Solution of logical problem of numerical program control using the software-implemented

D Abdullaeva¹, I Sayliev¹, R Nezhmetdinov² and R Nezhmetdinova²

¹Bukhara Engineering-Technological Institute, Bukhara, Uzbekistan
²“STANKIN” Moscow State Technological University, Moscow, Russian Federation

Abstract. The article covers the requirements for software-implemented logic controllers, analyzes the functionality of the controller with construction of the functional model. The given practical example of Soft PLC controller application in frame of the CNC system shows the advantages of considered approach to design of electrics control systems.

1. Introduction
The process of digital transformation of industrial enterprises is actively proceed in the world. During implementation the activities of these programs, enterprises face a number of problems; in particular, domestic samples of technological equipment do not always meet the requirements of customers. These problems are especially acute in the field of automation of equipment for mechanical engineering industry. This is because there are few competitive enterprises that continue to operate, and the range of equipment they produce is remains limited [1].

Until recently, the lack of domestic automation devices forced enterprises to purchase equipment from Western manufacturers. Nevertheless, a number of restrictions in force on territory of CIS countries force to look for other ways to solve the problem. In this regard, the problem of development and implementation of domestic automation and control devices is acute. However, development of advanced control systems is not always possible on the existing basis of domestic radio-electronic components. It is also should be noted that many domestic developers of automation devices design controllers and control systems based on imported radio-electronic components and manufacture finished products abroad [2, 3].

Nevertheless, analysis of the market for industrial controllers and automation equipment shows that there are trends in the design and implementation of controllers in the form of software solutions without the involvement of additional hardware and Programmable Logic Controllers (PLC). This approach allows avoiding the use of imported hardware, reducing the cost of the control system and gaining a number of advantages, including: adding new functionality and upgrading the controller in a short time; creation of cross-platform application depending on technological problem being solved; the possibility of reducing the start-up time; preservation of investments in software, etc. [4, 5].

2. Methods

2.1. Formation of requirements for a software-implemented controller
Development of computer technologies allowed making available computing resources from real-time control and providing the end user with additional capabilities (services) in the field of equipment
diagnostics, object visualization, electrical control, etc. The peculiarity of the implementation and application of this approach is in the following provisions [6]:

1. Cross-platform form of implementation of Soft PLC. Software-implemented controller should be directed to work within various real-time operating systems (Windows RTX, Windows CE, Linux, etc.). This approach involves installing the Soft PLC core on microcontrollers that support the work of operating system, which allows not to be tied to a specific type of microcontroller.

2. Built-in and off-line Soft PLC implementation options. The ability to install Soft PLC both based on a personal computer, for example, for a solution built into the numerical program control (CNC), and on the hardware base of single-chip microprocessors for autonomous solution.

3. Openness of controller at level of input/output modules. Maintenance of standard industrial communication protocols (CANbus, ModBus, Profibus, Sercos, etc.) will allow connecting ready input/output modules via standard physical communication channels: e.g. RS-232, RS-485, Ethernet, etc.

4. Preservation of investments attracted in the project. Because the core of Soft PLC is a software product, its development and modernization occurs constantly, and does not require serious capital investments, unlike a hardware solution. The use of Soft PLC also allows reducing the final cost of control system by replacing the expensive hardware component with a software solution.

5. Reducing the starting-up and adjustment time. The time of installation and starting-up and adjustment of software-implemented controller on technological equipment is reduced due to the use of software solutions that require only installation and configuration.

2.2. Functional model of a software-implemented controller
Design, development and analysis of the operation of software-implemented logic controller that solves a wide range of production and technical problems requires the use of specialized description and analysis tools. It is proposed to use the IDEF0 methodology as a toolkit for the initial study of the controller's functionality and structure. IDEF is an abbreviation for ICAM Definition (Integrated Computer Aided Manufacturing). The chosen methodology allows modelling functional of control system with the representation of model in graphic notation. The directivity of IDEF0 on the subordination of objects allows considering the logical relations between the functions of the system, without consideration their sequence in time.

To build a functional model of the Soft PLC controller operation, it is necessary to systematize and describe the full set of implemented functions (table 1). For that, it is necessary to select input and specialized data received by the function; the result of function operation as output data; software-implemented controller module that will implement the function.

| Function | Input data | Outputs | Specialized data | CS module implementing the function |
|----------|------------|---------|------------------|-------------------------------------|
| Logic control program development | - Initial conditions | - Logic control program; - Configuration of hardware inputs/outputs; | - Technical task; - Basic electrical diagram. | Development environment for logic control programs |
| Development of user subprogram | Description of a user object | User subprogram | - | Development environment for logic control programs |
3. Results
Because of analysis of the functional of logical control system, a functional model of the system in the IDEF0 notation was developed (figure 1). The logical control system is presented as a set of functions interconnected by links. Each function is a “black box” indicating the inputs, outputs, specialized data and the logic control module responsible for the implementation of the function. Input arrows come to the left edge of the function activity, arrows indicating specialized data - to the upper edge, arrow indicating the system module - to the lower edge, output arrows - to the right edge.
Figure 1. Functional model of logic control systems in IDEF 0 notation.

Functional modeling allowed marking out the main functions of software-implemented controller, binding functions to individual components of a software-implemented controller.

Based on modeling results, the following features can be noted:

- to develop a logic control program, it is necessary to have a technical assignment and a schematic electrical diagram of technological equipment, as well as to determine the initial conditions for the operation of the system;
- developed logic control program must contain user subprograms and configuration of hardware inputs/outputs;
- configuration of hardware inputs/outputs is created based on the table of linking the inputs/outputs of technological equipment to the inputs/outputs of control system;
- debugging of logic control program is carried out according to the testing method;
- performance of the logic control program is carried out in module for the implementation of logic control cycle;
- visualization of operation of the software-implemented controller is carried out in the supervisory control system (SCADA - Supervisory Control and Data Acquisition);
- CNC system can act as an upper-level control system for a software-implemented controller;
• for remote diagnostics and configuration, a specialized application that is not included in the main package of the software-implemented controller is used;
• all the main modules of the software-implemented controller are closed by feedback on error.

3.1. General architecture of the software-implemented SoftPLC controller

In the architecture of the software-implemented controller (figure 2), which controls the electrics, four components can be distinguished: the programming environment (runs under Windows OS), the Soft PLC core (runs under the real-time OS), protocols for interaction with external devices, and physical input/output devices [7].

![Figure 2. General architecture of the software-implemented SoftPLC controller.](attachment://image.png)

Programming environment allows designing and debugging electrics control programs in FBD (functional block diagram) language of the IEC 61131-3 standard [8].

Tool for configuring input/output modules allows creating a hierarchy of devices in the project tree, configuring device parameters and binding variables to input/output channels. The finished configuration is packed into a special structure and transferred to the core of the software-implemented controller with loading application programs.

During debugging, the operator looks through the current parameter values in the configuration module, controls the current values of the input/output channels, receives diagnostic information, and scans the available devices. Remote input/output modules are passive devices that operate on one of the two fastest and most promising twisted pair network protocols: SERCOS III and EtherCAT.

It is advisable to organize the connection between the programming environment and the Soft PLC core based on the TCP/IP protocol stack, with a specialized add-on that allows separating the main data flows of the control system and the data flows of the logical task.

Platform-independent core implies the portability of the controller software to different platforms. This can be, for example, an industrial-grade personal computer with Linux or Windows real-time operating system with the RTX extension, or a single-board computer with an ARM processor and Windows CE operating system. Cross-platform is achieved by moving platform-dependent code into separate libraries and creating wrapper functions for it, used in platform-independent code.

4. Discussion

At the current stage of information technologies progress, when the power and resources of computing core of CNC system do not have strong restrictions, it is possible to replace the traditional PLC, as hardware-software device, with a software controller. In this case, the Soft PLC configuration is
considered as part of the general software and mathematical support of CNC system. As an example, consider a CNC system with a tilted arrangement - machining center.

The network architecture of the specified CNC system includes: real-time machine with a control system core and integrated software-based electrics controller of Soft PLC type; operator terminal (consisting of operator panel on the .NET platform, a standard machine panel with an optional handwheel connection and specialized machine panel) connected to the core via TCP/; input/output extension modules for connecting electrics and linear measuring devices; drive of the main movement and feeds; spindle and counter-spindle controllers.

![Network architecture of numerical program control system](image)

**Figure 3.** Network architecture of numerical program control system.

Data collection and exchange between computing devices in the network is based on the open high-speed EtherCAT protocol. The software-implemented logic controller is responsible for controlling additional technological equipment of machining center, such as automatic tool change system, hydraulic station, pneumatic station, coolant supply, lubrication of units and guides, etc. The large project number of electrical equipment connected to control system required the use of three head input/output modules, which allowed distributing the electrical load between the individual devices. Head input/output modules are also called buscouplers.

Buscouplers are expanded with passive electronic data input/output modules: 120 discrete inputs (15 eight-channel modules), 48 discrete outputs (6 eight-channel modules), 10 analog inputs (2 four-channel modules and 1 two-channel module, 4 ... 20 mA), 6 heat-variable resistor input terminals (3 two-channel modules), 9 analog outputs (2 four-channel and 1 single-channel, 0 ... 10 V), 6 modules of incremental encoder interface.

5. Conclusion
The article covers approach to the construction of software-implemented controllers as means of automation of technological equipment in mechanical engineering industries. Requirements to software-
implemented controller have been formulated, and functional model of a control system based on Soft PLC controller has been constructed.

Proposed approach allows automating technological equipment without using expensive hardware. Implementation of software-implemented controller gives the greatest advantages in control systems, which initially included industrial computer (for example, CNC systems), in this case the Soft PLC controller is integrated in software-mathematical support of control system.

References
[1] Martinov G, Kozak N and Nezhmetdinov R 2018 Approach in Implementing of Logical Task for Numerical Control on Basis of Concept “Industry 4.0” International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM)
[2] Nezhmetdinov R, Nikishechkin P and Nikich A 2018 Approach to the Construction of Logical Control Systems for Technological Equipment for the Implementation of Industry 4.0 Concept International Russian Automation Conference (RusAutoCon) (Sochi: IEEE)
[3] Kovalev I, Nezhmetdinov R and Nikishechkin P 2019 Approach to Assessing the Possibility of Functioning of CNC and PAC Systems on Various Software and Hardware Platforms. International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon Vladivostok: IEEE) 1-5 doi: 10.1109/FarEastCon.2019.8933999
[4] Nikishechkin P, Kovalev I, Kazaryan O and Grigorev N 2020 Practical aspects of constructing a specialized control system for compact machines IOP Conf. Series: Materials Science and Engineering 971(2020) 042085. doi: 10.1088/1757-899X/971/4/042085
[5] Pushkov R, Martinova L and Evstafieva S 2020 Extending functionality of the control system using MTConnect data collection standard IOP Conf. Series: Materials Science and Engineering 971(2020) 042080. doi: 10.1088/1757-899X/971/4/042080
[6] Martinova L, Obukhov A and Sokolov S 2020 Practical Aspects of Ensuring Accuracy of Machining on CNC Machine Tools within Framework of “Smart Manufacturing” International Russian Automation Conference (RusAutoCon) pp 898 - 902 doi: 10.1109/RusAutoCon49822.2020.9208079
[7] Nikishechkin P A, Chervonnova N Yu and Nikich A N 2020 Approach to Development of Specialized Terminals for Equipment Control on the Basis of Shared Memory Mechanism. Lecture Notes in Electrical Engineering LNEE Advances in Automation. 641 181-8. doi: 10.1007/978-3-030-39225-3_20
[8] Nikishechkin P, Grigoriev N and Chervonnova N 2019 Construction of a specialized control system for brush machine and rounding the cutting edges of a metal cutting tool In MATEC Web of Conferences 298 00064 EDP Sciences. doi: 10.1051/matecconf/201929800064