REFERENCES

1. Kozlov V.I., Yufit G.A. Computer design of microwave devices. – Moscow: Sovetskoe radio, 1975. – 157 p.
2. Stukach O.V. Variable Attenuator with Low Phase Shift / Proceedings of the 9th European Conference on Wireless Technology. – Manchester, UK, 10-12 September 2006. – P. 241–244.
3. Ilyushenko V.N., Stukach O.V., Tuev V.I. Optimization of attenuators with phase shift invariant to insertion attenuation // Tekhnika Sredstv Svyazi. Ser. Radioizmeritelnaya Tekhnika. – 1990. – Issue 7. – P. 25–29.
4. Walker S. A Low Phase Shift Attenuator // IEEE Transactions on Microwave Theory and Techniques. – 1994. – V. 42. – № 2. – P. 182–185.
5. Application statistic analysis / S.V. Aleksakhin, A.V. Baldin. – Moscow: Prior, 2004. – 221 p.
6. Controlled attenuators / G.M. Krylov, S.I. Khonyak, A.N. Tynynyka et al. – Moscow: Radio i Svyaz, 1985. – 200 p.

Received on 20.09.2007

UDC 004.94

STUDYING SIMULATING TECHNIQUES OF THE SYSTEMS WITH DISCRETE TIME AND SOFTWARE TOOLS IN THE SYSTEM MARS

T.N. Zaychenko

«Research institute of automatics and electromechanics», Tomsk
E-mail: aem@.snc.com.ru

Questions of formalized presentation and simulation of systems with discrete time and software tools in domestic simulation system MARS have been considered. The ways of formalized presentation of discrete systems specified by difference equations and transfer functions and models of algorithm diagram elements are given. The examples of simulating in the system MARS are presented.

Introduction and statement of the problem

The universal system of simulation for personal computer and operational system *Windows* 97/98/NT [2, 3] was developed at the department of electrical engineering theory of Tomsk State university of control systems and radio electronics on the basis of the method of component circuits (MCC) of E.A. Arais and V.M. Dvitriev [1]. In papers [3–5] the questions of solving the problems of electric device dynamics in the system MARS were examined. Simulation of electron control tools of key of power electronics devices and electric apparatus of controlling electric device implementing some private control modes (launch, inhibition and electrical motor reverse, load reception and rejection) was implemented with the help of structure chart components – source of actions of saw tooth and square-stepwise form, comparators and logic elements. The analysis of the devices with microprocessor control sets the problems of simulation of software tools (ST) on PC without development computer. The aim of the given article is studying the approaches to modeling systems with discrete time and software tools in the range of MCC and simulation system MARS.

Concept of simulating systems with discrete time

There are two ways of describing processes in discrete systems: by difference equation (DE) and transfer functions (TF). So, for example, for linear DE of the form

$$y(k)=\sum_{l=0}^{L}a_l x(k-l)-\sum_{m=1}^{M}b_m y(k-m)=0,\ \ k\geq 0, \qquad (1)$$

TF of the argument $z=e^{j\omega}$ has the form

$$W(z)=\frac{a_0+\sum_{l=1}^{L}a_l z^{k-l}}{1+\sum_{m=1}^{M}b_m z^{k-m}}=0,\ \ k\geq 0, \qquad (2)$$

where $x(k-l)$, $y(k-m)$ are input and output signals at time moments $(k-l)T$ and $(k-m)T$; k, l, m, L, M are the integers; T is the sampling time; a_l, b_m are the coefficients; ω is the frequency.

The aim of studying the systems with discrete time is calculation of timing charts or signal numerical sequences.

System MARS gives opportunities of simulating objects permitting formalize presentation in the form of component circuits (CC) from components with bonds of energy and information type in time and frequency domains for linear CC. Energy type bonds are incident two variables – potential and stream; information type bonds – potential. CC mathematical with energy type bonds consists of topological and component equations; with information type bond – only of component ones. The main modes of analysis in time domain in the system MARS are «statics» and «dynamics» including «dynamics explicit» and «dynamics implicit». In the first case differential equation of component models is algebrized by the explicit method of Euler, in the second case – by implicit one. In both cases nonlinear equations are linearized by the method of Newton [1–3]. To organize computing experiment in CC the component-sources of actions and components for visualization of computation results are included. Let us study the capabilities of applying MCC and system MARS for formalize presentation and simulation of systems with discrete time.

It is obvious that using structure chart components referring to the class of components with information type bonds the CC of DE may be formed according to the standard rules of constructing structure charts. To construct CC of linear DE of the form (1) the components are required:

- delays for delayed signal formation;
- proportional link for DE summand formation subject to coefficients a_l, b_m;
- adders for formation of CC equation;
- initial conditions for specifying nonzero values of component variable bonds.

These components enter into composition of the library of structure chart component models in the system MARS [3, 5]. Input and output signals of discrete systems are CC bond variables. To solve DE in the system MARS it is necessary to simulate DE CC in time domain. The results of solution – timing charts of variable bonds of DE CC – may be visualized both in tabular and graphical form.

Computing experiment in DE simulation in the system MARS.

Let us examine DE of this form as an example

$$y(k) = x(k) - x(k-1) - y(k-1), \ k \geq 0, \qquad (3)$$

where $x(k)$, $y(k)$ are input and output signals at time moments kT, k is the integer; T is the sampling time; $x(k–1)$, $y(k–1)$ are input and output signals at previous time moment $(k–1)T$;

$$x(k) = \begin{cases} 1, \text{if } k \text{ even}; \\ 0, \text{if } k \text{ odd}; \end{cases} \qquad (4)$$

$$x(-1) = y(-1) = 0. \qquad (5)$$

Analytical equation (1) has the form:

$$y(0) = x(0) - x(-1) - y(-1) = 1 - 0 - 0 = 1;$$

$$y(1) = x(1) - x(0) - y(0) = 0 - 1 - 1 = -2;$$

$$y(2) = x(2) - x(1) - y(1) = 1 - 0 + 2 = 3;$$

$$y(3) = x(3) - x(2) - y(2) = 0 - 1 - 3 = -4;$$

$$y(4) = x(4) - x(3) - y(3) = 1 - 0 - 4 = 5 \text{ ets.}$$

For the first ten members of input sequence **X**=[1, 0, 1, 0, 1, 0, 1, 0, 1, 0] we obtain **Y**=[1, –2, 3, –4, 5, –6, 7, –8, 9, –10].

CC for simulation of DE of the form (3) in the system MARS is given in Fig. 1, *a*. DE CC is constructed of components «delay» – 1 implementing delay of input and output signals in one time cycle and «adder» – 2. Component «signal source» – 3 being a generator of square pulses is intended for specifying input pulse sequence **X** of the form (4). Zero initial conditions (5) in the system MARS are supported automatically. The gauge of variable bonds – 4 and component «graph» – 5 are included into CC for result visualization. Let us take sampling period equal to T=1 mcs. In this case the component «signal source» should generate singular square pulses with duration of 1 mcs at interval 1 mcs and component «delay» – to delay signal by 1 mcs.

Diagrams given in Fig. 1, *b*, are obtained at simulation of DE CC with constant and variable cycle in modes «dynamics explicit» and «dynamics implicit». In this case simulation cycle is taken equal to h=0,2T=0,2 microsecond. Results of simulation are interpreted as signal timing charts. Increase of a cycle results in rise of signal front duration.

Mode of analysis «with inner iterations»

To obtain results of simulation in the form of numerical sequence elements it is necessary for simulation pitch to be equal to sampling time. If simulation pitch h is taken equal to T=1 s then time integral values may be interpreted as sequence element numbers. However, at $h=T$ numerical solution of the model CC in modes «dynamics explicit» and «dynamics implicit» does not correspond to the analytical one. Therefore, to simulate systems with discrete time the mode of analysis «dynamics with inner iterations» was developed. In the given mode $n_и$ iterations are automatically implemented in the range of simulation pitch h specified by a user that is virtually the C model is computed with constant pitch $h/n_и$ and values of variable bonds of CC with the pitch h are produced to the graph. It should be noticed that for this mode of analysis the sampling time and pitch of solution are taken equal T=h and delay time being a parameter of components «delay» is specified multiple of T.

Diagrams given in Fig. 1, *c* were obtained at simulation of the DE CC examined above (3) in the mode of analysis «dynamics with inner iterations». In this case diagram node points carry meaning. They are interpreted as sequence elements. Here duration of pulse and signal source pause, delay time and simulation pitch equal to 1 s.

Formalize presentation and simulation of discrete elements specified by TF

To describe discrete elements the TF are used along with DE while the aim of simulation is still the computation of timing charts or signal numerical sequences. In this connection the question of selecting the ways of formalize presentation and simulation of the elements the mathematical models of which are specified in the form of TF occurs.

As it is known, TF represents an equivalent method of describing DE that is TF may be converted to the form of DE and vice versa [6]. Therefore, the model of discrete element specified by TF may be presented in the form of CC of equivalent DE which is simulated in time domain. At TF conversion to DE form it should be taken into account that TF of the form $W_з(z)=z^{-k}$ has a lagging element delaying signal by an integer of cycles k. On the basis of this approach the library of discrete component models specified by TF may be developed. In this case, system MARS includes the instrument supporting investigation of component structural models in the form of CC graphics – macrocomponent editor.

Let us examine the example of simulation of discrete filter with TF of the form [7] $W_ф(z)=1/(1+Kz^{-1})$. Taking into account the intermediate conversion

$y(z)=x(z)-Kz^{-1}y(z)$ we obtain the difference equation $y(k)=x(k)-Ky(k-1)$ on the basis of which filter CC may be constructed. CC of investigating this filter is given in Fig. 2, *a*. Filter CC consists of components «delay» – 1, «proportional link» – 2, «adder» – 3. «Source of step excitation» – 4 is connected to input node of filter CC. To measure input $x(k)$ and output $y(k)$ signals meters – 5 are included into CC and component «graph» – 6 for their visualization. Analytical solution of filter model at K=0,5, singular input action $x(k)$=1 at k>0 and zero initial conditions $y(0)$=0 has the form:

$$y(1)=x(1)-y(0)=1-0{,}5\cdot 0=0{,}5;$$
$$y(2)=x(2)-y(1)=1-0{,}5\cdot 0{,}5=0{,}75;$$
$$y(3)=x(3)-y(2)=1-0{,}5\cdot 0{,}75=0{,}625;$$
$$y(4)=x(4)-y(3)=1-0{,}5\cdot 0{,}625=0{,}6875 \text{ и т. д.}$$

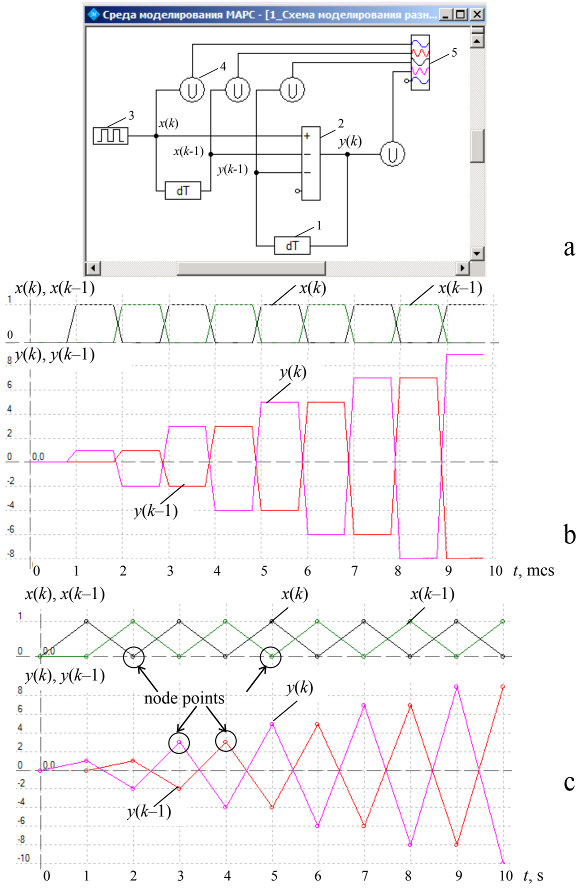


Fig. 1. *Component circuit (a) and results of simulation (b, c) of difference equation*

The results of filter simulation in time domain at sampling interval 1 microsecond are given in Fig. 2, *b*. They coincide with analytical solution and simulation results obtained in the system *Matlab/Simulink* [7].

Concept of software tool simulation

System represents a sequence of operators performing a certain action over the data. The most convenient form of representing algorithms and systems are algorithm diagrams. Actions connected with data processing are reflected at algorithm diagrams as process symbols and data and control flows – by line symbols. It is obvious that there are prerequisites in MCC for solving the tasks of ST simulation. Firstly, CC language having graphic notation allows implementing formalize presentation of algorithms and systems in the form of algorithm diagrams. Secondly, with the help of structural chart components the formalize presentation and computation of rather wide spectra of mathematical expressions [5] being integral part of system operator may be carried out. In this connection let us study the approaches to formalize presentation and simulation of ST within MCC.

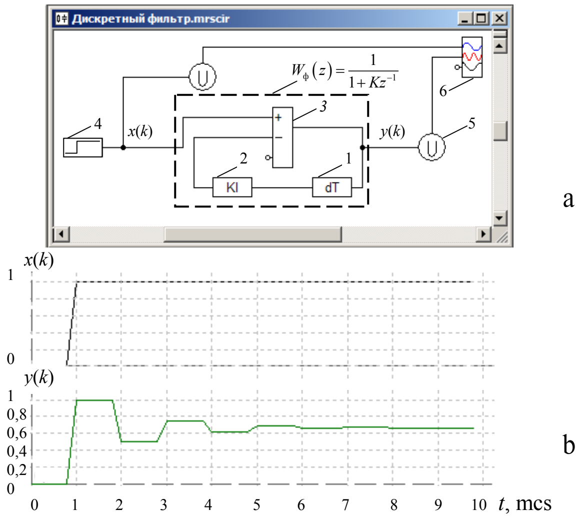


Fig. 2. *Component circuit (a) and simulation results (b) of discrete filter*

According to the standard decomposition of algorithm and program diagrams it is natural to segregate a basic component of algorithm diagram «process» implementing data processing. Data used in the program are variables of CC bonds as at formalize presentation of structural charts: data of scalar type – variables of scalar type bond; data of structural type – variables of vector type bond representing combination of scalar bonds [3]. CC model is the system of algebro-differential equations relative to variable bonds which may take only numerical values. Therefore, in the range of MCC operation with data of integer and real type is naturally organized. Other types of data may be used after introduction of analogies. For example, for Boolean type analogy has the form: meaning «truth» corresponds to a unit value of variable bond and «lie» – to a zero one. This analogy was applied in papers [3, 5] at simulation of digital elements and logical functions.

In connection with the fact that the system is the array of actions the event aspect and control transfer from one component of the type «process» to another should be taken into account at topology-mathematical interpretation of algorithm diagram components. To implement the event aspect within MCC in paper [8] independently simulated subchains with control bonds of information type intended for subchain activation were suggested to be used. At this approach the equations of component models activated only at the moment of subchains are included into CC model. As a result an object CC is CC with variable topology and CC model has the variable dimension.

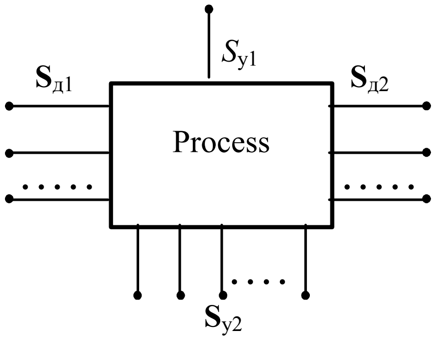


Fig. 3. *Component «process»*

Let us consider the generalized component «process» as independently simulated subchain with information type bonds (Fig. 3). Let us separate among the bonds the controlling bonds supporting control transfer from one process to another and bond-data intended for data transformation. One of controlling bonds S_{y1} is the input one. The variable of this bond is used for activation of the process of input data transformation into output ones. The rest controlling bonds form a set of output bonds $\mathbf{S}_{y2}$. Variables of these bonds are intended for launching data processing processes implemented in other components of the type «process». Among bonds-data a set of input bonds $\mathbf{S}_{д1}$ and output ones $\mathbf{S}_{д2}$, the variables of which have sense of input and output data of the process are also singled out. The main content of the model of component «process» is CC from structural chart components reflecting the transformation of input data $\mathbf{S}_{д1}$ into output $\mathbf{S}_{д2}$. For convenience of formalize presentation the bond-data may be combined in bonds of vector type. To combine scalar bonds into vector one and split vector bond into scalar ones in the system MARS the component «mixer» is supported [5]. Semantics of the bond is reflected by the notation of the form

$$S_n \to (n) \to (V_n) \to (u),$$

where S_n is the bond identifier; n is the topological bond coordinate (node number); V_n is the potential variable of the bond; u is the physical variable influencing the bonds and used in mathematical model of the component.

In the case of one input S_{y1} and one output S_{y2} control bond of the form

$$S_{y1} \to (n_{y1}) \to (V_{ny1}) \to (u_{y1});\ S_{y2} \to (n_{y2}) \to (V_{ny2}) \to (u_{y2}) \quad (6)$$

control is transferred by the following algorithm:

- at u_{y1}=0 process fulfillment is prohibited; CC of data transformation is not activated u_{y2}=0;
- at u_{y1}=1 process fulfillment is allowed; CC of data transformation is activated; the process is fulfilled, u_{y2}=0; if the process is completed then u_{y1}=0, u_{y2}=1, where u_{y1} and u_{y2} are the flags of operation control in this and next component «process».

Component models and rules of construction of algorithm diagrams

Operators used in programs are divided into three groups: simple including operators of assignment, unconditional branch, procedure call; structural – compound, conditional and iteration; input – output. The main control forms of program performance process include performance of operator sequence; performance of operator sequence till a certain condition is true; using test of truth of condition for choice between various possible actions. Therefore, the main operators are operators of assignment and structural. Symbols of algorithm diagrams «process», «solution» and symbol «cycle bound» consisting of two parts correspond to these operators and control forms. Let us examine the methods of topology-mathematical interpretation of the main operators in the range of MCC formalism.

Central operator of the program is the assignment operator of the form «*y*:=*x*» prescribing to fulfill the expression *x* and assign the result of variable *y*. The value of expression is computed in the system MARS when solving the model of CC expression constructed of structural chart components. In this case one of the variable bonds of CC expression is interpreted as its result. Two types of assignment operators should be distinguished: operators in which identifier of variable *y* is not contained in the expression *x* and operators where *y* enters into *x*.

In the first case special component implementing assignment operation is not required but it may be introduced for more convenient visual presentation of CC expression. This task is solved by doubly connected component «assignment operation» ($S_1 \to (n_1) \to (V_{n1}) \to (x)$, $S_2 \to (n_2) \to (V_{n2}) \to (y)$) with model $V_{n1} - V_{n2} = 0$. To construct CC of assignment operator it is necessary to connect bond S_1 to CC bond the variable of which has a sense of expression meaning. Let us also introduce for examining a simply connected component «variable» ($S_1 \to (n_1)$) supporting visualization of variable identifier and having no mathematical model. The examples of CC construction implementing assignment function are given in Fig. 4. Fig. 4, *a* corresponds to initial CC of the expression formed according to structural chart formalism. Variable of the adder output bond makes sense of variable *y* that is reflected by a mark on CC draft. In Fig. 4, *b* component «variable» – 1 is

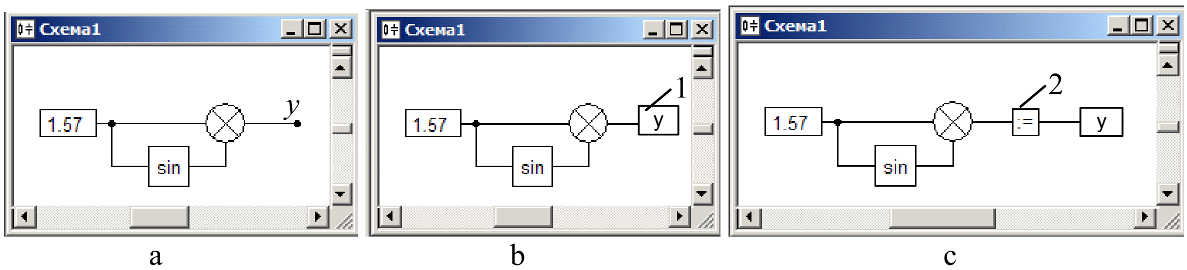


Fig. 4*.* *Examples of construction of expression y=1,57+sin(1,57) CC*

included into CC. The variable name is visualized on component pictorial symbol. Components «variable» and «assignment operation» – 2 are used in CC in Fig. 4, *c*.

In the case when variable identifier is used in both parts of assignment operator the value of the variable in the right part is the value at the previous step of computational process of data processing that is the delayed value. Let us introduce for examining a doubly connected component «assign operation with delay» ($S_1\to(n_1)\to(V_{n1})\to(x)$, $S_2\to(n_2)\to(V_{n2})\to(y)$) intended for assignment of delayed value x_3 by the variable y. The model of the component has the form of linear equation relative to bond variable V_{n2} with variable right part: $V_{n2}=x_3$. Variable bonds are initialized using component «initial conditions» if their values are nonzero at the beginning of simulation.

The example illustrating construction of CC of assignment operator components organization of cyclic computation is given in Fig. 5, *a*. CC of algorithm for computing expressions $X_1=\sum_i i$ and $X_2=\sum_i i^2$ by performing in the cycle two assignment operators of the form $X_1:=X_1+i$; $X_2:=X_2+i^2$, where i is the cycle parameter is given here. Component – 1 supports discrete change of variable i in prescribed limits. Components «initial conditions» – 2 are used for initialization of variables X_1, X_2. Expressions being in the right parts of assignment operators are simulated by components of structural charts of adder and multiplier. Component – 3 is «assignment operation with delay». The results of simulating the given CC at $i=\overline{1,10}$ coinciding with analytical solution $X_1=55$; $X_2=385$ are given in Fig. 5, *b*.

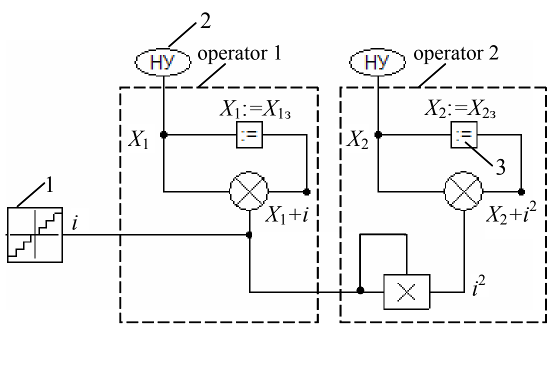
One of operators supporting organization of cyclic computation is the operator of cycle of type FOR prescribing performance of loop body operator at change of current value of cycle i parameter from initial value $N1$ to finite one $N2$. To generate the values of cycle parameter let us introduce for examining the component «cycle FOR parameter». As cycle parameter is usually used at mathematical expression computation and values $N1$ and $N2$ may be not constants it is reasonable to provide three bonds of the component the variables of which make sense $N1$, $N2$ and i. Model of the component supports discrete change of variable bond making sense of current value of cycle i parameter from $N1$ to finite $N2$.

To separate the combination of repetitive computations as a process within the program let us introduce for examination the component «operator of cycle FOR» referring to the group of components of the type «process». The component has two controlling bonds of the type (6) and bonds-data two of which are intended for transmission parameters of cycles $N1$, $N2$ into component the rest bonds $\mathbf{S}_{д2}$ make sense of input-output data of the process. The main content of structural model of component «operator of cycle FOR» as a subchain is CC of loop body operators. Component «parameter of cycle FOR» is included into its composition and in this case discrete change of cycle i parameter from $N1$ to finite $N2$ starting at the moment when component was activated that is at $u_{y1}=1$.

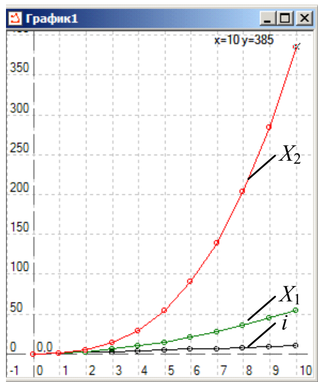
To organize the choice between the processes the components «conditional operator IF» and «conditional operator CASE» of the type «process» are provided. Components have control bonds and bond for data transfer. Component «conditional operator IF» supports activation of one of two alternative processes according to value of expression L being a variable of component bond. If the component is activated ($u_{y1}=1$) the component model initializes variables of output control bonds in accordance with algorithm: if $L=1$ then $u_{y21}=1$, or $u_{y22}=1$. Component «conditional operator CASE» solves the task of activation of one from several processes in accordance with the value of parameter C. In this case the range of parameters $C_z[j]$ determining selection of j process of data processing is the component parameter and value C is the bond variable. If the component is activated ($u_{y1}=1$) the component model initializes variables of output control bonds according to the algorithm: if $C=C_z[j]$ then $u_{y2j}=1$. The components of the type «process» are connected to output control bonds of components «conditional operator IF» and «conditional operator CASE». Their control bonds activate processes following the components «conditional operator IF» and «conditional operator CASE».

To activate operation of algorithm as a whole a singly connected component «operator start» ($S_{y1}\to(n_{y1})\to(V_{ny1})\to(u_{y1})$) supporting initialization of control bond variable $u_{y1}=1$ is provided. Pictorial symbols of base components of algorithm diagrams are given in Fig. 6.

On the basis of suggested concept of ST formalize presentation and models of components of algorithm diagrams let us state the rules of ST CC construction.



a



b

Fig. 5. *CC for performance of operators in cycle (a) and results of its simulation (b)*

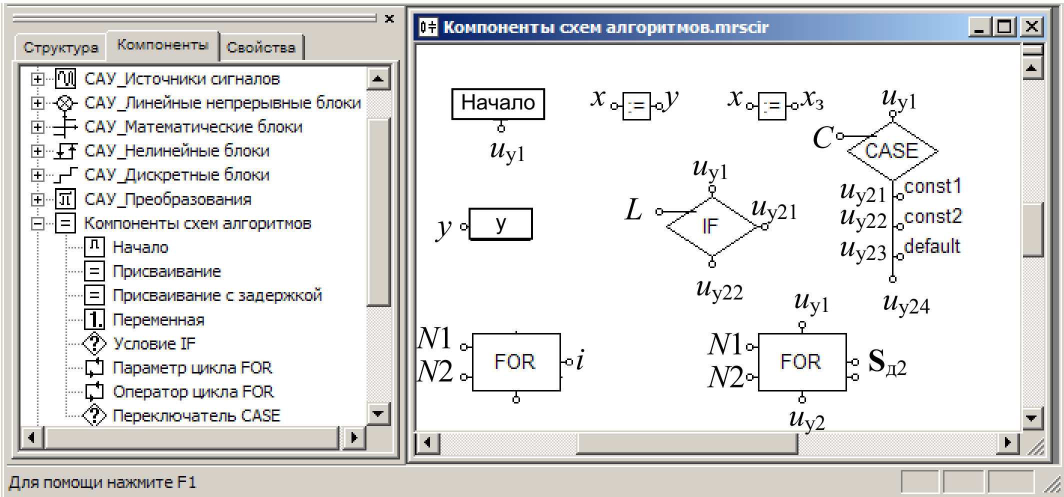


Fig. 6. *Components of algorithm diagrams*

1. ST CC is constructed of algorithm diagram components and structural chart components using block-hierarchical approach.
2. All components are connected into CC by combining control bonds and bond-data according to the chart of computational algorithm.
3. To activate a start of computational process the component «operator start» is connected to input control bond of the first component-process.

To measure variable bonds component-meters are connected to ST CC. ST CC is simulated in time domain in the mode «dynamics with inner iterations».

It should be noted that algorithms implemented by real microprocessor systems may seem to be compound and the process of their presentation in CC form turns out to be labor-intensive. Therefore, further development of approaches to ST simulation is connected with integration of simulation system MARS with the system of automation of mathematical computations «Macrocalculator» [9] developed at the department of electrical engineering theory as well. Theoretical basis of the system «Macrocalculator» is MCC but mathematical language is used as its input language but not chart language as in the system MARS. In this case mathematical expression computation presupposes its conversion to CC form of structural chart and its simulation using simulation programs of the system MARS.

Conclusion

The approaches to formalize presentation and simulation of systems with discrete time and software tools in the simulation system MARS were studied. It was shown that the library of the models of structural chart components in the system MARS allows implementing promptly the formalize presentation and simulation of systems with discrete time. To obtain the results in the form of waveform diagrams it is necessary to use analysis modes «dynamics explicit» or «dynamics implicit»; in the form of numerical sequence – in the mode «dynamics with inner iterations». The models of algorithm chart components and methods of formalize presentation of software tools in the form of component circuits were suggested. The models of basic components of algorithm charts were developed. Correctness of discrete system models and algorithm charts in the system MARS was shown by the examples. The system MARS may be used for studying systems with discrete time and software tools within training tasks.

The suggested techniques of formalize presentation of algorithm diagram elements are the basis of chain presentation of algorithm and software tools in the system of automation of mathematical computation «Macrocalculator».

REFERENCES

1. Arais E.A., Dmitriev V.M. Computer simulation of heterogeneous circuits and systems. – Moscow: Radio i Svyaz, 1982. – 160 p.
2. Automation of functional design of electromechanical systems and devices of transforming engineering // V.M. Dmitriev, T.N. Zaychenko, A.G. Garganeev, Yu.A. Shurygin. – Tomsk: Tomsk University Press, 2000. – 292 p.
3. Dmitriev V.M., Shutenkov A.V., Gandzha T.V. Architecture of universal computer kernel for implementation of virtual laboratories // Pribory i Sistemy. Upravlenie, Kontrol, Diagnostika. – 2004. – № 2. – P. 24–28.
4. Zaychenko T.N. Solving the problems of dynamics of electromechanical systems in the medium of automated simulation MARS // Bulletin of the Tomsk Polytechnic University. – 2005. – V. 308. – № 4. – C. 147–153.
5. Zaychenko T.N. Structural simulation of electrical devices in the system MARS // Pribory i Sistemy. Upravlenie, Kontrol, Diagnostika. – 2005. – № 10. – P. 1–9.
6. Fillips Ch., Harbor R. Control system with feedback. – Moscow: Laboratoriya Basovykh Znaniy, 2001. – 616 p.
7. Dyakonov V.P. Simulink 4. Special manual. – St. Petersburg: Piter, 2002. – 528 p.
8. Zaychenko T.N. Simulation of robot EYE-ARM with velocity vector control // Proizvodstvenno-Tekhnicheskiy Opyt. – 1990. – № 11. – P. 83–88.
9. Eroshkin M.A., Gandzha T.V. Language of mathematical expression presentation for implementation of the editor of Macrocalculator // Kompyuternye Tekhnologii v Obrazovanii / Ed. by V.M. Dmitriev. – Tomsk: Tomsk University Press, 2004. – Issue 2. – P. 23–28.

Received on 20.12.2006