DOI: 10.17587/mau.24.462-468

N. L. Dembitsky, dembitsky@mai.ru,
Moscow Aviation Institute, Institute No. 4 "Radioelectronics, Infocommunications and Information Security",
Moscow, 125993, Russian Federation

Corresponding author: **Dembitsky N. L.,** Associate Professor, Ph.D.,
National Research University Moscow Aviation Institute, Institute No. 4 "Radioelectronics,
Infocommunications and Information SecurityMoscow, 125993, Russian Federation, e-mail: dembitsky@mai.ru

Accepted on May 31, 2023

Built-in Analog Automatic Controls for Small Robots

Abstract

One of the most promising methods for improving control efficiency is the parallel synthesis of commands by a group of computing devices. However, delays in digital-to-analog conversions and software processing, the need to synchronize calculations, and signal transmission coordination prevent the achievement of maximum performance in distributed systems. Problems can be solved by increasing the power of embedded microprocessors. This leads to an undesirable increase in design complexity, energy consumption, and dimensions of controls. For the miniaturization of embedded control systems, the article proposes to apply the principles of the indivisibility of physical processes in a technical system and the execution of control commands based on the functional combination of aggregates with analog logic devices (agents). A generalized model of changing the states of a technical system is considered, which sets the principles for switching on or off groups of units. The concept of an automaton of ternary logic is introduced, which determines the order of interaction of aggregates in the time domain. The synthesis of the block diagram of the automaton is carried out. In the mathematical model of the automaton, the operations of ternary logic receive a specific physical content in the form of retrospective processing of changes in the states of aggregates. It is shown that ternary logic automata can be combined into open and closed chains, setting hardware control algorithms units robot. The circuits for connecting automata controlled by elements of asynchronous logic are given. When connected in series, the automata control the order of switching on and off the units. Chains of three-valued logic automata can be arbitrarily long, determining the behavior of the system in the process of its operation. Due to the rejection of program control, such automata can operate in real time, limited only by the delay of signals in logic elements. An important advantage of the proposed technical solution is the unification of the automatic control system. The simplicity of the design of devices and the absence of additional add-ons make it possible to integrate such circuits into the controls of microminiature robots, reducing their weight and size parameters. Promising areas of application of analog automata are microelectromechanical and microelectronic devices.

Keywords: *embedded systems, analog automata, sequential analog logic, ternary logic, microelectromechanical systems, microelectronic control systems*

For citation:

Dembitsky N. L. Built-in Analog Automatic Controls for Small Robots, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2023, vol. 24, no. 9, pp. 462—468.

DOI: 10.17587/mau.24.462-468

УДК 681.5.01

DOI: 10.17587/mau.24.462-468

Н. Л. Дембицкий, канд. техн. наук, доц., dembitsky@mai.ru,
Московский авиационный институт (НИУ)

Встроенная аналоговая автоматика для малоразмерных роботов

Одним из наиболее перспективных методов повышения эффективности управления является параллельный синтез команд группой вычислительных устройств. Однако задержки цифро-аналоговых преобразований и программной обработки, необходимость синхронизации вычислений, согласование передачи сигналов препятствуют достижению максимального быстродействия в распределенных системах. Проблемы могут решаться повышением мощности встроенных микропроцессоров. Это приводит к нежелательному увеличению конструктивной сложности, энергопо-

требления, габаритных размеров органов управления. Для миниатюризации встроенных систем управления в статье предлагается применить принципы неделимости физических процессов в технической системе и исполнения команд управления на основе функционального объединения агрегатов с аналоговыми логическими устройствами (агентами). Рассматривается обобщенная модель смены состояний технической системы, которая задает принципы включения или отключения групп агрегатов. Вводится понятие автомата троичной логики, определяющего порядок взаимодействия агрегатов во временной области. Выполнен синтез структурной схемы автомата. В математической модели автомата операции троичной логики получают конкретное физическое наполнение в виде ретроспективной обработки изменений состояний агрегатов. Показано, что автоматы троичной логики могут объединяться в разомкнутые и замкнутые цепочки, задавая аппаратные алгоритмы управления агрегатами роботов. Приведены схемы соединения автоматов, управляемые элементами асинхронной логики. При последовательном соединении автоматы управляют порядком включения и отключения агрегатов. Цепочки автоматов трехзначной логики могут быть сколь угодно длинными, определяя поведение системы в процессе ее функционирования. Благодаря отказу от программного управления такие автоматы могут работать в режиме реального времени, ограниченного только задержкой сигналов в логических элементах. Важным преимуществом предложенного технического решения является унификация системы управления автоматами. Простота исполнения устройств и отсутствие дополнительных надстроек позволяет встраивать такие схемы в органы управления микроминиатюрных роботов, снижая их массогабаритные параметры. Перспективными областями применения аналоговых автоматов являются микроэлектромеханические и микроэлектронные устройства.

Ключевые слова: *встраиваемые системы, аналоговые автоматы, последовательная аналоговая логика, троичная логика, микроэлектромеханические системы, микроэлектронные системы управления*

Introduction

Software control is the basis of modern automatic control systems. Control algorithms must accurately and promptly respond to changes in external influences. The role of instructions embedded in the program code is growing sharply, and the requirements for speed and reliability of their execution are becoming more stringent. [1]. The combination of complexity of algorithms with restrictions on weight and size parameters causes especially acute problems in the development of computer control of small-sized objects [2].

One of the effective creation of complex control algorithms is the use of embedded systems that have the efficiency of decision-making, increased reliability during operation and reduced development complexity through the use of an evolutionary increase in functional complexity [3—5]. However, in the digital version, this technical solution has certain limitations due to the peculiarities of the operation of microprocessors. Parallel execution of commands by several technical system control aggregates requires their coordinated work [6—8]. Synchronization of calculations, mandatory ADC stages, coordination of controller interaction protocols lead to delays, and high energy costs and an increase in the total complexity of computing devices lead to an increase in weight and size characteristics. This approach becomes especially unacceptable when creating of microelectromechanical systems [9, 10]. The development of highly sensitive microsensors, spatial orientation devices, micromechanical gearboxes creates the prerequisites for the development of new methods of algorithmic control of miniature objects, in which the size and weight of computing devices can become a determining factor [11—13].

This article proposes a way to overcome the above problems of distributed control systems for aggregates, the principles of constructing compact reactive systems are considered [14]. Their feature is the reaction to changes in the parameters of the control object in real time. The solution was obtained by integrating the functional-logical control system into the aggregates of the technical system. The purpose of the work is to create a high-speed distributed control network in the conditions of restrictions on the weight and size parameters of the technical system.

Consider a technical system (TS) consisting of a set of interacting aggregates A. Let us assume that the functioning of such a TS can be represented as a deterministic sequence of processes of various physical nature. For example, the operation of an internal combustion engine involves mechanical, electrical and thermal processes associated with the movement of the piston, ignition and combustion of fuel.

The sequence of processes execution is influenced by various external and internal factors that change the modes and algorithms of the control units. To build a generalized model of changing states, we represent the TS in the form of three sets, the structure of which is capable of continuously changing at each moment of time:

- a set of working aggregates $A(t)$ of the technical system;
- the set of energy flows (EF) synthesized by the aggregates at time t;
- set of EF parameters $X(t)$.

At each moment of time, a certain group of aggregates $A(t)$ is in the active state, determining the mode of operation of the TS. Changes in the values of the energy flows parameters lead to the activation or deactivation of the aggregates. This, in turn, causes a change in the set of energy flows circulating in the network. Changes in the energy flow affect many parameters characterizing the state of the TS.

We will assume that all the listed events occur continuously, affecting the structural, flow and event dynamics of the TS. The dynamic connections of the structural and parametric states of the TS generate a sequential logic of turning on/off the aggregates, which must exist in the time continuum of the functional dynamics of the processes occurring in the TS. To establish a clear interaction between processes, we apply a mathematical apparatus that determines the methods of sequencing and venjunction in the logic of connecting aggregates when changing the states of the vehicle.

In the sequential logic of digital automata, changes are determined by the states before the arrival of the signal and after its arrival. However, these two states do not fully convey the dynamics of changes occurring in the TS. Often, events are associated not only with current processes, but are also determined by previous states of the system. Under the conditions of program control, such problems are solved quite simply. If, in order to increase the efficiency of control, the software component is abandoned, then hardware circuits of the built-in sequential logic will be required. The creation of unified models for successive queues for connecting automata will make it possible to create hardware control algorithms. The developed methods should link the event dynamics of changes in the parameters of EFs with structural changes in the TS. To do this, it is necessary to determine the conditions and rules for the control logic of continuous processes in the TS.

Ternary Parametric Logic for Process Switching

We will evaluate the operation of aggregate automata in three states: the aggregate has not yet connected to the network, the aggregate is working and transfers EF to the network, the aggregate has completed its work and is disconnected from the network. In this aspect, the task of analyzing the operation of the TS will require a transition from binary to ternary logic, in which each process is allocated three consecutive states: "not performed", "in progress", "completed".

To define the logical operations on the EF, we introduce the concept of a physical processor (PhP), which in a distributed system we mean the union of the aggregate and the control body (agent). In the PhP, the aggregate performs transformations and transfers to the EF network, and the agent analyzes the conditions for connecting the aggregate, controls the EF conversion modes and transmits information about the state of the aggregate to the network.

A ternary logic automaton (TLA) consists of three PhPs $\pi\alpha$, $\pi\beta$, $\pi\gamma$ (Fig. 1), the inputs of which receive activating EFs α^-, β^-, γ^-. Each PhP in TLA controls the connection of its aggregate to the network. The aggregates transmit EFs α^+, β^+, γ^+ to the outputs of the PhPs.

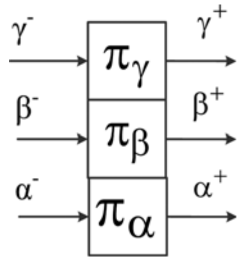

Fig. 1. Conventional designation of the ternary logic automaton

TLA can be in three consecutive states:

a) $r(\pi_\alpha) \in \Re(t), r(\pi_\beta) \notin \Re(t), r(\pi_\gamma) \notin \Re(t)$, EF transfer α^+;

b) $r(\pi_\alpha) \notin \Re(t), r(\pi_\beta) \in \Re(t), r(\pi_\gamma) \notin \Re(t)$, EF transfer β^+;

c) $r(\pi_\alpha) \notin \Re(t), r(\pi_\beta) \notin \Re(t), r(\pi_\gamma) \in \Re(t)$, EF transfer γ^+,

where $r(\pi_\alpha)$, $r(\pi_\beta)$, $r(\pi_\gamma)$ — outgoing EFs of PhP π_α, π_β, π_γ aggregates.

Let us denote by symbols $\mathbb{C}_\alpha$, $\mathbb{C}_\beta$, $\mathbb{C}_\gamma$ the area of definition of the parameters of the incoming EFs α^-, β^-, γ^-. Let us denote by symbols D_α, D_β, D_γ the range of values of outgoing EFs α^+, β^+, γ^+ parameters.

Let's write the predicates that establish whether the parameters $\vec{x}_\alpha, \vec{x}_\beta, \vec{x}_\gamma$ the incoming EFs α^-, β^-, γ^- belong to the areas $\mathbb{C}_\alpha$, $\mathbb{C}_\beta$, $\mathbb{C}_\gamma$:

$$p_\alpha = \{\vec{x}_\alpha \in \mathbb{C}_\alpha\};\ \bar{p}_\alpha = \{\vec{x}_\alpha \notin \mathbb{C}_\alpha\}; \quad (1)$$

$$p_\beta = \{\vec{x}_\beta \in \mathbb{C}_\beta\};\ \bar{p}_\beta = \{\vec{x}_\beta \notin \mathbb{C}_\beta\}; \quad (2)$$

$$p_\gamma = \{\vec{x}_\gamma \in \mathbb{C}_\gamma\};\ \bar{p}_\gamma = \{\vec{x}_\gamma \notin \mathbb{C}_\gamma\}. \quad (3)$$

Let's write down the predicates that establish the belonging of the outgoing EFs α^+, β^+, γ^+ parameters to the ranges of values:

$$p^+_\alpha = \{\vec{x}_{\alpha^+} \in D_\alpha\};\ \bar{p}^+_\alpha = \{\vec{x}_{\alpha^+} \in \bar{D}_\alpha\};$$
$$p^+_\beta = \{\vec{x}_{\beta^+} \in D_\beta\};\ \bar{p}^+_\beta = \{\vec{x}_{\beta^+} \in \bar{D}_\beta\}; \quad (4)$$
$$p^+_\gamma = \{\vec{x}_{\gamma^+} \in D_\gamma\};\ \bar{p}^+_\gamma = \{\vec{x}_{\gamma^+} \in \bar{D}_\gamma\},$$

where $\bar{D}_\alpha, \bar{D}_\beta, \bar{D}_\gamma$ are the ranges of values of outgoing EFs α^+, β^+, γ^+, at which the outgoing EF of PhP π_α, π_β, π_γ aggregates are not connected to the network. The TLA state transition graph is shown in Fig. 2.

The input TLA alphabet is determined by the values of the predicate p_β = {TRUE, FALSE}. The initial state of the TLA S_0 is unstable and corresponds to the disconnection of all outgoing EF of PhP π_α, π_β, π_γ aggregates from the network. It is installed at the beginning of work. In this case, the transition to the state S1 occurs, which corresponds to the inclusion of the aggregate in PhP πα. The condition S1 does not change until the transition of the parameters of the activating EF β^- to the area of permissible values, i.e. until p_β = "FALSE", while, at the same time $p^+_\alpha \cdot \bar{p}^+_\beta = \text{"TRUE"}$. The condition of connecting of PhP πα aggregate to the network will write down in the form of a logical expression

$$p_\alpha = \bar{p}^+_\alpha \cdot \bar{p}^+_\beta \cdot \bar{p}^+_\gamma + p^+_\alpha \cdot \bar{p}^+_\beta = \overline{\overline{\bar{p}^+_\alpha \cdot \bar{p}^+_\beta \cdot \bar{p}^+_\gamma} \cdot p^+_\alpha \cdot \bar{p}^+_\beta}. \quad (5)$$

Using the formula (5), we build a network model for connecting of PhP π_α aggregate (Fig. 3), in which the outgoing EF is formed using logical schemes.

Consider the conditions for the formation of incoming EPs for π_β. In TLA state management, internal EF β^+, γ^+ and external EF β^- are used. According to the conditions of sequential inclusion of aggregates, the EF in this case does not affect of PhP π_β.

Consider two process states:

1) When aggregates of the PhP π_β and π_γ are not connected to the network, p_β = "FALSE", p_γ = "FALSE". The connection condition will be determined by the expression $\bar{p}^+_\beta \cdot \bar{p}^+_\gamma \cdot p^-_\beta = \text{"TRUE"}$.

2) When aggregates of the PhP πβ is already connected to the network, p_β = "TRUE", the condition for continuing the connection will be determined by the expression $p^+_\beta \cdot p^-_\beta = \text{"TRUE"}$.

Combining 1) and 2) disjunctively, we write a generalized expression for connecting aggregate of the PhP π_β to the network:

$$p'_\beta = \bar{p}^+_\beta \cdot \bar{p}^+_\gamma \cdot p^-_\beta + p^+_\beta \cdot p^-_\beta. \quad (6)$$

After performing equivalent transformations and separating the external EF β^-, we obtain an expression for calculating the condition for connecting aggregate of the PhP π_β to the network

$$p_\beta = \overline{(\overline{\bar{p}^+_\beta \cdot \bar{p}^+_\gamma}) \cdot \bar{p}^+_\beta} \cdot p^-_\beta. \quad (7)$$

Using formula (7), we construct a network model for connecting the PhP block π_β (Fig. 4), in which the outgoing EF β^+ is formed using logical circuits.

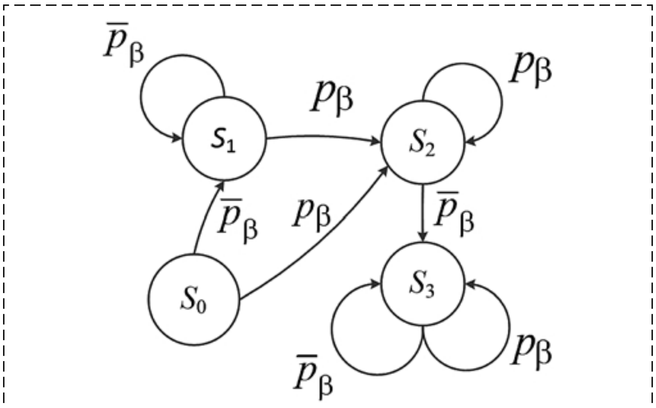

Fig. 2. Graph of transitions of a ternary logic automaton

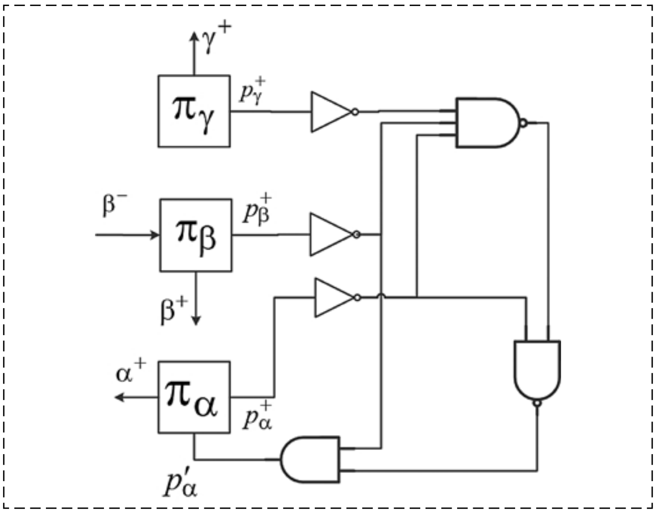

Fig. 3. Fragment of an analog network for connecting of PhP π_α aggregate

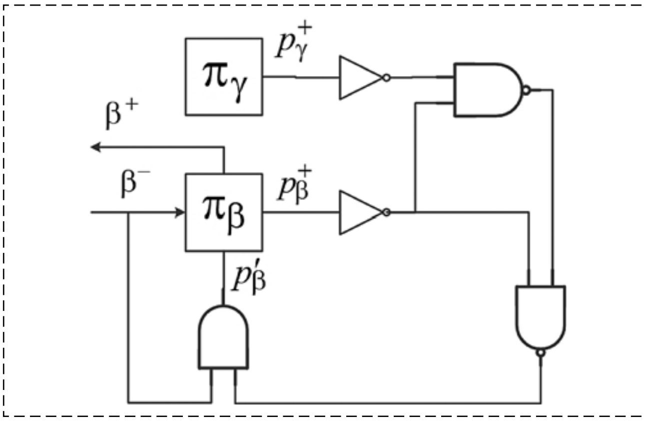

Fig. 4. Fragment of the analog network for connecting aggregate of the PhP π_β

Let us consider the conditions for the formation of incoming EFs for connecting the block of PhP π_γ. We will supply two internal streams α^+, β^+ and one external stream γ^- to the TLA input. Prior to the start of the TLA operation, the EF γ^- parameters are outside t he domain of definition, $p_\gamma^- = \text{"FALSE"}$, so the block of PhP π_γ is disconnected from the network. After the start of the TLA operation, the EF γ^- parameters go into the domain $\mathbb{C}_\gamma$ of definition and $p_\gamma^- = \text{"TRUE"}$. The inclusion of the block PD π_γ will occur when the condition is met

$$p'_\gamma = p_\gamma^- \cdot \bar{p}_\alpha^+ \cdot \bar{p}_\beta^+. \quad (8)$$

Using formula (8), we construct a network model for connecting the FP block π_γ (Fig. 5), in which the outgoing EF is formed using logical circuits.

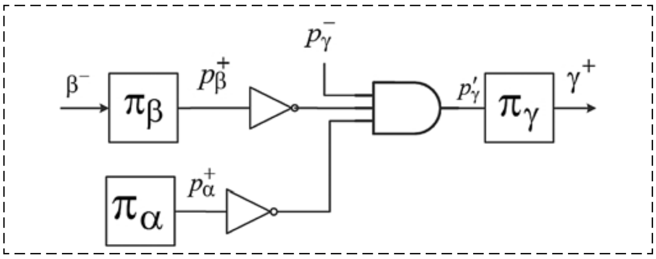

Fig. 5. Fragment of the analog network for connecting aggregate of the PhP π_γ

Combining formulas 5, 7 and 8, we obtain the TLA model (Fig. 1), in which outgoing flows influence the connection of aggregates through predicates in the sequence of execution of rules a, b, c.

From the written conditions for connecting the aggregates, it follows that the main control element is the PhP π_β. Before it is turned on, aggregate of the PhP π_α operates in the network (condition 5). After the activation of β^- is applied to the input of the EF, condition 7 is satisfied, the PhP π_β block is connected, and the blocks of the remaining PhPs are turned off. Violation of condition 7 will lead to the connection aggregate of the PhP π_γ, and aggregate of the PhPs πα, πβ will be permanently disconnected from the network according to conditions 5, 7. Thus, a strict sequence of events is observed, which generate the ternary TLA switching logic.

Ternary logic determines the successive states of aggregates in the PhP π_α, π_β, π_γ. The active states of the aggregates will be denoted by s_α, s_β, s_γ, and the states of switching off the aggregates $\bar{s}_\alpha$, $\bar{s}_\beta$, $\bar{s}_\gamma$. If we use the advance relation symbol "$\prec$" and take into account the conditions for connecting the PhP, then the algorithm for asynchronous switching of aggregates is determined by applying the activation EF β^- to the input of the PhP π_β. We write the process of changing states as an expression

$$s_\alpha, \bar{s}_\beta, \bar{s}_\gamma \prec \bar{s}_\alpha, s_\beta, \bar{s}_\gamma \prec \bar{s}_\alpha, \bar{s}_\beta, s_\gamma. \quad (9)$$

The last state is stable. TLA in this state retains for an arbitrarily long time the fact of turning off the aggregate in the PhP π_β after the completion of the processing of the EF. If a set of TLAs is used in the TS, then for each TLA aggregate s information about the completion of work will be stored. The network formed on the TLA set will record transitions from one stable state to another. Such a network can be interpreted as a finite state machine for processing aggregate states.

Information about the state of the network can be transmitted to other TLAs, setting the algorithm for connecting their aggregates. Next, we consider networks that successively change the states of the PhPs included in them.

Sequential algorithms for controlling a network of aggregates of a technical system

Consider a network consisting of the union of two TLAs A_1 and A_2. TLA A_1 consists of PhPs $\pi_{\alpha 1}$, $\pi_{\beta 1}$, $\pi_{\gamma 1}$, the second TLA A_2 consists of $\pi_{\alpha 2}$, $\pi_{\beta 2}$, $\pi_{\gamma 2}$.

TLA aggregates are activated by internal and external flows. Internal streams are created at the TLA's own outputs, external flows are fed from other TLA's. The incoming EFs β_1 and β_2 control changes in the states of TLA aggregates A_1 and A_2.

Let us direct the EF γ_1 outgoing from TLF A_1 to the input of the PhP $\pi_{\beta 2}$, and the outgoing EF α_1 to the input of the initial setting of the PhP $\pi_{\gamma 2}$. Then the TLA interaction, according to (5), (7), (8), will be associated with the execution of the following logical expressions:

$$\begin{gathered}
p_{\alpha_1} = (\bar{p}_{\alpha_1}^+ \cdot \bar{p}_{\gamma_1}^+ + p_{\alpha_1}^+) \cdot \bar{p}_{\beta_1}^+ = \overline{\overline{\bar{p}_\alpha^+ \cdot \bar{p}_\beta^+ \cdot \bar{p}_\gamma^+} \cdot \bar{p}_\alpha^+ \cdot \bar{p}_\beta^+} \\
p_{\beta_1} = (\bar{p}_{\beta_1}^+ \cdot \bar{p}_{\gamma_1}^+ + p_{\beta_1}^+) \cdot p_{\beta_1}^- = \overline{(\overline{\bar{p}_\beta^+ \cdot \bar{p}_\gamma^+}) \cdot \bar{p}_\beta^+} \cdot p_\beta^- \\
p_{\gamma_1} = p_{\gamma 1}^- \cdot \bar{p}_{\alpha_1}^+ \cdot \bar{p}_{\beta_1}^+ \\
p_{\alpha_2} = (\bar{p}_{\alpha_2}^+ \cdot \bar{p}_{\gamma_2}^+ + p_{\alpha_2}^+) \cdot \bar{p}_{\beta_2}^+ = \overline{\overline{\bar{p}_\alpha^+ \cdot \bar{p}_\beta^+ \cdot \bar{p}_\gamma^+} \cdot \bar{p}_\alpha^+ \cdot \bar{p}_\beta^+} \\
p_{\beta_2} = (\bar{p}_{\beta_2}^+ \cdot \bar{p}_{\gamma_2}^+ + p_{\beta_2}^+) \cdot p_{\gamma_1}^+ \cdot p_{\beta_2}^- = \\
= \overline{(\overline{\bar{p}_{\beta_2}^+ \cdot \bar{p}_{\gamma_2}^+}) \cdot \bar{p}_{\beta_2}^+} \cdot p_{\gamma_1}^+ \cdot p_{\beta_2}^- \\
p_{\gamma_2} = \bar{p}_{\alpha_1}^+ \cdot \bar{p}_{\alpha_2}^+ \cdot \bar{p}_{\beta_2}^+
\end{gathered} \quad (10)$$

As a result of the connection of two TLAs, we obtain a network in which aggregates of the PhP $\pi_{\beta1}$ and $\pi_{\beta2}$ are connected in series. Let's call such a network a two-link chain of switching aggregates (CSU).

According to (10) in TLA A_2:

- The block in $\pi_{\beta2}$ will be connected to the network only if the block in $\pi_{\gamma1}$ is enabled, i. e. EF stream will appear at its output, and $\beta_2^- = \gamma_1^+$, $p_{\beta_2}^- = \{\text{TRUE}\}$.
- In the initial state $\pi_{\gamma1}$, according to the switching conditions, TLA A_1 is blocked, so aggregate of the $\pi_{\beta2}$ will be turned off.
- The aggregate in $\pi_{\gamma2}$ will be connected to the network only if the aggregate in $\pi_{\alpha1}$ is turned off, $p_{\gamma_2}^- = \bar{p}_{\alpha1}^+ = \{\text{TRUE}\}$. This condition guarantees the initial blocking of the aggregate in $\pi_{\gamma2}$.

Let us designate the states of connection of aggregates in TLA A_1 — $s_{\alpha1}$, $s_{\beta1}$, $s_{\gamma1}$, in TLA A_2 — $s_{\alpha2}$, $s_{\beta2}$, $s_{\gamma2}$, and the states of disconnection of aggregates, respectively, by symbols $\bar{s}_{\alpha1}, \bar{s}_{\beta1}, \bar{s}_{\gamma1}, \bar{s}_{\alpha2}, \bar{s}_{\beta2}, \bar{s}_{\gamma2}$. If we use the advance relation symbol $\prec$, then the algorithm for asynchronous switching of FP states can be written as an expression

$$s_{\alpha1}, \bar{s}_{\beta1}, \bar{s}_{\gamma1}, s_{\alpha2}, \bar{s}_{\beta2}, \bar{s}_{\gamma2} \prec \bar{s}_{\alpha1}, s_{\beta1}, \bar{s}_{\gamma1}, s_{\alpha2}, \bar{s}_{\beta2}, \bar{s}_{\gamma2} \prec$$
$$\prec \bar{s}_{\alpha1}, \bar{s}_{\beta1}, s_{\gamma1}, s_{\alpha2}, \bar{s}_{\beta2}, \bar{s}_{\gamma2} \prec$$
$$\prec \bar{s}_{\alpha1}, \bar{s}_{\beta1}, s_{\gamma1}, \bar{s}_{\alpha2}, s_{\beta2}, \bar{s}_{\gamma2} \prec \bar{s}_{\alpha1}, \bar{s}_{\beta1}, s_{\gamma1}, \bar{s}_{\alpha2}, \bar{s}_{\beta2}, s_{\gamma2}. \quad (11)$$

All transitions in the chain (11) are associated with a change in the state of $\pi_{\beta1}$ and $\pi_{\beta2}$, so we can shorten the notation of the sequence:

$$\bar{s}_{\beta1}, \bar{s}_{\beta2} \prec s_{\beta1}, \bar{s}_{\beta2} \prec \bar{s}_{\beta1}, \bar{s}_{\beta2} \prec \bar{s}_{\beta1}, s_{\beta2} \prec \bar{s}_{\beta1}, \bar{s}_{\beta2}. \quad (12)$$

Let's expand the concept of a sequential PhP switching chain. We will connect additional TLAs to the built CSU on the right. At the i + 1-st step, the extension is performed according to the rules (10) for two TLAs A_i with PhPs $\pi_{\alpha i}$, $\pi_{\beta i}$, $\pi_{\gamma i}$ and TLAs A_{i+1} with PhPs $\pi_{\alpha i+1}$, $\pi_{\beta i+1}$, $\pi_{\gamma i+1}$.

On Fig. 6 shows a connection diagram of such a chain.

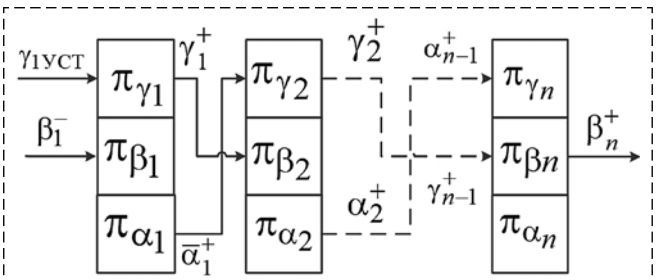

Fig. 6. Series connection of ternary logic automata

Algorithms for cyclic control of an analog network of aggregates of technical systems

The CSU models obtained in the above model provide only one pass through the links and any activating EF β_i do not repeatedly affect the states of the network. To use the CSU again, you will need to completely reset it.

Consider a CSU of two TLAs. TLA A_1 consists of the PhP's $\pi_{\alpha1}$, $\pi_{\beta1}$, $\pi_{\gamma1}$, the second A_2 consists of the PhP's $\pi_{\alpha2}$, $\pi_{\beta2}$, $\pi_{\gamma2}$. The logic of their states is described by expressions (10).

We modify the predicate formulas for calculating the states of the PhP $\pi_{\gamma1}$, adding to it the influence of the EF's coming from the PhP $\pi_{\gamma1}$ and $\pi_{\gamma2}$:

$$p_{\gamma1}^- = \overline{p_{\gamma_1}^+ \cdot p_{\gamma_2}^+}. \quad (13)$$

Then the condition for connecting aggregate of the PhP $\pi_{\gamma1}$ will be determined by the predicate expression:

$$p_{\gamma1} = \overline{p_{\gamma_1}^+ \cdot p_{\gamma_2}^+} \cdot \bar{p}_{\alpha1}^+ \cdot \bar{p}_{\beta1}^+. \quad (14)$$

After turning off the PhP $\pi_{\beta2}$, $p_{\gamma_1}^+ = \text{"TRUE"}$, $p_{\gamma_2}^+ = \text{"TRUE"}$, therefore, according to (14), the block in the PhP $\pi_{\gamma1}$ is turned off, and TLA A_1 goes into the state of waiting for the activating EF. The cycle of connecting CSU blocks is closed.

If we use the advance relation symbol $\prec$, then the algorithm for asynchronous switching of aggregate states can be written as an expression:

$$s_{\alpha1}, \bar{s}_{\beta1}, \bar{s}_{\gamma1}, s_{\alpha2}, \bar{s}_{\beta2}, \bar{s}_{\gamma2} \prec \bar{s}_{\alpha1}, s_{\beta1}, \bar{s}_{\gamma1}, s_{\alpha2}, \bar{s}_{\beta2}, \bar{s}_{\gamma2} \prec$$
$$\prec \bar{s}_{\alpha1}, \bar{s}_{\beta1}, s_{\gamma1}, s_{\alpha2}, \bar{s}_{\beta2}, \bar{s}_{\gamma2} \prec$$
$$\prec s_{\alpha1}, \bar{s}_{\beta1}, \bar{s}_{\gamma1}, \bar{s}_{\alpha2}, s_{\beta2}, \bar{s}_{\gamma2} \prec s_{\alpha1}, \bar{s}_{\beta1}, \bar{s}_{\gamma1}, \bar{s}_{\alpha2}, \bar{s}_{\beta2}, s_{\gamma2}. \quad (15)$$

Thus, a cyclic hardware algorithm for serial connection of TLA A_1 and TLA A_2 aggregates was created (Fig. 7).

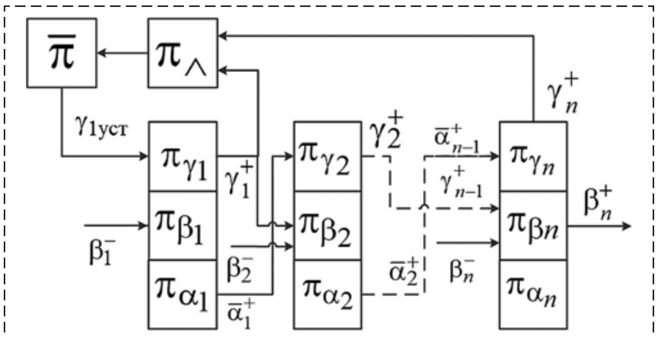

Fig. 7. TLA cyclic compound scheme

Similarly, it is possible to create cyclic hardware algorithms that combine a large number of ATLs.

Consider a CSU consisting of n TLA's. TLA A_1 consists of PhP's $\pi_{\alpha 1}$, $\pi_{\beta 1}$, $\pi_{\gamma 1}$, TLA A_i from PhP's $\pi_{\alpha i}$, $\pi_{\beta i}$, $\pi_{\gamma i}$, TLA A_n from PhP $\pi_{\alpha n}$, $\pi_{\beta n}$, $\pi_{\gamma n}$. Let us direct the EF γ_n^+ to the input of the PhP $\pi_{\gamma 1}$. Let's represent the logic of network state switching in the form of predicate formulas:

$$
\begin{aligned}
p_{\alpha_1} &= (\overline{p}_{\alpha_1}^+ \cdot \overline{p}_{\gamma_1}^+ + p_{\alpha_1}^+) \cdot \overline{p}_{\beta_1}^+ \\
p_{\beta_1} &= (\overline{p}_{\beta_1}^+ \cdot \overline{p}_{\gamma_1}^+ + p_{\beta_1}^+) \cdot p_{\beta_1}^- \\
p_{\gamma_1} &= \overline{p_{\gamma_1}^+ \cdot p_{\gamma_n}^+} \cdot \overline{p}_{\alpha_1}^+ \cdot \overline{p}_{\beta_1}^+ \\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
p_{\alpha_n} &= (\overline{p}_{\alpha_n}^+ \cdot \overline{p}_{\gamma_n}^+ + p_{\alpha_n}^+) \cdot \overline{p}_{\beta_n}^+ \\
p_{\beta_n} &= (\overline{p}_{\beta_n}^+ \cdot \overline{p}_{\gamma_n}^+ + p_{\beta_n}^+) \cdot p_{\gamma_n}^+ \cdot p_{\beta_n}^- \\
p_{\gamma_n} &= \overline{p}_{\alpha_n}^+ \cdot \overline{p}_{\beta_n}^+ \cdot \overline{p}_{\alpha_{n-1}}^+ .
\end{aligned}
\qquad (16)
$$

A closed CSU performs a cyclical algorithm for sequential switching of aggregates. Activating threads β_i^- control the process of asynchronous switching of the TLA.

Conclusion

Unification of hardware solutions for automatic control of technical systems has been carried out. In the considered models, ternary logic receives a specific physical content for processing the states of a technical system according to the signs of sequential execution of processes. A generalized model of combining aggregates for hardware control of branching and cyclic processes in technical systems has been developed. The presented models of functional-logical combination of aggregates into serial chains allow to systematize the development of hardware algorithms for embedded control systems for technical and technological objects. The developed technical solutions create the ability to control objects using unified logic circuits integrated into the aggregates, which do not require analog-to-digital conversion of sensor signals and programmable control devices. As a result, the speed of command synthesis is significantly increased, and the energy and weight and size parameters of the control system are reduced. The declared properties of analog automata open up prospects for their application in microelectromechanical and microelectronic systems.

References

1. **Shibu K. V.** Introduction to Embedded Systems. Tata McGraw-Hill Education, 2009, 402 p.
2. **Syzdykov E. K., Gainutdinova A. V., Krikun K. G.** Development of control system algorithms for mini unmanned aerial vehicle of combined scheme-convertiplane, *Russian Aeronautics*, 2012, vol. 55, no. 2, pp. 199—202 (in Russia).
3. **Michael Barr M., Massa A. J.** Introduction. Programming embedded systems: with C and GNU development tools, O'Reilly Media, Inc., 2006, 336 p.
4. **Oluwole O.** Oyetoke Embedded Systems Engineering, the Future of Our Technology World; A Look Into the Design of Optimized Energy Metering Devices, *International Journal of Recent Engineering Science (IJRES),* December 2015 vol. 18, pp. 17—25.
5. **Catsoulis J.** Designing Embedded Hardware, Released May 2005, O'Reilly Media, Inc., 398 p.
6. **Polikarpova N. I., Shalyto A. A.** Automatic programming, St. Petersburg, 2008, 167 p. (in Russian).
7. **Pandiankal Abhilash V.** A Key to Program Microcontroller System, S Chand & Co Ltd, 2012, 332 p.
8. **Pal A.** Microcontrollers: Principles and Applications, PHI Learning Pvt. Ltd., 2012, 374 p.
9. **Baek S., Bermudez F. G., Fearing R.** Flight control for target seeking by 13 gram ornithopter, *Intelligent Robots and Systems (IROS*), 2011 IEEE/RSJ International Conference on, IEEE, 2011, pp. 2674—2681.
10. **Vijay K., Nathan M.** Opportunities and challenges with autonomous micro aerial vehicles, *The International Journal of Robotics Research*, 2012, vol. 31, no. 11, pp. 1279—1291.
11. **Duhamel P.-E. J., Perez-Arancibia N. O., Barrows G. L., Wood R. J.** Biologically inspired optical-flow sensing for altitude control of flapping-wing microrobots, *Mechatronics, IEEE/ASME Transactions on*, 2013, vol. 18, no. 2, pp. 556—568.
12. **De Wagter C., Tijmons S., Remes B. D., de Croon G.** Autonomous flight of a 20-gram flapping wing mav with a 4-gram onboard stereo vision system, *Robotics and Automation (ICRA), 2014 IEEE International Conference on,* IEEE, 2014, pp. 4982—4987.
13. **Helbling E. F., Fuller S. B., Wood R. J.** Pitch and yaw control of a robotic insect using an onboard magnetometer, *Robotics and Automation (ICRA), 2014 IEEE International Conferenceon*, 2014, IEEE, pp. 5516—5522.
14. **Harel D., Pnueli A.** On the development of reactive systems / In "Logic and Models of Concurrent Systems", NATO Advanced Study Institute on Logic and Models for Verification and Specification of Concurrent Systems, Springer Verlag, 1985, pp. 477—498.
15. **Vasyukevich V. O.** Elements of asynchronous logic. Venjunction and sequence, 2009, 123p., avsilable at: http://asynlog.balticom.lv/Content/Files