# DECOMPOSITIONAL DESIGN OF AUTOMATA BASED ON PLA WITH MEMORY

I. S. Levin

Avtomatika i Vychislitel'naya Tekhnika,
Vol. 20, No. 2, pp. 66-73, 1986

UDC 681.3-181.48:512.56

A decomposition of microprogramming automata is suggested on the basis of coding complete output sets and communication signals transmitted between network components. The method is intended for use in the design of automata on the basis of PLA with internal memory.

The design of control devices on the basis of a PLA with internal memory (PLAM) is usually reducible to the decomposition of a microprogramming automaton (MPA) that scribes the operation of the device being constructed. The initial MPA, whose parameters (the number of input and output variables, the number of transitions, and the requisite number of memory elements) do not meet the constraints specified (the number of inputs, outputs, terms, and memory elements of PLAM), is realized as a network of component automata each of which satisfies the constraints.

The need for the decomposition stems from the insufficiency of the available resource in PLAM, particularly, the shortage of external terminals. The interaction of the automata in the network is organized by introducing additional links between component automata, which further reduces the available PLAM resources. One of the possible ways for minimizing the number of connections is by encoding the signals transmitted between the component automata (communication signals). The existence of a memory in PLA provides an additional capability for reducing the number of links.

We suggest two decomposition methods:

1. The method of decomposition of the initial MPA S into a network of two automata, where one (V-automaton) has a small number of input variables, and the other (G-automaton) has a small number of output variables.

2. The method of decomposition of the G-automaton for a given partition of the set of states.

When these methods are used to create MPA on the basis of PLAM, the number of variables transmitted between component automata is minimized, and the efficiency of the utilization of the internal memory is increased.

## DECOMPOSITION OF MPA S

In [1], a method of synthesis of MPA on PLA was suggested, based on encoding the rows of the structural table of the automata. The method of decompositional synthesis suggested here is based on independently encoding complete output sets inside the fragments of the table describing the transitions from a single state of MPA. The coding makes use of the same variables in each of the fragments, reducing the number of code variables.

The input signals (conjunctions) of the initial MPA are converted into the encoded output sets to create a G-automaton. The variables which encode the complete output sets are output variables of G-automaton, and, at the same time, input variables of V-automaton, which is responsible for decoding of the output sets.

The joint operation of G- and V-automata occurs as follows. The G-automaton receives the input set, and, according to its output functions, sends to the input of the V-automaton the code of the output set. After receiving this code, V-automaton produces at its output the appropriate output set. The two automata go into the new state defined by the transition function of the initial automaton.

Table 1

| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|
| $N$ | $a_m$ | $a_s$ | $X(a_m, a_s)$ | $Y(a_m, a_s)$ | $G(a_m, a_s)$ | $R(a_m, a_s)$ |
| 1 | $a_1$ | $a_2$ | $\bar{x}_2\bar{x}_4$ | $y_1y_2y_3y_4y_5$ | $g_1$ | — |
| 2 | | $a_2$ | $x_1\bar{x}_2$ | $y_1y_2y_3y_4y_5$ | $g_1$ | — |
| 3 | | $a_3$ | $\bar{x}_1\bar{x}_2x_4$ | $y_2y_3y_9y_{10}$ | $g_2$ | $r_1$ |
| 4 | | $a_3$ | $x_2\bar{x}_3x_4$ | $y_2y_3y_9y_{10}$ | $g_2$ | $r_1$ |
| 5 | | $a_4$ | $x_2x_3$ | $y_{14}y_{18}$ | $g_3$ | $r_2$ |
| 6 | | $a_6$ | $x_2\bar{x}_3\bar{x}_4$ | $y_1y_4$ | $g_4$ | $r_2r_1$ |
| 7 | $a_2$ | $a_6$ | $x_1x_3$ | $y_{19}y_{20}$ | $g_1$ | — |
| 8 | | $a_7$ | $\bar{x}_3$ | $y_8y_{19}$ | $g_2$ | $r_1$ |
| 9 | | $a_7$ | $\bar{x}_1$ | $y_8y_{19}$ | $g_2$ | $r_1$ |
| 10 | $a_3$ | $a_4$ | 1 | $y_{14}y_{18}$ | $g_1$ | — |
| 11 | $a_4$ | $a_5$ | $\bar{x}_5\bar{x}_6x_7$ | $y_1y_4$ | $g_1$ | — |
| 12 | | $a_6$ | $x_5x_6$ | $y_1y_4y_{15}$ | $g_2$ | $r_1$ |
| 13 | | $a_7$ | $\bar{x}_5x_6$ | $y_8y_{19}$ | $g_3$ | $r_2$ |
| 14 | | $a_8$ | $x_5\bar{x}_6$ | $y_2y_{10}y_{13}$ | $g_4$ | $r_2r_1$ |
| 15 | | $a_8$ | $\bar{x}_5\bar{x}_6\bar{x}_7$ | $y_2y_{10}y_{13}$ | $g_4$ | $r_2r_1$ |
| 16 | $a_5$ | $a_6$ | $\bar{x}_7\bar{x}_2x_3$ | $y_1y_4y_{15}$ | $g_1$ | — |
| 17 | | $a_7$ | $x_7x_2$ | $y_8y_{19}$ | $g_2$ | $r_1$ |
| 18 | | $a_8$ | $\bar{x}_7x_2$ | $y_2y_{10}y_{13}$ | $g_3$ | $r_2$ |
| 19 | | $a_9$ | $x_7\bar{x}_2$ | $y_{10}y_{12}y_{15}$ | $g_4$ | $r_2r_1$ |
| 20 | | $a_{10}$ | $\bar{x}_2\bar{x}_3\bar{x}_7$ | $y_8y_{10}y_{20}$ | $g_5$ | $r_3$ |
| 21 | $a_6$ | $a_2$ | $x_3x_4\bar{x}_8$ | $y_6y_7y_8$ | $g_1$ | — |
| 22 | | $a_3$ | $\bar{x}_3x_4$ | $y_{11}y_{12}y_{13}$ | $g_2$ | $r_1$ |
| 23 | | $a_6$ | $x_3x_4x_5\bar{x}_6x_7x_8$ | $y_1y_4y_{16}$ | $g_3$ | $r_2$ |
| 24 | | $a_7$ | $\bar{x}_4\bar{x}_6\bar{x}_7$ | $y_8y_{19}$ | $g_4$ | $r_2r_1$ |
| 25 | | $a_9$ | $\bar{x}_4x_5x_6$ | $y_{10}y_{12}y_{15}$ | $g_5$ | $r_3$ |
| 26 | | $a_{10}$ | $\bar{x}_4x_5\bar{x}_6$ | $y_8y_{10}y_{20}$ | $g_6$ | $r_3r_1$ |
| 27 | | $a_{10}$ | $\bar{x}_4\bar{x}_6x_7$ | $y_8y_{10}y_{20}$ | $g_6$ | $r_3r_1$ |
| 28 | | $a_8$ | $\bar{x}_4\bar{x}_5x_6x_7$ | $y_6y_7y_{12}$ | $g_7$ | $r_3r_2$ |
| 29 | | $a_8$ | $x_3x_4x_6x_8$ | $y_6y_7y_{12}$ | $g_7$ | $r_3r_2$ |
| 30 | | $a_8$ | $x_3x_4\bar{x}_7x_8$ | $y_6y_7y_{12}$ | $g_7$ | $r_3r_2$ |
| 31 | | $a_8$ | $x_3x_4\bar{x}_6x_8$ | $y_6y_7y_{12}$ | $g_7$ | $r_3r_2$ |
| 32 | $a_7$ | $u_8$ | 1 | $y_2y_{10}y_{13}$ | $g_1$ | — |
| 33 | $a_8$ | $a_8$ | $\bar{x}_1$ | — | $g_1$ | — |
| 34 | | $a_9$ | $x_1$ | $y_8y_9y_{16}$ | $g_2$ | $r_1$ |
| 35 | $a_9$ | $a_5$ | $x_2$ | $y_{11}y_{14}y_{15}$ | $g_1$ | — |
| 36 | | $a_6$ | $\bar{x}_2$ | $y_3y_8y_{15}$ | $g_2$ | $r_1$ |
| 37 | $a_{10}$ | $a_8$ | $x_1$ | $y_8y_9y_{16}$ | $g_1$ | — |
| 38 | | $a_9$ | $\bar{x}_1$ | $y_9y_{16}$ | $g_2$ | $r_1$ |

We can now give a formal definition of G- and V-automata.

The following notations are used for the description of the initial MPA S(A, X, Y, $\delta$, $\lambda$, $a_1$): $A = \{a_1, \ldots, a_M\}$, $X = \{x_1, \ldots, x_L\}$, $Y = \{y_1, \ldots, y_N\}$ are the sets of states of the input and output variables, respectively; $\delta$, $\lambda$ are the functions of transitions and outputs; and $a_1$ is the initial state. We will describe the process in parallel to decompositional synthesis of MPA S(A, X, Y, $\delta$, $\lambda$, $a_1$) defined by the transition table (columns 1, 2, 3, 4, Table 1), on the basis of PLAM with these parameters: 7 inputs, 8 outputs, 32 terms, 6 memory elements.

We will first define the G-automation ($A^G$, $X^G$, $Y^G$, $\delta^G$, $\lambda^G$, $a_1{}^G$). The sets of states, the input variables, and the initial state of the G-automaton are identical with the respective sets of the initial automaton S: $A^G = A$, $X^G = X$, $a_1{}^G = a_1$.

To define the output alphabet and the functions of G-automaton we form, on the set $H(a_m)$ of the transitions from the state $a_m$ (m = 1,..., M) of the initial automaton, the partition $\chi_m = \{\chi_1{}^m, \ldots, \chi_{r_m}{}^m\}$, such that the transitions $h_k$ and $h_l$ are contained within the same

Table 2

| $N$ | $a_m$ | $a_s$ | $R(a_m, a_s)$ | $Y(a_m, a_s)$ |
|---|---|---|---|---|
| 1 | $a_1$ | $a_2$ | $\bar r_1 \bar r_2$ | $y_1y_2y_3y_4y_5$ |
| 2 | | $a_3$ | $r_1 \bar r_2$ | $y_2y_3y_9y_{10}$ |
| 3 | | $a_4$ | $\bar r_1 r_2$ | $y_{14}y_{15}$ |
| 4 | | $a_5$ | $r_1 r_2$ | $y_1y_4$ |
| 5 | $a_2$ | $a_6$ | $\bar r_1$ | $y_{19}y_{20}$ |
| 6 | | $a_7$ | $r_1$ | $y_8y_{19}$ |
| 7 | $a_3$ | $a_4$ | 1 | $y_{14}y_{15}$ |
| 8 | $a_4$ | $a_5$ | $\bar r_1 \bar r_2$ | $y_1y_4$ |
| 9 | | $a_6$ | $r_1 \bar r_2$ | $y_1y_4y_{15}$ |
| 10 | | $a_7$ | $\bar r_1 r_2$ | $y_1y_{19}$ |
| 11 | | $a_8$ | $r_1 r_2$ | $y_2y_{10}y_{13}$ |
| 12 | $a_5$ | $a_6$ | $\bar r_1 \bar r_2 \bar r_3$ | $y_1y_4y_{15}$ |
| 13 | | $a_7$ | $r_1 \bar r_2 \bar r_3$ | $y_8y_{19}$ |
| 14 | | $a_8$ | $\bar r_1 r_2 \bar r_3$ | $y_2y_{10}y_{13}$ |
| 15 | | $a_9$ | $r_1 r_2 \bar r_3$ | $y_{10}y_{13}y_{15}$ |
| 16 | | $a_{10}$ | $\bar r_1 \bar r_2 r_3$ | $y_8y_{10}y_{20}$ |
| 17 | $a_6$ | $a_2$ | $\bar r_1 \bar r_2 \bar r_3$ | $y_6y_7y_8$ |
| 18 | | $a_3$ | $r_1 \bar r_2 \bar r_3$ | $y_{11}y_{12}y_{13}$ |
| 19 | | $a_6$ | $\bar r_1 r_2 \bar r_3$ | $y_1y_4y_{15}$ |
| 20 | | $a_7$ | $r_1 r_2 \bar r_3$ | $y_8y_{19}$ |
| 21 | | $a_9$ | $\bar r_1 \bar r_2 r_3$ | $y_{10}y_{13}y_{15}$ |
| 22 | | $a_{10}$ | $r_1 \bar r_2 r_3$ | $y_8y_{19}y_{20}$ |
| 23 | | $a_{11}$ | $\bar r_1 r_2 r_3$ | $y_6y_7y_{12}$ |
| 24 | $a_7$ | $a_8$ | 1 | $y_2y_{10}y_{13}$ |
| 25 | $a_8$ | $a_8$ | $\bar r_1$ | — |
| 26 | | $a_9$ | $r_1$ | $y_6y_7y_{13}$ |
| 27 | $a_9$ | $a_5$ | $\bar r_1$ | $y_1y_{14}y_{15}$ |
| 28 | | $a_6$ | $r_1$ | $y_3y_6y_{15}$ |
| 29 | $a_{10}$ | $a_8$ | $\bar r_1$ | $y_8y_9y_{16}$ |
| 30 | | $a_9$ | $r_1$ | $y_8y_{16}$ |

partition block $\chi_m$ ($h_k \equiv h_l(\chi_m)$, if and only if $\delta(a_m, X_k) = \delta(a_m, X_l)$, $\lambda(a_m, X_k) = \lambda(a_m, X_l)$, $X_k$, $X_l$ are the output signals (conjunctions) on the transitions $h_k$ and $h_l$, respectively. The output alphabet of G-automaton: $Y^G = \{g_1, \dots, g_F\}$, $F = \max_{m=\overline{1,M}} |\chi_m|$ . We put into correspondence to each output symbol $g_f$ of G-automaton a defining conjunction $\hat R^f$ of the set $R^f$ of code variables $r_1, \dots, r_w$ $(w = \mathrm{intlog}_2 F)$. In the example: $F=7$, $w=3$, $Y^G = \{g_1, g_2, g_3, g_4, g_5, g_6, g_7\}$, $R^f\{r_1, r_2, r_3\}$ .

The transition and output functions of G-automaton are defined as follows:

$$(\delta(a_m, X_l) = a_j) \& (\lambda(a_m, X_l) = Y_t) \& (h_l \in \chi_j^m) \Rightarrow$$
$$\Rightarrow (\delta^G(a_m, X_l) = a_j) \& (\lambda^G(a_m, X_l) = g_f).$$

The transition table of the G-automaton is presented as Table 1 (columns 1, 2, 3, 6, and 7).

We will now define the V-automaton $(A^V, X^V, Y^V, \delta^V, \lambda^V, a_1^V)$.

The set of states, the output alphabet, and the initial state of the V-automaton are the same as those of the respective sets of the initial MPA S. The input alphabet of V-automaton coincides with the output alphabet of G-automaton. The functions of V-automaton are defined as follows:

$$(\delta(a_m, X_l) = a_j) \& (\lambda(a_m, X_l) = Y_t) \& (h_l \in \chi_j^m) \Rightarrow$$
$$\Rightarrow (\delta^V(a_m, g_f) = \delta(a_m, X_l) = a_j) \& (\lambda^V(a_m, g_f) = \lambda(a_m, X_l) = Y_t).$$

V-automaton for this example is given in Table 2.

Table 3

| $a_m$ | $a_s$ | $X(a_m, a_s)$ | $Q_s'$ | $R(a_m, a_s)$ | $P(a_m, a_s)$ | |
|---|---|---|---|---|---|---|
| $a_6$ | $b_1$ | $x_3x_4\bar{x}_8$ | — | — | $P_2^1$ | $p_1p_2p_3$ |
| | $b_1$ | $\bar{x}_3x_4$ | — | $r_1$ | $P_3^1$ | $p_1p_3$ |
| | $a_6$ | $x_3x_4x_5\bar{x}_6x_7x_8$ | — | $r_2$ | $P_0$ | — |
| | $b_1$ | $\bar{x}_4\bar{x}_5\bar{x}_7$ | — | $r_2r_1$ | $P_7^1$ | $p_2p_3$ |
| | $b_1$ | $\bar{x}_4x_5x_6$ | — | $r_3$ | $P_9^1$ | $p_1p_2$ |
| | $b_1$ | $\bar{x}_4x_5\bar{x}_6$ | — | $r_3r_1$ | $P_{10}^1$ | $p_1$ |
| | $b_1$ | $\bar{x}_4\bar{x}_6x_7$ | — | $r_3r_1$ | $P_{10}^1$ | $p_1$ |
| | $b_1$ | $\bar{x}_4\bar{x}_5x_6x_7$ | — | $r_3r_2$ | $P_8^1$ | $p_2$ |
| | $b_1$ | $x_3x_4x_6x_8$ | — | $r_3r_2$ | $P_8^1$ | $p_2$ |
| | $b_1$ | $x_3x_4\bar{x}_7x_8$ | — | $r_3r_2$ | $P_8^1$ | $p_2$ |
| | $b_1$ | $x_3x_4\bar{x}_6x_8$ | — | $r_3r_2$ | $P_8^1$ | $p_2$ |
| $b_1$ | $a_6$ | — | $q_1$ | — | $P_0$ | — |
| | $b_1$ | — | $\bar{q}_1$ | — | $P_0$ | — |

For the V-automaton, only the constraint on the number of output variables is violated in the example. The realization of this automaton that is minimal with respect to the number of PLAM bodies is obtained by expanding PLA in terms of outputs [2]. G-automaton, the constraint to the number of input variables is violated.. A method of decompositional synthesis of G-automaton is described below.

## DECOMPOSITION OF G-AUTOMATON

Suppose that on the set of states of G-automaton a partition $\pi = \{A^1, \ldots, A^U\}$ has been adopted (where U is the number of partition blocks. The principles of partition choice are not considered.) Each partition block is assigned uniquely a component automaton of the network, defined as follows:

1) the set of states of the component automaton is the set of states of the respective partition block, plus an additional standby state, maintained by this automaton while the other component automata are working;

2) the transition and output functions of the component automaton coincide with the transition and output functions of the initial automaton if the transitions occur between the states of the same partition block; if the initial state and the final state of the transition belong to different transition blocks, then the component automaton containing the initial state is put on standby and produces the output signal and the connection signal, and the component automaton containing the final state of the transition is caused by the connection signal to pass from the standby state into the final state of the transition, and sends no output signal;

3) each input signal of the component automaton consists of an external input signal and connection input signal;

4) each output signal consists of external output signal and connection output signal.

The decomposition method suggested here uses independent coding of input and output connection signals. The possibility of independent coding, which minimizes the number of external PLAM terminals occupied by connection variables, is achieved by introducing into the automata network a connection automaton (C-automaton); its function is to convert the output connection signals into input connection signals of component automata.

C-automaton is constructed as follows:

1) the states of C-automaton are put into one-to-one correspondence to partition blocks of the initial MPA;

2) the transitions of C-automaton are put into one-to-one correspondence to the transitions of the initial MPA between the states belonging to different partition blocks (for transition between the states of the same partition block, C-automaton does not change its state);

3) the cardinality of the input alphabet of C-automaton is equal to that of the largest alphabet of output connection signals of the component automata;

Table 4

| $a_m$ | $a_s$ | $X(a_m, a_s)$ | $Q_s^2$ | $R(a_m, a_s)$ | $P(a_m, a_s)$ | |
|---|---|---|---|---|---|---|
| $a_1$ | $a_2$ | $\bar{x}_3\bar{x}_4$ | — | — | $P_0$ | — |
| | $a_2$ | $x_1\bar{x}_3$ | — | — | $P_0$ | — |
| | $a_3$ | $\bar{x}_1\bar{x}_3x_4$ | — | $r_1$ | $P_0$ | — |
| | $a_3$ | $x_2\bar{x}_3x_4$ | — | $r_1$ | $P_0$ | — |
| | $b_2$ | $x_2x_3$ | — | $r_2$ | $P_4^2$ | $p_1p_2$ |
| | $b_2$ | $x_2\bar{x}_3\bar{x}_4$ | — | $r_2r_1$ | $P_5^2$ | $p_1$ |
| $a_2$ | $b_2$ | $x_1x_3$ | — | — | $P_6^2$ | $p_2$ |
| | $a_7$ | $\bar{x}_3$ | — | $r_1$ | $P_0$ | — |
| | $a_7$ | $\bar{x}_1$ | — | $r_1$ | $P_0$ | — |
| $a_8$ | $a_8$ | $\bar{x}_1$ | — | — | $P_0$ | — |
| | $a_9$ | $x_1$ | — | $r_1$ | $P_0$ | — |
| $a_9$ | $b_2$ | $x_2$ | — | — | $P_5^2$ | $p_1$ |
| | $b_2$ | $\bar{x}_2$ | — | $r_1$ | $P_6^2$ | $p_2$ |
| $a_{10}$ | $a_8$ | $x_1$ | — | — | $P_0$ | — |
| | $a_9$ | $\bar{x}_1$ | — | $r_1$ | $P_0$ | — |
| $a_7$ | $a_8$ | 1 | — | — | — | — |
| $a_3$ | $b_2$ | 1 | — | — | $P_4^2$ | $p_1p_2$ |
| $b_2$ | $a_1$ | — | $q_2q_3q_4$ | — | $P_0$ | — |
| | $a_2$ | — | $q_2q_3\bar{q}_4$ | — | $P_0$ | — |
| | $a_8$ | — | $q_2\bar{q}_3q_4$ | — | $P_0$ | — |
| | $a_9$ | — | $\bar{q}_2q_3q_4$ | — | $P_0$ | — |
| | $a_{10}$ | — | $\bar{q}_2q_3\bar{q}_4$ | — | $P_0$ | — |
| | $a_7$ | — | $\bar{q}_2\bar{q}_3q_4$ | — | $P_0$ | — |
| | $a_3$ | — | $q_2\bar{q}_3\bar{q}_4$ | — | $P_0$ | — |
| | $b_2$ | — | $\bar{q}_2\bar{q}_3\bar{q}_4$ | — | $P_0$ | — |

4) each output set of C-automaton is a concatenation of the input connection signals of all component automata.

The interaction of component automata in the network occurs as follows. At each point in time, one component automaton is operational. The other component automata are on standby. After passing into the standby state, a component automaton sends a signal to the input of connection automaton. At the same point in the automaton time, the connection automaton produces the signal initiating the operation of some of the component automata of the network. The connection automaton then passes into the state corresponding to that component automaton.

We will now describe the network in formal terms.

In our example, we set: $\pi=\overline{6},\overline{1.2.3.7.8.9.10},\overline{4.5}$ .

We define the component automaton $G^u(B^u, X^u, Y^u, \delta^u, \lambda^u, a_1^u)$ as follows.

1. The set of states of MPA $G^u: B^u = A^u \cup \{b_u\}$ , where $b_u$ is the standby state maintained by $G^u$ during the operation of the other component automata. In the example:

$$B^1 = \{a_6, b_1\};$$

$$B^2 = \{a_1, a_2, a_3, a_7, a_8, a_9, a_{10}, b_2\}; \quad B^3 = \{a_4, a_5, b_3\}.$$

2. The set of input variables $X^u = \{ \bigcup_{a_m \in A^u} X(a_m)\} \cup Q^u$, where $X(a_m)$ is the set of input variables of the initial automaton sampled at transitions from $a_m$; $Q^u$ is the set of additional variables received at the input of $G^u$ from the output of the connection automaton. In the example:

$$X^1 = \{x_3, x_4, x_5, x_6, x_7, x_8, q_1\};$$

$$X^2 = \{x_1, x_2, x_3, x_4, q_2, q_3, q_4\}; \quad X^3 = \{x_2, x_3, x_5, x_6, x_7, q_5, q_6\}.$$

Table 5

| $a_m$ | $a_s$ | $X(a_m, a_s)$ | $Q_s^3$ | $R(a_m, a_s)$ | $P(a_m, a_s)$ | |
|---|---|---|---|---|---|---|
| $a_4$ | $a_6$ | $\bar{x}_5\bar{x}_6x_7$ | — | — | $P_6$ | — |
| | $b_3$ | $x_5x_6$ | — | $r_1$ | $P_6^3$ | $p_1p_3$ |
| | $b_3$ | $\bar{x}_5x_6$ | — | $r_2$ | $P_7^3$ | $p_3$ |
| | $b_3$ | $x_5\bar{x}_6$ | — | $r_2r_1$ | $P_8^3$ | $p_1p_2$ |
| | $b_3$ | $\bar{x}_5\bar{x}_6\bar{x}_7$ | — | $r_2r_1$ | $P_8^3$ | $p_1p_2$ |
| $a_6$ | $b_3$ | $\bar{x}_7\bar{x}_2x_3$ | — | — | $P_6^3$ | $p_1p_3$ |
| | $b_3$ | $x_7x_2$ | — | $r_1$ | $P_7^3$ | $p_3$ |
| | $b_3$ | $\bar{x}_7x_2$ | — | $r_3$ | $P_8^3$ | $p_1p_2$ |
| | $b_3$ | $x_7\bar{x}_2$ | — | $r_2r_1$ | $P_9^3$ | $p_2$ |
| | $b_3$ | $\bar{x}_7\bar{x}_2\bar{x}_3$ | — | $r_3$ | $P_{10}^3$ | $p_1$ |
| $b_3$ | $a_4$ | — | $q_5q_6$ | — | $P_0$ | — |
| | $a_6$ | — | $\bar{q}_5q_6$ | — | $P_0$ | — |
| | $b_3$ | — | $\bar{q}_5\bar{q}_6$ | — | $P_0$ | — |

3. The set of output variables $Y^u = \{\bigcup_{a_m \in A^u} R(a_m)\} \cup P^u$, where $R(a_m)$ is the set of output variables of G, generated at transitions from $a_m$; $P^u$ is the set of additional variables received from the output of $G^u$ at the input of C-automaton. In the example.

$$Y^1 = \{r_1, r_2, r_3, p_1, p_2, p_3\};$$

$$Y^2 = \{r_1, r_2, p_1, p_2\}; \quad Y^3 = \{r_1, r_2, p_1, p_2; p_3\}.$$

4. The transition and output functions:

$$\text{a)} \; (\delta(a_m, X_h) = a_j) \& (\lambda(a_m, X_h) = R_t) \& (a_m \in A^u, a_j \in A^u) \Rightarrow$$
$$\Rightarrow (\delta^u(a_m, X_h) = \delta(a_m, X_h) = a_j) \& (\lambda^u(a_m, X_h) = \lambda(a_m, X_h) = R_t);$$
$$\text{б)} \; (\delta(a_m, X_h) = a_j) \& (\lambda(a_m, X_h) = R_t) \& (a_m \in A^u, a_j \in A^k) \&$$
$$\& (u \neq k) \Rightarrow (\delta^u(a_m, X_h) = b_u) \& (\lambda^u(a_m, X_h) = R_t \cup P_j^u, \; P_j^u \subseteq P^u) \&$$
$$\& (\delta^k(b_k, Q_j^k) = \delta(a_m, X_h) = a_j) \& (\lambda^k(b_k, Q_j^u) = Y_0),$$

where $\hat{Q}_j^k$ is the conjunction of additional input variables (elements of the set $Q^k$) sent to the input of MPA $G^k$ from the output of the connection automaton; and

c) $\delta(b^u, Q_0^u) = b_u, \; \lambda^u(b_u, Q_0^u) = Y_0,$ where $\hat{Q}_0^u$ is the conjunction of the variables of the set $Q^u$, where all of the terms have been inverted.

5. The initial states of the component automata are:

$$(a_1 \in A^u) \Rightarrow (a_1^u = a_1); \quad (a_1 \notin A^u) \Rightarrow (a_1^u = b_u).$$

The transition tables of component automata $G^1$, $G^2$ and $G^3$ are presented as Tables 3, 4, and 5, respectively.

We can define the connection automaton $C(D, P, Q, \delta^c, \lambda^c, d_1^c)$ in formal terms.

1. The elements of the set D of the states of C-automaton are in one-to-one correspondence with the partition blocks of the set of states of the automaton $G(|D| = U)$ being decomposed. In the example:

$$D = \{d_1, d_2, d_3\}.$$

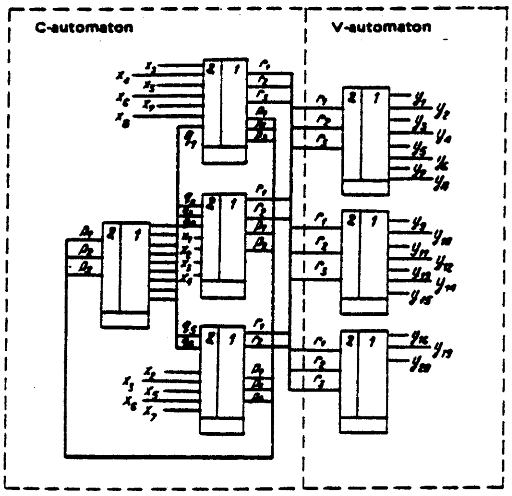
2. The input variables of C-automaton are the additional output variables of the component automata $P = P^l$, $|P^l| = \max_{u=\overline{1,U}} |P^u|, u, l = 1, \ldots, U.$ In the example:

$$P = \{p_1, p_2, p_3\}.$$

3. The output variables of C-automaton are the additional input variables of the

Table 6

| $d_m$ | $d_s$ | $P(d_m, d_s)$ | | $Q(d_m, d_s)$ | |
|---|---|---|---|---|---|
| $d_1$ | $d_2$ | $P_2^1$ | $p_1p_2p_3$ | $Q_2^2$ | $q_2q_3$ |
| | $d_2$ | $P_3^1$ | $p_1\bar{p}_2p_3$ | $Q_3^2$ | $q_3$ |
| | $d_2$ | $P_7^1$ | $\bar{p}_1p_2p_3$ | $Q_7^2$ | $q_4$ |
| | $d_2$ | $P_9^1$ | $p_1p_2\bar{p}_3$ | $Q_9^2$ | $q_3q_4$ |
| | $d_2$ | $P_{10}^1$ | $p_1\bar{p}_2\bar{p}_3$ | $Q_{10}^2$ | $q_3$ |
| | $d_2$ | $P_8^1$ | $\bar{p}_1p_2\bar{p}_3$ | $Q_8^2$ | $q_2q_4$ |
| | $u_1$ | $P_0$ | $\bar{p}_1\bar{p}_2\bar{p}_3$ | $Q_0$ | — |
| $d_2$ | $d_3$ | $P_4^2$ | $p_1p_2$ | $Q_4^3$ | $q_5q_6$ |
| | $d_3$ | $P_5^2$ | $p_1\bar{p}_2$ | $Q_5^3$ | $q_6$ |
| | $d_1$ | $P_6^2$ | $\bar{p}_1p_2$ | $Q_6^1$ | $q_1$ |
| | $d_2$ | $P_0$ | $\bar{p}_1\bar{p}_2$ | $Q_0$ | — |
| $d_3$ | $d_1$ | $P_6^3$ | $p_1\bar{p}_2p_3$ | $Q_6^1$ | $q_1$ |
| | $d_2$ | $P_7^3$ | $\bar{p}_1\bar{p}_2p_3$ | $Q_7^2$ | $q_4$ |
| | $d_2$ | $P_8^3$ | $p_1p_2\bar{p}_3$ | $Q_8^2$ | $q_2q_4$ |
| | $d_2$ | $P_9^3$ | $\bar{p}_1p_2\bar{p}_3$ | $Q_9^2$ | $q_3q_4$ |
| | $d_2$ | $P_{10}^3$ | $p_1\bar{p}_2\bar{p}_3$ | $Q_{10}^2$ | $q_3$ |
| | $d_3$ | $P_0$ | $\bar{p}_1\bar{p}_2\bar{p}_3$ | $Q_0$ | — |




component automata $Q = \bigcup_{u=1}^{U} Q^u$. In the example:

$$Q = \{q_1, q_2, q_3, q_4, q_5, q_6\}.$$

4. The functions of the C-automaton:

a) $(\delta(a_i, X_h) = a_j) \& (\lambda(a_i, X_h) = g_f) \& (a_i, a_j \in A^u) \Rightarrow (\delta^c(d_u, \hat{P}_0) = d_u) \& (\lambda^c(d_u, \hat{P}_0) = Q_0);$

b) $(\delta(a_i, X_h) = a_j) \& (\lambda(a_i, X_h) = g_f) \& (a_i \in A^u) \& (a_j \in A^k) \& (u \neq k) \Rightarrow (\delta^c(d_u, \hat{P}_j^u) = d_k) \& (\lambda^c(d_u, \hat{P}_j^u) = Q_j^k),$

where $\hat{P}_j^u$ is the conjunction of additional output variables of MPA $G^u$ (the elements of the set $P^u$); $\hat{P}_0$ is the conjunction of additional output variables corresponding to the zero set.

5. The initial state of the C-automaton: $(a_1 \in A^u) \Rightarrow d_1^c = d_u$ . In the example:

$$d_1^c = d_1.$$

The transition table of C-automaton for this example is given as Table 6.

This completes the construction of the network.

The initial MPA S has, thus, been partitioned into a network of five interacting automata: $G^1$, $G^2$, $G^3$, C and V. The variables of the first four of these automata satisfy the given constraints; therefore, each of these automata can be realized on the basis of a single PLAM. The V-automaton is realized by expanding PLAM in respect of outputs. The schematic realization of the network on the basis of PLAM (7, 8, 32, and 6) is shown in the figure.

The methods of decomposition described here have been realized in a software package for US [Unified System] computers and included into the Universal Automated System for Synthesis of Microprogramming Automata (UASSMA-US) [3] under the title "MPA Synthesis on the PLA Basis."

The methods suggested in the paper make it possible to:

a) use PLA memory to reduce the number of terminals occupied by connection signals; and

b) expand the class of automata which can be realized on this basis for given PLAM parameters.

In conclusion, it may be noted that these interaction models of automata could be implemented in different circuit configurations. For example, one could use a different method of coding output signals of connection automaton.

## REFERENCES


1. V. A. Sklyarov, "Synthesis of automata on the basis of circuits with a matrix structure," Upravlyayushchie Sistemy i Mashiny, no. 1, pp. 66-70, 1982.
2. S. I. Baranov and V. N. Sinev, Automata and Programmable Matrices [in Russian], Vysheishaya Shkola, Minsk, 1980.
3. S. I. Baranov and L. N. Zhuravina, "Universal automated system for microprogramming automata synthesis (UASSMA-US)," Upravlyayushchie Sistemy i Mashiny, no. 4, pp. 85-86, 1980.


20 April 1985