ISSN 1068-7998, Russian Aeronautics (Iz.VUZ), 2016, Vol. 59, No. 2, pp. 293–296. © Allerton Press, Inc., 2016.
Original Russian Text © G.M. Martinov, R.A. Nezhmetdinov, A.U. Kuliev, 2016, published in Izvestiya Vysshikh Uchebnykh Zavedenii, Aviatsionnaya Tekhnika, 2016, No. 2, pp. 128–131.

AIRCRAFT PRODUCTION TECHNOLOGY

Approach to Implementing Hardware-Independent Automatic Control Systems of Lathes and Lathe-Milling CNC Machines

G. M. Martinov, R. A. Nezhmetdinov *, and A. U. Kuliev

Moscow State Technical University of Technology "STANKIN", per. Vadkovskii 1, Moscow, 127055 Russia

**e-mail: neramil@ncsysmemc.ru*

Received October 9, 2014

Abstract—We propose the solution of a logical task in CNC system using software-implemented programmable logical controller (Soft PLC). The solution proposed allows unifying the control algorithms of auxiliary equipment of machine tools and reducing the product launch time of new equipment.

DOI: 10.3103/S1068799816020239

Keywords: CNC system, mill turn center, SoftPLC, tool turret, automatic control systems.

Hardware-independent automatic control of CNC systems assumes transition from the programmable logical controller (PLC) of autonomous implementation to the Soft PLC controller integrated in the core of the control system [1]. This transition allows us to split the level of implementation of the control program and the PLC hardware. In this case, the control program can be adapted to operation with different sets of input/output modules by changing the configuration settings directly in the programming environment.

In the case of temporary, financial, personnel, and other constraints, it is possible to build a combined automatic control system using an autonomous PLC running the existing and well-functioning program and a Soft PLC controller implements new control features that are absent in the previous solution [3].The approach proposed allows reducing the time to market entry for new modifications of CNC machine tools. A functional model of hardware-independent control is shown in Fig. 1.

In designing the automatic control systems, a technical specification for system development serves as the input data. Functions of the system being developed are distributed between the autonomous PLC and the software-based controller, taking into account the complete future transition to the SoftPLC. Control commands (S-, M-, T-functions or program block commands from the machine controls) through the CNC-PLC interface are stored in the shared memory (transferred from the kernel in CNC controller). A software controller performs both a function of scheduling between the core numerical control systems and autonomous PLC and functions of the newly added equipment control [3].

The client terminal generates a control PLC program using complete functional blocks and configuration of devices that are connected to the control system as the input/output modules. The generated program and device configuration are transferred to a module of the software controller that resides in the core of the control system. At startup of technological equipment in the controller cycle, inputs of remote devices are periodically sampled, the algorithm of control is executed, and the corresponding status outputs are set. Data synchronization with the inputs and outputs of remote devices is implemented by the shared memory mechanism [4].

METHODS AND ALGORITHM FOR CALCULATING THE CONFIGURATION OF HARDWARE INPUT/OUTPUT WITH HARDWARE INDEPENDENT CONTROL AUTOMATIC SYSTEMS

Hardware-independent control of automatic systems enables a developer of the control system to determine the range and composition of hardware automation depending on the cost, delivery time, ability to reuse the already installed modules (for example, in the machine modernization), avoiding expensive integrated solutions. The use of open high-speed communication protocols (e.g, the SERCOS or EtherCAT) allows us to build distributed, well-structured industrial real-time systems with high operation speed. The mechanism of interaction uses a shared memory, in which at each execution cycle of the controller, the hardware input/output data and control system kernel are synchronized and the data in the operator interface is updated. [5].

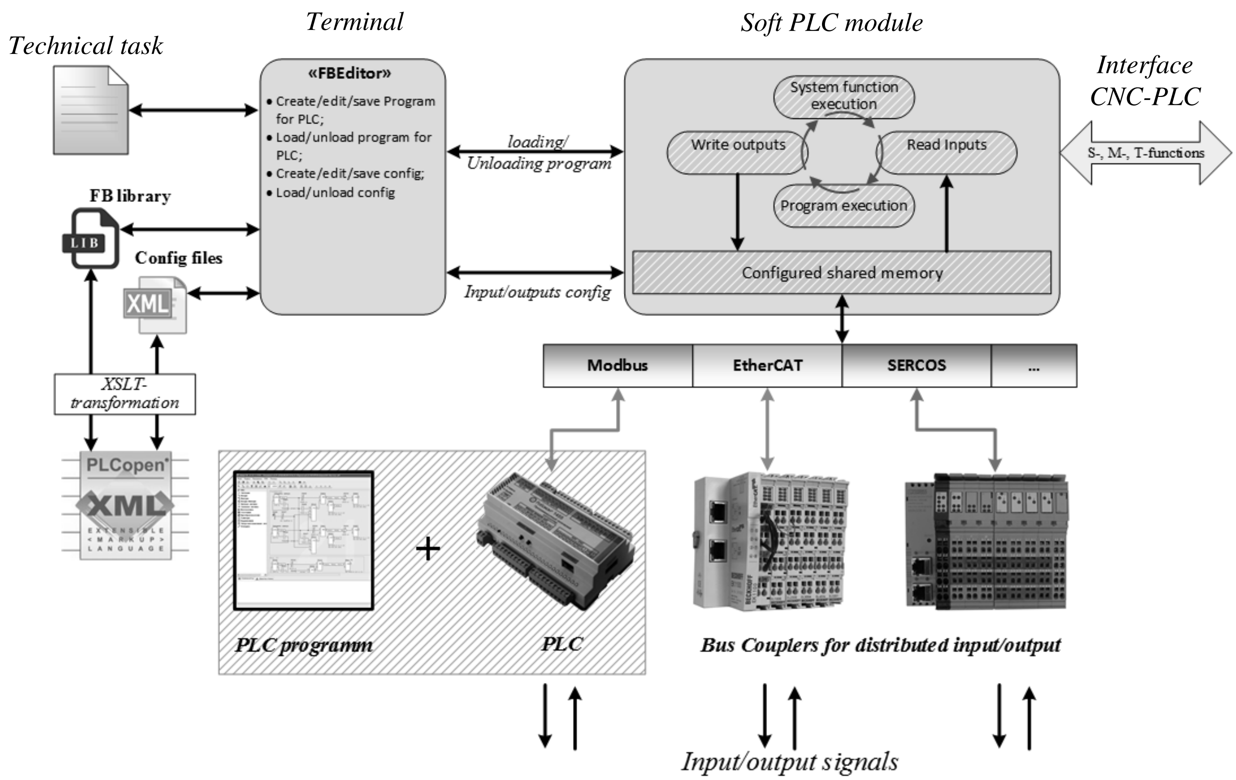

Fig. 1.

A comparison of the logical address used in the control program and the physical address of the hardware modules requires the calculation of the configuration parameters that identify each input/output device in the total array of data. In the technique developed of configuring the hardware input/output resources (Fig. 2), the following conceptual definition is accepted. Module is an independent device having a unique identification number within the framework of a high-speed industrial network; slot is an integral part of the module, operating with one kind of signal (digital, analog, or the like).

At the first step, we perform the initial identification of devices that are connected to the core of CNC system. The program module of communication initiates the acquisition of initial data on the input/output modules, identifying the manufacturer code and model (“Vendor ID” and “Device ID”) and generates a list of available devices.

At the second step, a list of available devices is converted into a matrix of hardware input/output devices. The input/output device is a set consisting of the head (communication) module and input/output slots that are united by an internal bus. The input slot converts a physical signal (digital or analog) to a logic signal for subsequent its synchronization to a cell of shared memory and data processing in the core of the software controller. The output slot reconverts the logic signal generated by the controller to a physical one by the mechanism of shared memory. Each slot is characterized by two parameters, namely, by the volumes of input and output data. The cell matrix is matched against the slot as a minimum unit of the available hardware unit and a row—against a separate input/output device.

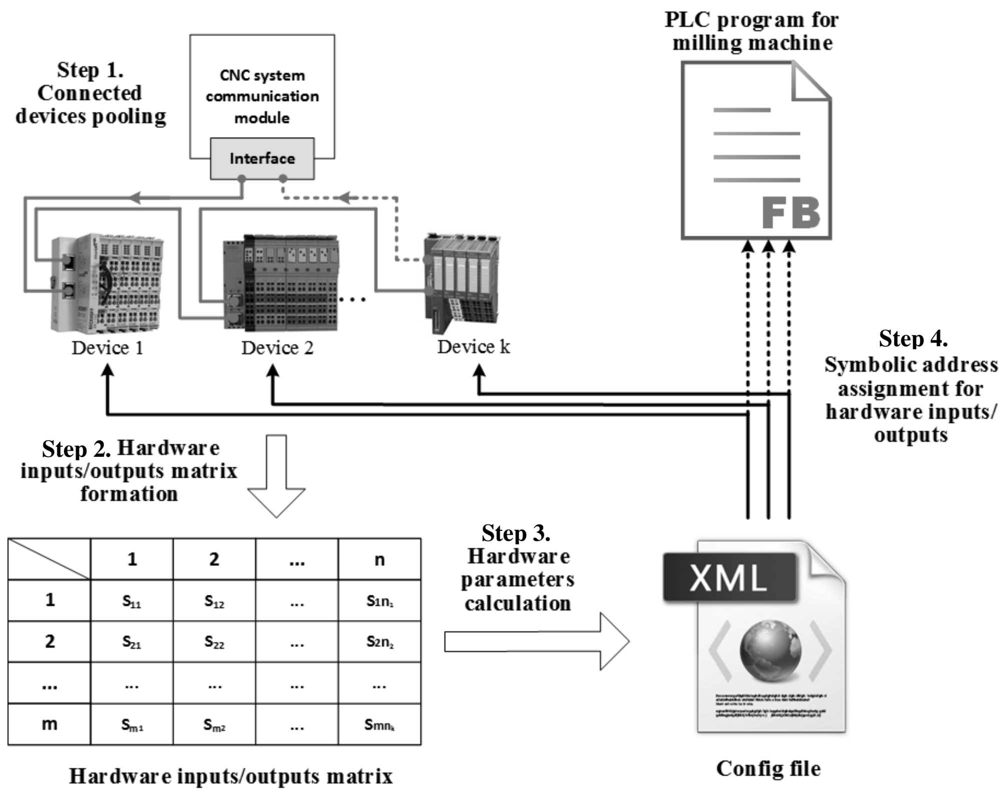

Fig. 2.

At the third step, we calculate the hardware device configuration settings. The address of the initial byte $(N)_{ij}$ of the cell in the shared memory of size $(S_S)_{ij}$ is determined. A data packet is stored in this cell for each *j*th slot, which is part of the *i*th module. For each *i*th module, the total volume of the packet data $(S_D)_i$ for all the slots contained in it is calculated (1):

$$(S_D)_i = \sum_{j=1}^{m} (S_S)_{ij}\,, \tag{1}$$

for each *i*th module the offset $(O_D)_i$ of the relative primary byte of the memory area is calculated as a total sum of the of the preceding module data packets (from the 1st to the (i–1)th). By default, the input offset of the primary byte (Off) is 204800, for the output offset—- (Off) 256000:

$$(O_D)_i = \text{Off} + (O_D)_{i-1} + (S_D)_{i-1} = \text{Off} + \sum_{k=1}^{i-1} (S_D)_k\,. \tag{2}$$

For each *j*th slot, the offset $(O_S)_{ij}$ relative to the primary byte of the *i*th module data is calculated

$$(O_S)_{ij} = (O_S)_{i(j-1)} + (S_S)_{i(j-1)} = \sum_{i-1}^{(j-1)} (S_S)_{il}\,. \tag{3}$$

The total number of the primary byte $(N)_{ij}$ is equal to the sum of the primary byte offset for the *i*th module relative to the zero byte of shared memory and the primary byte offset of the *j*th slot relative to the primary byte of the *i*th module (Eq. 4):

$$\left(N\right)_{ij} = \left(O_D\right)_i + \left(O_S\right)_{ij}. \tag{4}$$

At the fourth step, the number of the first byte of each slot and the total size of each slot of the data packet are associated with a logical address using conventional notation: I_{xy} – the bit Y of the byte X of the input variables area; Q_{xy} – the bit Y of the byte X of the output variables area; I_{B_x} – the byte X (B_x) of the input variables area; Q_{B_x} – is the byte X (B_x) of the output variables area.

In the design of the control program, the approach proposed allows us to abstract from the type and a particular manufacturer of input/output modules. As a result, the executable code is tied to a particular platform not at compile time but at runtime via an XML configuration file, which defines the protocol used by industry and specific hardware modules.

CONCLUSIONS

Wide range of metalworking equipment for engineering companies on the market requires an individual solution for organizing the control of the electric automatic technological equipment [6]. An approach proposed by the authors to hardware-independent control uses the division of auxiliary equipment by the characteristic features on the groups with the separation of the key criteria in them, that helps to unify the process of developing the control programs for the PLC and allows reusing the modules of PLC control programs developed [7]. The approach proposed makes it possible to develop electric automatics machine control system, independent of the type of hardware automation and re-use of the functional blocks developed and debugged.

REFERENCES

1. Grigorev, S.N., Dolgov, V.A., Krasnov, A.V., Kabanov, A.A., and Andreev, N.S., A Method of Technologic Audit of Technical Re-Equipment Projects in Aircraft Production Enterprises, *Izv.Vuz. Av. Tekhnika*, 2015, vol. 58, no. 2, pp. 103–108 [Russian Aeronautics (Engl. Transl.), vol. 55, no. 2, pp. 244–250].
2. Mori, M. and Fujishima, M., Remote Monitoring and Maintenance System for CNC Machine Tools, *Procedia CIRP*, 2013, vol. 12, pp. 7–12.
3. Martinov, G.M., Nezhmetdinov, R. A., and Sokolov S.V., The Principles of Constructing a Toolchain for Monitoring and Setup Mechatronic Equipment Parameters Based on Integration of Specialized Software Components into Control System Structure, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2012, no. 7, pp. 45–50.
4. Martinova, L.I., Grigor'ev, A.S., and Sokolov, S.V., Diagnostics and Forecasting of Cutting Tool Wear at CNC Machines, *Avtomatizatsiay v Promyshlennosti*, 2010, no. 5, pp. 46–50 [Automation and Remote Control (Engl. Transl.), vol. 73, no. 4, pp. 742–749].
5. Nezhmetdinov, R.A., Sokolov, S.V., Obukhov, A.I., and Grigor'ev, A.S., Extending the Functional Capabilities of NC Systems for Control Over Mechano-Laser Processing, *Avtomatizatsiay v Promyshlennosti*, 2011, no. 5, pp. 49–53 [Automation and Remote Control (Engl. Transl.), vol. 75, no. 5, pp. 945–952].
6. Martinova, L.I., Kozak, N.V., Nezhmetdinov, R.A., Pushkov, R.L., and Obukhov, A.I., The Russian Multi-Functional CNC System AxiOMA control: Practical Aspects of Application, *Avtomatizatsiay v Promyshlennosti*, 2012, no. 5, pp. 36–40 [Automation and Remote Control (Engl. Transl.), vol. 76, no. 1, pp. 179–186].
7. Korovin E.M., Lunev A.N., and Tsareva V.V., Grigorev, S.N., Dolgov, V.A., Krasnov, A.V., Kabanov, A.A., and Andreev, N.S., Optimization of NC Parts Machining Processes with Respect to Economic Criteria, *Izv.Vuz. Av. Tekhnika*, 2012, vol. 55, no. 1, pp. 54–57 [Russian Aeronautics (Engl. Transl.), vol. 55, no. 1, pp. 76–82].