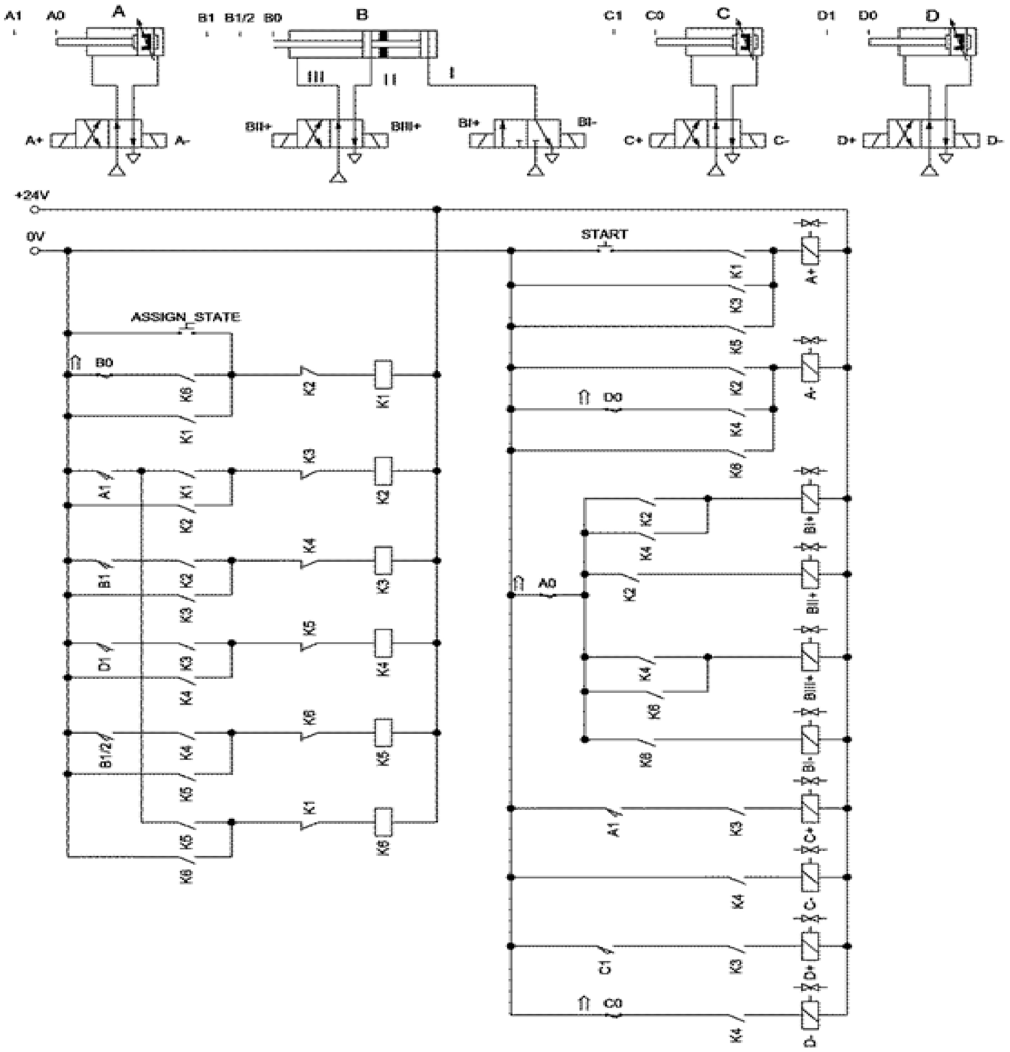


Fig. 3. Sequential circuit using a ladder diagram

Received signals x1 = a1k1; x2 = b1k2; x3 = d1k3; x4 = b1 / 2k4; x5 = a1k5; x6 = b0k6 are the signals that control the memory of the chip.

A detailed description of the creation of a graph is presented in the literature [8].

III. Designing a sequential electro-pneumatic system using a diagram ladder

In the analyzed case, a multi-position B actuator was used, allowing stopping in three positions. The B actuator takes the positions 0, 1/2, 1 marked as B0, B1/2, B1. The design of the actuator requires appropriate control, Fig. 3, depending on the method of supplying or venting the appropriate connection openings I, II, III. Following the cyclogram (system graph), the actuator B takes the positions B0, B1, B1/2, B0 in one cycle of the machine. The function of the valves should be determined by ensuring the appropriate conditions of ports I, II, III. If the actuator is turned on B + (state k2), i.e. it starts from position 0 to 1 (B0 to B1), then valves BI + and BII + must be on. The next position according to the graph is moving the actuator to B1/2 (state k4), i.e. changing the position of the actuator from state B to state B1/2. To ensure this, turn on the BI + and BIII + valves. The next position according to the graph is the insertion of the B- actuator (state k6), i.e. changing the position of the actuator from the B1/2 state to the B0 state. To ensure this, turn on the BIII + valve and turn off the BI - valve.

The actuators are controlled by solenoid valves. The solenoid valves are connected to the actuator chambers in such a way as to ensure the initial state of the actuators. In this case, all actuators are in the retracted position, and their states are signaled by position sensors a0, b0, c0, d0, respectively.

The solenoid valve inputs are controlled by intermediary relays that perform the functions read from the system graph. It is shown on the right side of the ladder in Fig. 3. For example, according to the diagram of the system Fig. 2, the actuator A- switch-off state occurs many times during one cycle, i.e. in the k2, k4 and k6 memory state under the influence of the signals k2, d0k4 and k6, respectively. This is shown as the sum of these signals in the Ladder Diagram.

We proceed in a similar way when defining the functions for determining the state of the other actuators, which was reflected in the control ladder, the right part of the ladder in Fig 3.

For example, the high state of k2 memory is performed by the signal x1 = a1k1 (serial connection of contacts a1 and k1). This state is maintained by the k2 relay contact creating the function a1k1 + k2. The memory state k2 is erased by the next state, i.e. k3, according to the graph, i.e. the function (a1k1 + k2) k3 is created. In this way, the next memory state erases the previous state, setting the memory k4 erases the previous state k3, setting the memory k5 erases the previous state k4, etc.

It should be emphasized that in the case of continuous actuator position control systems, proportional valves [19,20] are used, while for the position of the actuator, digital position transducers [21] are used, which enable precise control of the actuator's position [22,23,24].

The verification of the sequence system with the use of the ladder diagram was carried out using the computer-aided design program FluidSim from Festo.

Fig. 4 shows the selected state of the system during the simulation. The system is in the k6 memory (Fig. 2), in which the actuator A is in the off position, which is signaled by the position sensor A0. This causes the phase of switching off the B actuator to position B0 by switching on the BIII+ valve and switching off the BI- valve.

Fig. 5. shows the cyclograms showing the simulation results. The cyclogram shows the state of the actuators A, B, C, D and the memory state of the k1-k6 circuit in accordance with the machine graph. It should be emphasized that setting the next memory to high state clears the state of the previous memory.

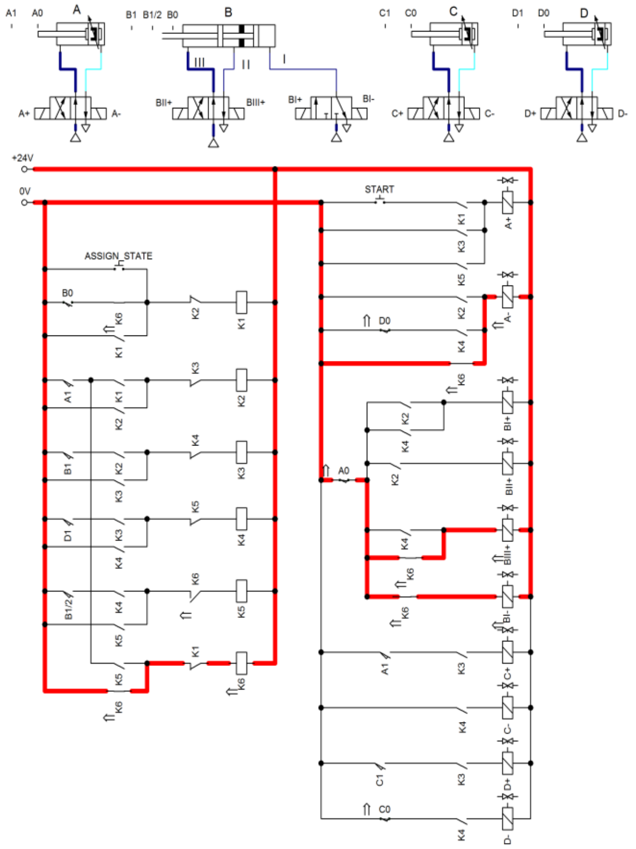
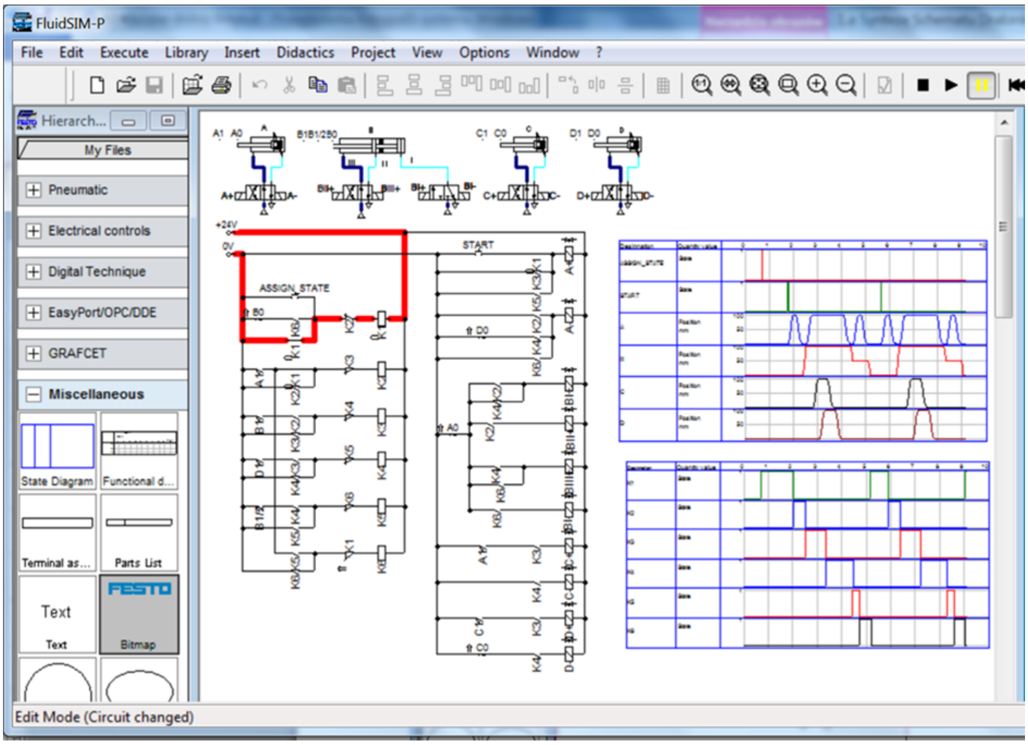


Fig. 4. Sequential circuit with the use of a ladder diagram - Simulation, an exemplary state of the circuit

For example, setting the memory k2 to high state resets the previous state of k1, setting the memory k3 resets the previous state of k2, etc., resulting in the effect of "wandering 1".



Edit Mode (Circuit changed)

Fig. 5. Cyclograms from the simulation of a sequential system with the use of a ladder diagram

TABLE I

DESCRIPTION OF SYMBOLS USED WHEN DEVELOPING THE PROGRAM AND CORRESPONDING ELEMENTS OF THE DESIGNED AUTOMATIC MACHINE

Controller inputs		**Controller outputs**		**Memory**		**Other**	
controller	limit switches	controller	valves	controller	memory states	controller	other
I0.0	a0	Q0.0	A+	M0.0	k1	I0.9	S
I0.1	a1	Q0.1	A-	M0.1	k2	I0.10	ASSIGN_ STATE
I0.2	b0	Q0.2	BI+	M0.2	k3		
I0.3	b1/2	Q0.3	BII+	M0.3	k4		
I0.4	b1	Q0.4	BIII+	M0.4	K5		
I0.5	c0	Q0.5	BI-	M0.5	K6		
I0.6	c1	Q0.6	C+				
I0.7	d0	Q0.7	C-				
I0.8	d1	Q0.8	D+				
		Q0.9	D-				

The cyclogram shows the state of the actuators A, B, C, D and the memory state of the k1-k6 circuit in accordance with the machine graph. It should be emphasized that setting the next memory to high state clears the state of the previous memory. For example, setting the memory k2 to high state resets the previous state of k1, setting the memory k3 resets the previous state of k2, etc., resulting in the effect of "wandering 1".

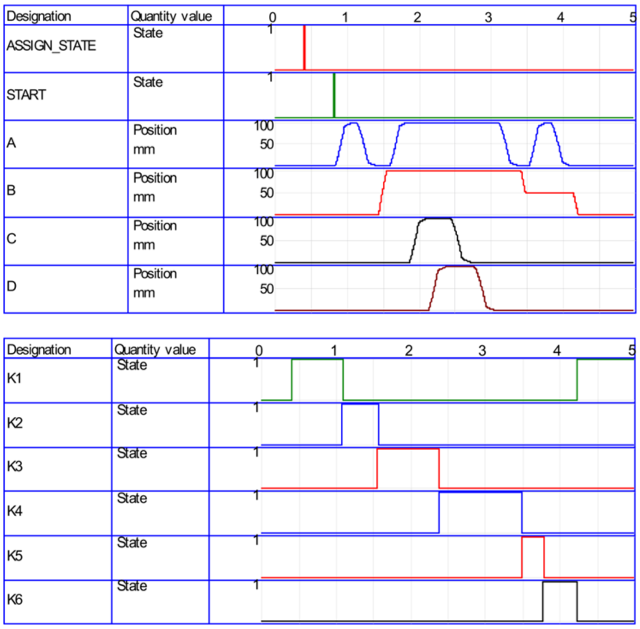


Fig. 6. View of the program window during the simulation of the electro-pneumatic system

In Fig. 6 a screenshot of the ladder diagram during simulation is shown. The simulation presents the system in the state of two cycles of the machine. Starting the system will cause the next cycle or cycles to be performed. The program allows you to assign colors to electric and pneumatic signals and to select the thickness of the signal line.

IV. SYSTEM FOR PROGRAMMING PLC CONTROLLER

For designing sequential electro-pneumatic systems using controllers [25,26,27], a ladder language program is often used, whose logic is identical to that of creating a re-lay-based electro-pneumatic diagram. Based on the graph in Fig. 2, the relay (ladder) system shown in Fig. 3 was created, the logic of which is reflected in the controller program. Signals of the limit sensors on the actuator are connected to the input of the controller, while the control valve coils are connected to the outputs. In addition, the S (start) control signal and other sensors used to control the device are connected to the inputs.

The organization of memory in the controller program is identical with the organization of memory in relays for the electro-pneumatic system. The product or logical sum is implemented by a suitable contact connection: serial or parallel. Description of symbols used when developing the program and corresponding el-ements of the designed automatic machine are presented in Table I.

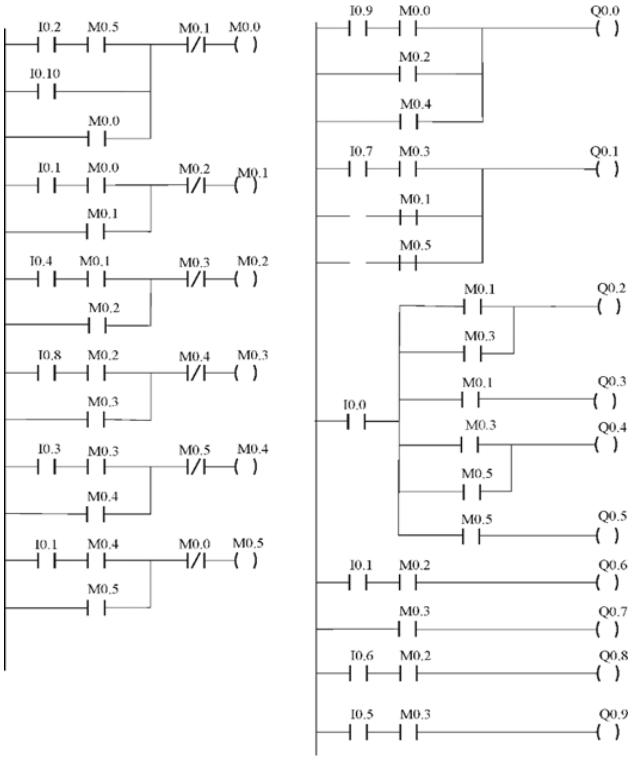


Fig. 7. System for programming PLC controller

The controller control program that implements the graph in Fig. 2 and the electrical ladder diagram Fig. 3 is shown in Fig. 7.

The ladder in the left part of Fig. 7 is responsible for controlling the valve coils. The contacts that control these coils correspond to the position sensors of the actuators. The right part of Fig. 6 is the memory block implementation. The signals controlling the transition to the next state are the signals described on the graph partition boundaries. Each relay coil corresponding to a memory state has a backup and a relay coil con-tact from the next memory state, causing the previous state to be reset to zero when the system enters this state.

The program was tested on a Siemens Simatic S7-1200 controller. The controller has the following parameters: CPU 1214C DC / DC / DC, 14 binary inputs (24V DC) / 10 binary outputs (24V DC) / 2 analog inputs (0-10V DC), power supply: 24V DC, program / data memory: 100 KB.

Conclusion

The course of action presented in the article makes it possible to obtain an electrical diagram of a ladder sequence system.

The electrical ladder diagram is easily converted into a PLC programming ladder diagram.

The applied algorithmic method for the synthesis of the system does not require the use of mathematical analysis to describe the machine function and the procedure of its minimization in order to obtain an electric ladder diagram.

The presented synthesis of the machine can be applied to any sequence of actuators, quickly obtaining a program in the form of a ladder diagram.

The system was verified by simulation using the Festo FluidSim program and on the actual controller Siemens Simatic S7-1200.

Accurate analysis of sequential circuits with the use of PLC circuits can be used to synthesize electric and electronic circuits [28,29,30,31].

Further research will be aimed at implementing the developed method of designing an asynchronous electro-pneumatic system to control the transport paths of rolling elements in the production of rolling bearings, leading to the obtaining of a control system in the form of ladder diagrams for a PLC controller.

References

[1] El-Maleh, A. A Note on Moore Model for Sequential Circuits. ResearchGate.com. Published on July 2016. Available online: https://www.researchgate.net/publication/305268049_A_Note_on_Moore_Model_for_Sequential_Circuits (accessed on 10 September 2021).

[2] Tocci, R.; Widmer, N.; Moss, G. Digital Systems, 12th ed.; Pearson: London, UK, 23 March 2016; ISBN: 0134220137.

[3] Gorzałczany, M.B. Układy Cyfrowe—Metody Syntezy. Tom II: Układy Sekwencyjne, Układy Mikroprogramowane; Wydawnictwo Politechniki Świętokrzyskiej: Kielce, Poland, 2003, p. 370.

[4] Horowitz, P.; Hill. W. The Art of Electronics. Cambridge University Press. New York, 2015

[5] Myczuda, Z., Szcześniak, A. Analiza prądów upływu logarytmicznego przetwornika analogowo-cyfrowego z sukcesywną aproksymacją. Przegląd Elektrotechniczny, 2012, 5a, 247-250

[6] Mychuda, Z., Mychuda, L., Antoniv, U., Szcześniak, A. Logarithmic ADC with Accumulation of Charge and Impulse Feedback – Construction, Principle of Operation and Dynamic Properties, Intl Journal Of Electronics And Telecommunications, 2021, Vol. 67, No. 4, pp. 699-704 https://doi.org/10.24425/ijet.2021.137865

[7] Mychuda, Z., Mychuda, L., Antoniv, U., Szcześniak, A. Logarithmic ADC with Accumulation of Charge and Impulse Feedback – Analysis and Modeling , Intl Journal Of Electronics And Telecommunications, 2021, Vol. 67, No. 4, pp. 705-710 https://doi.org/10.24425/ijet.2021.137866

[8] Szcześniak, A.; Szcześniak, Z. Algorithmic Method for the Design of Sequential Circuits with the Use of Logic Elements. Appl. Sci. 2021, 11, 11100. https://doi.org/10.3390/app112311100

[9] Borden, T.; Cox, R.A. Technician's Guide to Programmable Controllers, 6th ed.; Cengage Learning: Boston, MA, USA, 2021; ISBN: 978-1111544096.

[10] Wikarek, J.; Sitek, P. A Data-Driven Approach to Constraint Optimization. In Advances in Intelligent Systems and Computing; Springer: Singapore, 2019; Volume 920; pp. 135–144.

[11] Sanver, U.; Yavuz, E.; Eyupoglu, C.; Uzun, T. Design and implementation of a programmable logic controller using PIC18F4580. In Proceedings of the 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), Moscow, Russia, 29 January–1 February 2018; pp. 231–235.

[12] Vasu, P.; Chouhan, H.; Naik, N. Design and implementation of optimal soft—Programmable logic controller on multicore processor. In Proceedings of the 2017 International conference on Microelectronic Devices, Circuits and Systems (ICMDCS2017), Vellore, India, 10–12 August 2017; pp.410–414.

[13] Vinces, L.P.; Fernández, P.; Carpio, C.D.; Rocca, E. An Automatic Control System Using the S7-1200 Programmable Logic Controller for the Ethanol Rectification Process. In Proceedings of the 2018 IEEE XXV International Conference on Electronics, Electrical Engineering and Computing (INTERCON), Lima, Peru, 8–10 August 2018; pp. 96–100.

[14] Phan, V. D. ; Vo, C. P.; Dao, H. V.; Ahn, K. K. "Actuator Fault-Tolerant Control for an Electro-Hydraulic Actuator Using Time Delay Estimation and Feedback Linearization," in IEEE Access, vol. 9, pp. 107111-107123, 2021, https://doi.org/10.1109/ACCESS.2021.3101038

[15] Products for Totally Integrated Automation, available online: https://cache.industry.siemens.com/dl/files/167/109744167/att_920955/v1/simatic-st70-complete-english-2017.pdf (accessed on 12 January 2022)

[16] Chhillar, K.; Dahiya, S. Design of Sequential Circuits with Timing Analysis and Considerations. Int. J. Eng. Sci. Comput. 2017, 7, 808–11809.

[17] Ociepka, P.; Herbuś, K. Verification of operation of the actuator control system using the integration the B&R Automation Studio software with a virtual model of the actuator system. IOP Conf. Series: Materials Science and Engineering 227 (2017) 012056; https://doi:10.1088/1757-899X/227/1/012056

[18] Szcześniak, A.; Szcześniak, Z. Designing of Control Systems for Automation of Technological Processes; Kielce University of Technology: Kielce, Poland, 2015.

[19] Acuña-Bravo, W., Canuto, E., Agostani, M., Bonadei, M. Proportional electro-hydraulic valves: An Embedded Model Control solution. Elsevier, Control Engineering Practice Volume 62, May 2017, Pages 22-35, https://doi.org/10.1016/j.conengprac.2017.01.013

[20] Wu, D.; Wang, X.; Ma, Y.; Wang, J.; Tang, M.; Liu, Y. Research on the Dynamic Characteristics of Water Hydraulic Servo Valves Considering the Influence of Steady Flow Force. Flow Meas. Instrum. 2021, 80, 101966.

[21] Szcześniak, A., Szcześniak, Z. Microprocessor processing signals of optoelectronic position transducer, Przegląd Elektrotechniczny, 2009, 4, 153-158

[22] Vo, C.P.; Ahn, K.K. High-precision Position Control of Soft Actuator Systems. Conference: The 3rd International Workshop on Active Materials and Soft Mechatronics (AMSM2018)At: KAIST, Daejeon, South Korea

[23] Zhang, Y.; Yue, H.; Li, K.; Cai,M. Analysis of Power Matching on Energy Savings of a Pneumatic Rotary Actuator Servo-Control System. Chin. J. Mech. Eng. 2020, Pages 13 https://doi.org/10.1186/s10033-020-00445-3

[24] He, S.; Li, D.; Lyu, K.; Zhang, Y.; Yue, H.; Li, K.; Wei, F. Experimental Simulation and Verification of Position Servo Control of Mechanical Rodless Cylinder. Mechanical Engineering Science. 2020, Vol.2, No.2 https://doi.org/10.33142/mes.v2i2.3163

[25] Li, I.-H.; Lee, L.-W. Design and development of an active suspension system using pneumatic-muscle actuator and intelligent control. Appl. Sci. 2019, 9, 4453, https://doi.org/10.3390/app9204453

[26] Jing, X.; Yong, X.; Jiang, Y.; Li, G.; Yokoi, H. Anthropomorphic prosthetic hand with combination of light weight and diversiform motions. Appl. Sci. 2019, 9, 4203, https://doi.org/10.3390/app9204203

[27] Dávila-Vilchis, J.-M.; LAZ-Avilés; Avilés Vilchis, J.C.; Vilchis-González, A.H. Design methodology for soft wearable devices—The MOSAR case. Appl. Sci. 2019, 9, 4727, https://doi.org/10.3390/app9224727

[28] Różowicz, S.; Zawadzki, A.; Włodarczyk, M.; Różowicz, A.; Mazur, D. Assessment of the Impact of Per Unit Parameters Errors on Wave and Output Parameters in a Transmission Line. *Energies* 2021, *14*, 7440. https://doi.org/10.3390/en14217440

[29] Szcześniak,A. Analiza przetwarzania sygnałów logarytmicznego przetwornika analogowo - cyfrowego z sukcesywną aproksymacją. Kielce University of Technology: Kielce, Poland, 2019. ISBN: 978-83-65719-48-5

[30] Szcześniak, A.; Szcześniak, Z.; Mychuda, Z.; Zhuravel,I.; Mychuda, L.; Yelisieieva, H. Mathematical Modelling of the Influence of Parasitic Capacitances of the Components of the Logarithmic Analogue-to-Digital Converter (LADC) with a Successive Approximation on Switched Capacitors for Increasing Accuracy of Conversion. Electronics 2022, 11, 1485, https://doi:10.3390/electronics11091485

[31] Szcześniak, A.; Szcześniak, Z.; Mychuda, Z.; Zhuravel,I.; Mychuda, L. Modelling a New Multifunctional High Accuracy Analogue-to-Digital Converter with an Increased Number of Inputs. Electronics 2022, 11, 1677; https://doi:10.3390/electronics11111677