Int. J. Appl. Math. Comput. Sci., 2005, Vol. 15, No. 1, 151–158

DESIGN OF MEALY FINITE–STATE MACHINES WITH THE TRANSFORMATION OF OBJECT CODES

ALEXANDER A. BARKALOV*, ALEXANDER A. BARKALOV, Jr.**

* Institute of Computer Engineering and Electronics
University of Zielona Góra
ul. Podgórna 50, 65–246 Zielona Góra, Poland
e-mail: a.barkalov@iie.uz.zgora.pl

** Siemens-Ukraine
ul. Predslevskaja 11, 03–150 Kiev, Ukraine
e-mail: alexander.barkalov@siemens.com

An optimization method of the logic circuit of a Mealy finite-state machine is proposed. It is based on the transformation of object codes. The objects of the Mealy FSM are internal states and sets of microoperations. The main idea is to express the states as some functions of sets of microoperations (internal states) and tags. The application of this method is connected with the use of a special code converter in the logic circuit of an FSM. An example of application is given. The effectiveness of the proposed method is also studied.

Keywords: finite-state machine, programmable logic device, object, design, logic circuit

1. Introduction

The control unit of any digital system can be implemented as a Mealy finite-state machine (FSM) (Baranov, 1994). Nowadays programmable logic devices (PLDs) are widely used to implement the logic circuits of FSMs (Solovjev, 1996; 2001). This class of VLSI (PLA, PAL, CPLD, FPGA) evokes high cost that leads to high cost of FSM circuits. One of the main problems in the design of such circuits is to find a compromise between the price and the performance of a device (Barkalov, 2002). A single-level circuit of a Mealy FSM (Fig. 1) is the fastest but the most expensive solution to this problem.

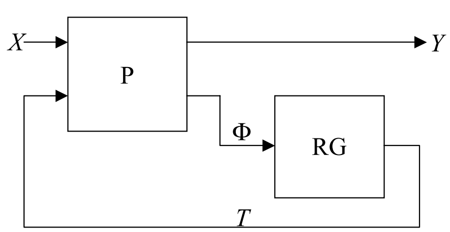

Fig. 1. Structural diagram of a single-level circuit of a Mealy FSM.

Here the circuit P implements the systems

$$Y = Y(T, X), \tag{1}$$

$$\Phi = \Phi(T, X), \tag{2}$$

where $Y = \{y_1, \ldots, y_N\}$ is a set of microoperations, $X = \{x_1, \ldots, x_L\}$ is a set of logic conditions, $T = \{T_1, \ldots, T_R\}$ is a set of internal variables to encode the states $a_m \in A$, $A = \{a_1, \ldots, a_M\}$ is a set of internal states, $R =]\log_2 M]$, and $\Phi = \{\phi_1, \ldots, \phi_R\}$ is a set of excitation functions. The register RG keeps the codes $K(a_m)$ of the internal states $a_m \in A$. Denote this structure by P FSM.

The subcircuit P of the P FSM has a maximal possible number of outputs $t(P) = N + R$. To minimize the cost of the P FSM, we can use various algorithmic methods (Ahmad and Dhodhi, 2000; Kania, 2003; Lahtinen *et al.*, 2002; Singh and Nowick, 2000).

If the performance is not a critical issue, then the minimization of the cost can be achieved by increasing the number of levels in the FSM circuit (Barkalov, 2002). One of the possible solutions is the application of either maximal encoding of microoperations sets (Barkalov and Palagin, 1997) or encoding the fields of compatible microoperations (Barkalov, 2003). Both methods lead to a two-level circuit (Fig. 2) that will be denoted by PY FSM.

Here the circuit P implements the system (2) and the system

$$Z = Z(T, X), \tag{3}$$

where $Z = \{z_1, \ldots, z_G\}$ is the set of variables to encode the sets of microoperations $Y_q \subseteq Y$. The value of the

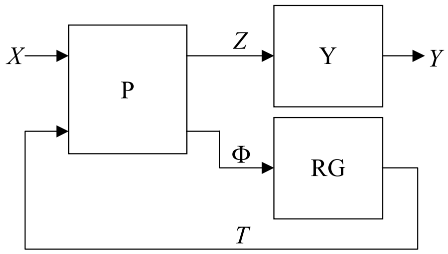

Fig. 2. Structural diagram of a two-level circuit of a Mealy FSM.

parameter G depends on the method of encoding microoperations. The circuit Y implements a system

$$Y = Y\left(Z\right). \tag{4}$$

In the case of maximal encoding of microoperations, the circuit Y is implemented using PROMs and $G =]\log_2 Q[$, where Q is the amount of different sets of microoperations in the initial flow chart (Barkalov and Palagin, 1997). In the case of encoding the fields of compatible microoperations, the circuit Y is a collection of I decoders $DC_1, \dots, DC_I$. Here I is the amount of the fields of compatible microoperations (Barkalov, 2003) and $G = G_1 + G_2 + \dots + G_I$, where G_i is the number of the variables to encode the microoperations forming the i-th field $(i = 1, \dots, I)$. It is clear that the circuit P of a PY FSM has $t(PY) = G + R < t(P)$ outputs.

It was shown (Solovjev, 2001) that the price of the FSM circuit directly depends on the number of the required outputs of the circuit P. The fewer outputs for a fixed number of inputs, the lower the price of the circuit. In this paper we propose methods of minimizing the number of the outputs of the circuit P based on the transformation of the codes of objects. In this case the internal states of the FSM and sets of microoperations are the objects to be transformed. The application of the proposed method is most effective if the circuit P is implemented using PLA, and a less notable gain can be expected in the case of FPGA and CPLD implementations.

2. Main Idea of the Method

Let a Mealy FSM be represented by a direct structural table (DST) with columns (Baranov, 1994). Here a_m is the initial state of the FSM, $K(a_m)$ is the code of the state $a_m \in A$, a_s is the state of the transition, $K(a_s)$ is the code of the state $a_s \in A$, X_h constitutes an input signal causing the transition $\langle a_m, a_s \rangle$ and it is equal to the conjunction of some variables $x_e \in X$, Y_h is an output signal for the transition $\langle a_m, a_s \rangle$, $Y_h \subseteq Y$, Φ_h is the set of excitation functions that are equal to 1 to switch the register RG from $K(a_m)$ to $K(a_s)$, $\Phi_h \subseteq \Phi$, $h = 1, \dots, H$ is the number of DST lines. Each line of the DST corresponds to one term F_h of the disjunction normal form of the functions (1) and (2):

$$F_h = A_m^h X_h, \quad h = 1, \dots, H. \tag{5}$$

Here A_m^h is a conjunction of the internal variables $T_r \in T$, corresponding to a code of the state $a_m \in A$ from the h-th line of the DST,

$$A_m = \bigwedge_{r=1}^{R} T_r^{l_{mr}}, \quad m = 1, \dots, M, \tag{6}$$

where $l_{mr} \in \{0, 1\}$ is the value of the r-th bit of the code $K(a_m)$, $T_r^0 = \overline{T_r}$, $T_r^1 = T_r$, $r = 1, \dots, R$.

The functions (1) and (2) are represented as

$$y_n = \bigvee_{h=1}^{H} C_{nh} F_h, \quad n = 1, \dots, N, \tag{7}$$

$$\varphi_r = \bigvee_{h=1}^{H} C_{rh} F_h, \quad n = 1, \dots, N, \tag{8}$$

where $C_{nh}(C_{rh})$ is the Boolean variable that is equal to 1 if a function $y_n(\varphi_r)$ is written in the h-th row of the DST.

The Mealy FSM S_1 (Table 1) has the following characteristics: $A = \{a_1, \dots, a_5\}$, $X = \{x_1, \dots, x_4\}$, $Y = \{y_1, \dots, y_7\}$, $M = 5$, $R = 3$, $L = 4$, $N = 7$ and $H = 12$.

Table 1. Direct structural table of the Mealy FSM S_1.

a_m	$K(a_m)$	a_s	$K(a_s)$	X_h	Y_h	Φ_h	h
a_1	000	a_2	010	x_1	$y_1\ y_2$	D_2	1
		a_3	011	$\overline{x_1}$	y_3	$D_2\ D_3$	2
a_2	010	a_2	010	x_2	$y_1\ y_2$	D_2	3
		a_3	011	$\overline{x_2}\ x_3$	y_4	$D_2\ D_3$	4
		a_4	100	$\overline{x_2}\ \overline{x_3}$	$y_1\ y_2$	D_1	5
a_3	011	a_4	100	x_1	$y_2\ y_5$	D_1	6
		a_5	101	$\overline{x_1}$	y_6	$D_1\ D_3$	7
a_4	100	a_5	101	1	$y_3\ y_7$	$D_1\ D_3$	8
a_5	101	a_2	010	$x_2\ x_3$	$y_1\ y_2$	D_2	9
		a_3	011	$x_2\ \overline{x_3}$	y_3	$D_2\ D_3$	10
		a_5	101	$\overline{x_2}\ x_4$	$y_3\ y_7$	$D_1\ D_3$	11
		a_1	000	$\overline{x_2}\ x_4$	—	—	12

The memory of the FSM S_1 is implemented as a register with information inputs of the D type that corresponds to a practical case (Solovjev, 1996). From Table 1, for example, we can form $F_1 = \overline{T_1}\ \overline{T_2}\ \overline{T_3} x_1$, $y_7 = F_8 \vee F_{11}$, $D_1 = F_5 \vee F_6 \vee F_7 \vee F_8 \vee F_{11}$.

Without no loss of generality, we restrict ourselves only to the method of maximal encoding of the sets of microoperations. Let us encode each set $Y_q \subseteq Y$ by a binary code $K(Y_q)$ and form a set $Z = \{z_1, \ldots, z_G\}$.

In the case of the FSM S_1 there are $Q = 7$ different sets of microoperations in Table 1: $Y_1 = \emptyset$, $Y_2 = \{y_1, y_2\}$, $Y_3 = \{y_3\}$, $Y_4 = \{y_4\}$, $Y_5 = \{y_2, y_5\}$, $Y_6 = \{y_6\}$, $Y_7 = \{y_3, y_7\}$. Therefore $G = 3$ and $Z = \{z_1, z_2, z_3\}$. Let $K(Y_1) = 000$, $K(Y_2) = 001$, $\ldots$, $K(Y_7) = 110$. Form the transformed DST of the Mealy FSM S_1 by replacing the column Y_h with the column Z_h which contains the variables $z_g \in Z$ that are equal to 1 in the code $K(Y_q)$ of the set of microoperations $Y_q \subseteq Y$ for each line of the DST. The resulting table (Table 2) is a base for the implementation of the PY FSM. From Table 2 we can form the system (3), e.g. $z_1 = F_6 \vee F_7 \vee F_8 \vee F_{11}$.

Table 2. Transformed DST of the Mealy FSM S_1.

a_m	$K(a_m)$	a_s	$K(a_s)$	Z_h	Y_h	Φ_h	h
a_1	000	a_2	010	x_1	z_3	D_2	1
		a_3	011	$\overline{x_1}$	z_2	$D_2\ D_3$	2
a_2	010	a_2	010	x_2	z_3	D_2	3
		a_3	011	$\overline{x_2}\ \overline{x_3}$	$z_2\ z_3$	$D_2\ D_3$	4
		a_4	100	$\overline{x_2}\ \overline{x_3}$	z_3	D_1	5
a_3	011	a_4	100	x_1	z_1	D_1	6
		a_5	101	$\overline{x_1}$	$z_1\ z_3$	$D_1\ D_3$	7
a_4	100	a_5	101	1	$z_1\ z_2$	$D_1\ D_3$	8
a_5	101	a_2	010	$x_2\ x_3$	z_3	D_2	9
		a_3	011	$x_2\ \overline{x_3}$	z_2	$D_2\ D_3$	10
		a_5	101	$\overline{x_2}\ x_4$	$z_1\ z_2$	$D_1\ D_3$	11
		a_1	000	$\overline{x_2}\ x_4$	—	—	12

Let us call the states of the FSM $a_m \in A$ (resp. the sets of microoperations $Y_q \subseteq Y$) the objects of the first (resp. second) kind. Let us represent the objects of one kind as functions of the objects of the other kind.

Let us represent the states of the transition as some functions of the sets of microoperations. From the analysis of Table 2 it is clear that there is no one-to-one correspondence between the states and the sets of microoperations. For example, the set $Y_1 \subseteq Y$ corresponds to the states a_2 (rows 1, 3, 9) and a_4 (row 5). Therefore, we need some labels to express the states of transitions (excitation functions) as functions of the sets of microoperations. Let $I = \{I_1, \ldots, I_K\}$ be a set of labels (we shall discuss later how to determine the parameter K). In this case we can represent the states $a_m \in A$ as functions:

$$A = A(Z, I). \tag{9}$$

If the tags $I_k \in I$ are encoded by the binary codes $K(I_k)$ using the variables $v_b \in V = \{v_1, \ldots, v_B\}$, where $B =]\log_2 K[$, then the excitation functions $\varphi_r \in \Phi$ can be represented as

$$\varphi_r = \varphi_r(Z, V), \quad r = 1, \ldots, R. \tag{10}$$

This leads to a Mealy FSM of the first kind or the $\mathrm{P_Y A}$ Mealy FSM (Fig. 3). Here the circuit P forms the functions (3) and the system

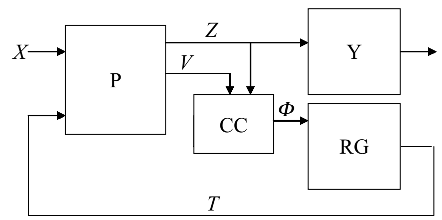

Fig. 3. Structural diagram of a $\mathrm{P_Y A}$ Mealy FSM.

$$V = V(T, X). \tag{11}$$

The systems (3) and (11) depend on the terms (5). The code converter CC implements the system (10) and the circuit Y forms the functions (4).

Let us represent the sets of microoperations $Y_q \subseteq Y$ as some functions of the states $a_m \in A$. Because a one-to-one correspondence between these objects usually does not exist, we need some tags $I_k \in I$ for a desirable representation. In this case we can represent the sets of microoperations $Y_q \subseteq Y$ as functions:

$$Y = Y(A, I). \tag{12}$$

The system (12) can be transformed into the form

$$y_n = y_n(T, V), \quad n = 1, \ldots, N, \tag{13}$$

where the variables $v_b \in V$ are used to encode the tags $I_k \in I$. This leads to a Mealy FSM of the second kind, or a $\mathrm{P_A Y}$ Mealy FSM (Fig. 4).

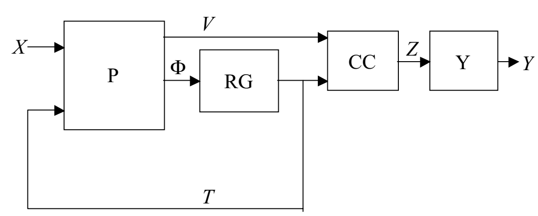

Fig. 4. Structural diagram of a $\mathrm{P_A Y}$ Mealy FSM.

Here the circuit P forms the functions (2) and (11), the circuit Y forms the functions (4), and the code converter CC forms the functions

$$Z = Z(T, V). \tag{14}$$

It is clear that here the system (13) is transformed into the system

$$y_n = y_n(Z(T,V)), \quad n = 1,\dots,N. \tag{15}$$

This means that the cycle time of the P_AY Mealy FSM is greater than that of the equivalent P_YA Mealy FSM and the difference is determined by the propagation time of the code converter CC.

Such an approach permits to reduce the number of the outputs of the circuit P to the value $t(P_YA) = G + B$ in the case of a P_YA FSM, or to the value $t(P_AY) = R + B$ in the case of a P_AY FSM. This permits to decrease the cost of the circuit P in comparison with the equivalent PY FSM. Of course, such an approach makes sense if the total cost of the circuits P and CC is lower then the cost of the circuit P of the PY Mealy FSM.

3. Design of Mealy FSMs of the First and Second Kinds

Let a Mealy FSM be represented by a transformed DST of a PY Mealy FSM. The proposed method of P_YA Mealy FSM design includes the following steps:

1. *Identification of states.* Let $A(Y_q)$ be a set of states such as the set $Y_q \subseteq Y$ is formed under the transition in the state $a_m \in A(Y_q)$. It is sufficient to employ $m_q = |A(Y_q)|$ tags for the states $a_m \in A(Y_q)$. It also suffices to use $K = \max(m_1,\dots,m_Q)$ tags for the determination of any state $a_m \in A$, and they form a set I. Let us encode each tag $I_k \in I$ by a binary code $K(I_k)$ with $B =]\log_2 K[$ bits and let us form the set $V = \{v_1,\dots,v_B\}$. Let the pair $\alpha_{qs} = \langle I_k, Y_q\rangle$ correspond to the state $a_s \in A(Y_q)$. In this case the code $C(a_s)^q$ corresponds in a one-to-one manner to the concatenation

$$C(a_s)^q = K(Y_q) * K(I_k), \tag{16}$$

where '*' denotes the concatenation operator.

2. *Formation of the table for the $P_Y A$ FSM.* This table is a base to form the systems (3) and (11). It is constructed by the replacement of the columns a_s, $K(a_s)$, Φ_h of the transformed DST with the column V_h. The column V_h contains the variables $v_r \in V$ that are equal to 1 in the code of the label for the state $a_s \in A$ from the h-th line of the DST.

3. *Formation of the table for the code converter.* This table is a base for the formation of the system (10). It contains the columns Y_q, $K(Y_q)$, I_k, $K(I_k)$, a_s, $K(a_s)$, Φ_h and h. This table contains $H_0 = m_1 + \dots + m_Q$ rows and each row corresponds to one pair α_{qs}, $q = 1,\dots,Q$ and $s = 1,\dots,M$. The system (10) is represented as

$$\varphi_r = \bigvee_{h=1}^{H_0} C_{rh} V_h Z_h, \quad r = 1,\dots,R, \tag{17}$$

where C_{rh} is the Boolean variable that is equal to 1 if the function $\varphi_r \in \Phi$ is written in the h-th row of the table of the CC, V_h is a conjunction of the variables $v_b \in V$ corresponding to the code $K(I_k)$ from the h-th line of this table, Z_h is a conjunction of the variables $z_r \in Z$ corresponding to the code $K(Y_q)$ from the h-th row of the table of the CC, $h = 1,\dots,H_0$.

4. *Formation of the table of microoperations.* This table is a base to form the system (4). It contains the columns Y_q, $K(Y_q)$, $y_1,\dots,y_N$ and q. Each line of this table corresponds to one set of microoperations $Y_q \subseteq Y$.

5. *Design of the logic circuit of the FSM.* The circuit of the P_YA FSM is designed using the system (3) and (11) for the circuit P, the system (17) for the code converter CC and the system (4) for the circuit Y. Here the circuit P and the CC are implemented using PLD, the circuit Y is implemented using PROM because the system (15) includes more than 50% of possible terms (Barkalov, 2002).

The proposed method of the P_AY Mealy FSM design is very similar to the previous one and includes the following steps:

1. *Identification of the sets of microoperations.* Let $Y(a_s)$ be a collection of the sets of microoperations such as $Y_q \in Y(a_s)$ if the set $Y_q \subseteq Y$ is formed under the transition in the state $a_s \in A$. It is sufficient to employ $n_s = |Y(a_s)|$ tags for the sets $Y_q \in Y(a_s)$. To identify any set $Y_q \subseteq Y$ it is enough to use $K = \max(n_1,\dots,n_M)$ labels $I_k \in I$. Let us encode each tag $I_k \in I$ by a binary code $K(I_k)$ with $B =]\log_2 K[$ and form a set $V = \{v_1,\dots,v_B\}$ to encode the tags. Let the pair $\beta_{sq} = \langle I_k, a_s\rangle$ with a unique tag $I_k \in I$ correspond to a set $Y_q \subseteq Y$. Now a code $C(Y_q)$ of the set $Y_q \subseteq Y$ can be expressed as

$$C(Y_q) = K(I_k) * K(a_s). \tag{18}$$

2. *Formation of the table for the $P_A Y$ FSM.* This table constitutes a base to form the functions (2) and (11). It is constructed by the replacement of the column Z_h of the transformed DST with the column V_h. This column contains the variables $v_b \in V$ that are equal to 1 in the code $K(I_k)$ corresponding to the expression (18) for the set $Y_q \subseteq Y$ from the h-th row of the transformed DST, $h = 1,\dots,H$.

3. *Formation of the table for the code converter.* This table constitutes a base for the formation of the system (14). It contains the columns a_s, $K(a_s)$, I_k,

$K(I_k)$, Y_q, Z_q and q. All pairs β_{sq} are written in this table for the sets $Y_q \subseteq Y$. The table of the code converter has $Q_0 = n_1 + \cdots + n_M$ rows and the system (14) is represented as

$$z_g = \bigvee_{q=1}^{Q_0} C_{gq} V_q A_q, \quad g = 1, \ldots, G. \tag{19}$$

Here C_{gq} is the Boolean variable that is equal to 1 if the variable $z_g \in Z$ is written in the q-th line of the table of the CC, $q = 1, \ldots, Q_0$, V_q is a conjunction of the variables $v_b \in V$ corresponding to the code $K(I_k)$ from the q-th row of the table, A_q is a conjunction of the internal variables $T_r \in T$ corresponding to the state $a_s \in A$ from the q-th line of the table, $q = 1, \ldots, Q_0$.

4. *Formation of the table of microoperations.* This step is executed in the same manner as for the P_YA Mealy FSM.
5. *Design of the logic circuit for the FSM.* The circuit P is implemented on PLD using the systems (2) and (12). The circuit Y is implemented on PROM using the system (4). The circuit CC is implemented on PLD using the system (19). The problems connected with the design of similar circuits are well known and can be found in (Solovjev, 1996; 2001). These problems are beyond the scope of our paper.

4. Example of the Application of the Proposed Method

The design methods of P_YA and P_AY FSMs are very similar. Taking this into account, we shall discuss only an example regarding the design of a P_YA Mealy FSM S_1 and we shall start from the transformed table (Table 2).

Let us form the sets $A(Y_q)$: $A(Y_1) = \{a_1\}$, $A(Y_2) = \{a_2, a_4\}$, $A(Y_3) = \{a_3\}$, $A(Y_4) = \{a_3\}$, $A(Y_5) = \{a_4\}$, $A(Y_6) = \{a_5\}$, $A(Y_7) = \{a_5\}$. Therefore $m_1 = m_3 = \cdots = m_7 = 1$, $m_2 = 2$, $K = 2$, $I = \{I_1, I_2\}$, $B = 1$, $V = \{v_1\}$. Let $K(I_1) = 0$, $K(I_2) = 1$. Let us form the pairs α_{qs} for all elements of the sets $A(Y_q) \subseteq A$. If $m_q = 1$, then the first component of the corresponding pair α_{qs} is written as $\emptyset$. This implies the code $K(I_k) = *$ for such a pair. In our case there are the following pairs: $\alpha_{11} = \langle \emptyset, Y_1 \rangle$, $\alpha_{22} = \langle I_1, Y_2 \rangle$, $\alpha_{24} = \langle I_2, Y_2 \rangle$, $\alpha_{33} = \langle \emptyset, Y_3 \rangle$, $\alpha_{43} = \langle \emptyset, Y_4 \rangle$, $\alpha_{54} = \langle \emptyset, Y_5 \rangle$, $\alpha_{65} = \langle \emptyset, Y_6 \rangle$ and $\alpha_{75} = \langle \emptyset, Y_7 \rangle$. The codes $C(a_s)^q$ corresponding to (16) are shown in Table 3.

Table 3 has $H_0 = 8$ rows. The first three positions of the column $C(a_s)^q$ contain the code $K(Y_q)$, while the last position corresponds to the code $K(I_k)$.

Table 4 represents the table of the P_YA FSM S_1 that is constructed from the transformed table of the FSM S_1 (Table 2). The symbol '-' in the column V_h means that $v_1 = 0$, while the symbol '*' means that v_1 can have any value (the 'don't care' situation). This table is used to form the system (3) and (11), for example, $v_1 = F_5$.

Table 5 corresponds to the code converter that is constructed using the encoding table for the states (Table 3)

Table 3. Encoding the state of the P_YA FSM S_1.

a_s	$C(a_s)^q$	α_{qs}	h
a_1	000*	α_{11}	1
a_2	0010	α_{22}	2
a_3	010*	α_{33}	3
a_3	011*	α_{43}	4
a_4	100*	α_{54}	5
a_4	0011	α_{24}	6
a_5	101*	α_{65}	7
a_5	110*	α_{75}	8

Table 4. Settings for the P_YA Mealy FSM S_1.

a_m	$K(a_m)$	X_h	Z_h	V_h	h
a_1	000	x_1	z_3	-	1
		$\overline{x_1}$	z_2	*	2
a_2	010	x_2	z_3	*	3
		$\overline{x_2}\,\overline{x_3}$	$z_2\ z_3$	*	4
		$\overline{x_2}\,\overline{x_3}$	z_3	v_1	5
a_3	011	x_1	z_1	*	6
		$\overline{x_1}$	$z_1 z_3$	*	7
a_4	100	1	$z_1 z_2$	*	8
a_5	101	$x_2 x_3$	z_3	-	9
		$x_2 \overline{x_3}$	z_2	*	10
		$\overline{x_2} x_4$	$z_1 z_2$	*	11
		$\overline{x_2} x_4$	-	*	12

Table 5. Settings for the code converter of the P_YA FSM S_1.

Y_q	$K(Y_q)$	I_k	$K(I_k)$	a_s	$K(a_s)$	Φ_h	h
Y_1	000	-	*	a_1	000	-	1
Y_2	001	I_1	0	a_2	010	D_2	2
		I_2	1	a_4	100	D_1	3
Y_3	010	-	*	a_3	011	$D_2 D_3$	4
Y_4	010	-	*	a_3	011	$D_2 D_3$	5
Y_5	011	-	*	a_4	100	D_1	6
Y_6	100	-	*	a_5	101	$D_1 D_3$	7
Y_7	101	-	*	a_5	101	$D_1 D_3$	8

and the transformed table of the FSM S_1 (Table 2). This table is used to form the system (17), where the conjunction V_h is determined as

$$V_h = v_1^{l_{h1}} \dots v_B^{l_{hB}}, \quad h = 1, \dots, H_0.$$

Here $l_{hb} \in \{0, 1, *\}$ is the value of the b-th bit of the code $K(I_k)$ from the h-th row of the table of the CC, $v_b^0 = \overline{v_b},\ v_b^1 = v_b,\ v_b^* = 1,\ b = 1, \dots, B$.

From this table we can form, for example, the function

$$D_2 = \overline{z_1}\ \overline{z_2} z_3 \overline{v_1} \vee \overline{z_1} z_2 \overline{z_3} \vee \overline{z}_1 z_2 z_3.$$

The table of microoperations describes the PROM with the inputs Z and the outputs Y (Table 6).

The circuit of the $\mathrm{P_Y A}$ Mealy FSM S_1 includes

- the circuit P with $S = L + R = 7$ inputs, $t = G + B = 4$ outputs and $H = 12$ terms;
- the circuit CC with $S = G + B = 4$ inputs, $t = R = 3$ outputs and $H_0 = 8$ terms;
- the circuit Y with $S = G = 3$ inputs, $t = N = 7$ outputs and $Q = 7$ terms.

Table 6. Microoperations of the $\mathrm{P_Y A}$ FSM S_1.

Y_q	$K(Y_q)$	y_1	y_2	y_3	y_4	y_5	y_6	y_7	q
Y_1	000	0	0	0	0	0	0	0	1
Y_2	001	1	1	0	0	0	0	0	2
Y_3	010	0	0	1	0	0	0	0	3
Y_4	010	0	0	0	1	0	0	0	4
Y_5	011	0	1	0	0	1	0	0	5
Y_6	100	0	0	0	0	0	1	0	6
Y_7	101	0	0	1	0	0	0	1	7

5. Analysis of the Proposed Method

Let us find an area where $\mathrm{P_Y A}$ Mealy FSMs have lower cost than the equivalent PY Mealy FSMs. Let us use the probabilistic approach suggested by Novikov (1974) and developed in (Barkalov, 2002). There are three key points in such an approach:

1. The use of a class of flow charts instead of a particular flow chart. This class is characterized by the parameters

$$p_1 = \frac{O(\Gamma)}{O(\Gamma) + C(\Gamma) + 2}, \quad p_2 = \frac{C(\Gamma)}{O(\Gamma) + C(\Gamma) + 2}, \tag{20}$$

where $O(\Gamma)$ is the number of the operational nodes in the flow chart Γ, $C(\Gamma)$ is the number of the conditional nodes in the flow chart Γ, number 2 being added to take account of the existence of the start and end nodes in the flow chart. It is clear that

$$\lim_{K(\Gamma)\to\infty} (p_1 + p_2) = 1, \tag{21}$$

where $K(\Gamma) = O(\Gamma) + C(\Gamma)$. Therefore p_1 (resp. p_2) can be treated as the probability of the event that a particular node of the flow chart Γ is an operational (resp. conditional) one.

2. The use of matrix realization for the circuit of the control unit (Baranov, 1994) instead of the standard VLSI. In this case we can determine a hardware amount as the volume of matrices for a given circuit of the control unit.

3. The study of the relation $S(U_1)/S(U_2)$, where $S(U_i)$ is the volume of matrices for the implementation of the circuit of the control unit U_i, $i = 1, 2$. In (Barkalov, 2002) it was proved that such relations for the cases of matrix realization and implementation of the circuit of the FSM using the standard PLD have the same values.

A matrix realization of a PY Mealy FSM is shown in Fig. 5.

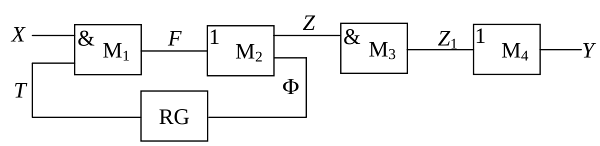

Fig. 5. Matrix realization of a PY Mealy FSM.

Here M_1 is a conjunctional matrix that implements the system F of the terms (5). M_2 is a disjunctional matrix that implements the systems (3) and (8). M_3 is a conjunctional matrix that implements the terms Z_1 of the system (4). M_4 is a disjunctional matrix that implements the system (4). Therefore the matrices M_1 and M_2 represent the circuit P and the matrices M_3 and M_4 represent the circuit Y. The complexity of these circuits can be expressed as

$$S(P) = 2(L + R)H + H(G + R), \tag{22}$$

$$S(Y) = 2GQ + QN. \tag{23}$$

A matrix realization of this $\mathrm{P_Y A}$ Mealy FSM is shown in Fig. 6.

Here the circuit P is realized by the matrices M_1 and M_2, the circuit Y is realized by the matrices M_3 and M_4, the circuit CC is realized by the matrices M_5 and M_6, where F_1 is a set of the terms of functions (10). The complexity of the circuits P and CC can be expressed as

$$S(P)^1 = 2(L + R)H + H(G + B), \tag{24}$$

$$S(CC)^1 = 2(G + B)H_0 + H_0 R. \tag{25}$$

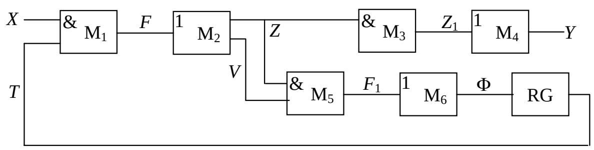

Fig. 6. Matrix realization of a $\mathrm{P_YA}$ Mealy FSM.

Here index 1 means that the circuits P and CC are parts of the $\mathrm{P_YA}$ Mealy FSM. It is clear that $S(Y)^1 = S(Y)$.

To find the area of an effective application of $\mathrm{P_YA}$ Mealy FSMs, we should examine the function

$$f = \frac{S(P)^1 + S(\mathrm{CC})^1 + S(Y)^1}{S(P) + S(Y)}. \qquad (26)$$

If $f < 1$, then the cost of the logic circuit of the $\mathrm{P_YA}$ FSM is lower than that of the equivalent PY FSM.

To reduce the number of variables in (22)–(26), we can use the results of (Barkalov, 2002), where the parameters L, R, H and Q are expressed as functions of $K(\Gamma)$ and some coefficients:

$$H = \frac{4.44 + 1.44 p_1 K(\Gamma)}{p_3}, \qquad (27)$$

$$M = \frac{3.55 + 0.44 p_1 K(\Gamma)}{p_3}, \qquad (28)$$

$$L = \frac{(1 - p_1) K(\Gamma)}{p_4}, \qquad (29)$$

$$Q = \frac{p_1 K(\Gamma)}{p_3}. \qquad (30)$$

Here $p_3 = O(\Gamma)/Q$, $p_4 = C(\Gamma)/L$, $1 \leq p_3$, $p_4 \leq 1.3$ (Barkalov, 2002).

Let us use the coefficients $p_5 = H/H_0$, $p_6 = B/R$, $p_7 = C_R/C_P$, where C_R is the cost of PROM, C_P is the cost of PLD with the same number of inputs and outputs. Now the function (26) can be expressed as the function

$$f = f(K(\Gamma), N, p_1, p_3, \ldots, p_7). \qquad (31)$$

Some results of investigation are shown in Figs. 7 and 8.

From the analysis of these figures it is clear that the $\mathrm{P_YA}$ Mealy FSMs always offers gains in the cost compared with PY FSMs. This gain is increased with reducing the number of the nodes of the initial flow chart (resp. decreasing the parameter $K(\Gamma)$) and decreasing the length of the codes of the sets of microoperations in the initial flow chart (resp. increasing the parameter p_3). The maximal gain from 9% to 13% is achieved for flow charts with the number of nodes $100 \leq K(\Gamma) \leq 200$.

The same results were obtained for a comparison of $\mathrm{P_YA}$ and PY Mealy FSMs. But here the gain was up to

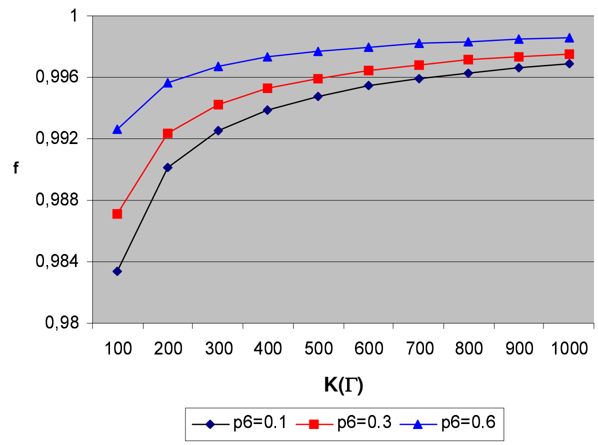

Fig. 7. Function f for $p_1 = 0.1$ and $p_3 = 1$.

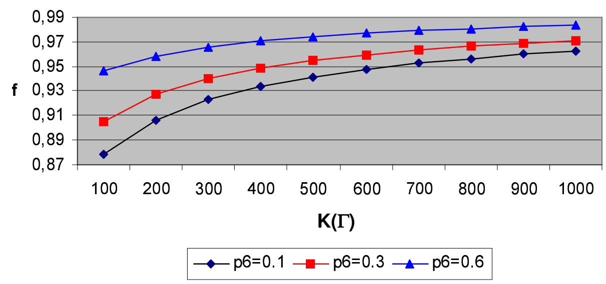

Fig. 8. Function f for $p_1 = 0.9$ and $p_3 = 1.2$.

16% and it was changed in a similar way as the function f. It should be pointed out that the application of the well-known methods of state encoding (Devadas and Newton, 1991), and algorithms and tools such as KISS, NOVA, MUSTANG, JEDI, MUSE, MIS, SIS (De Micheli, 1994) can increase this gain, but this issue requires a separate study and is beyond the scope of this paper.

6. Conclusion

The proposed methods of the implementation of Mealy FSMs with the transformation of the codes of objects allow reducing the cost of the logic circuit of the control unit in comparison with two-level circuits based on maximal encoding of the sets of microoperations (PY Mealy FSMs). In this article, two kinds of the FSM have been proposed:

1. $\mathrm{P_YA}$ Mealy FSMs based on the transformation of the codes of the sets of microoperations into the codes of internal states.

2. $\mathrm{P_YA}$ Mealy FSMs based on the transformation of the codes of the internal states into the codes of the sets of microoperations.

The analysis of the effectiveness of the proposed methods has shown that the proposed circuits always have lower cost than the equivalent PY Mealy FSMs. It was shown that P_YA Mealy FSMs have lower cost than the equivalent P_AY Mealy FSMs, but the latter have shorter times of cycle because of the sequential path "P–CC–Y" in P_YA Mealy FSMs. Therefore, P_AY Mealy FSMs can be used if the criterion of design effectiveness is its maximal cost, P_YA Mealy FSMs can be used if P_AY Mealy FSMs cannot allow reaching the desirable performance. A similar approach (the transformation of the object codes) can be applied to the optimization of Moore FSMs, but it is the subject of further research.

References

Ahmed I. and Dhodhi M.K. (2000): *State assignment of finite state machines*. — IEE Proc. Comp. Digit. Techn., Vol. 147, No. 1, pp. 15–22.

Baranov S. (1994): *Logic Synthesis for Control Automata*. — Boston: Kluwer.

Barkalov A.A. and Palagin A.V. (1997): *Synthesis of Microprogram Control Units*. — Kiev: IC NAS of Ukraine, (in Russian).

Barkalov A.A. (2002): *Synthesis of Control Units on Programmable Logic Devices*. — Donetsk: DNTU, (in Russian).

Barkalov A.A. (2003): *Synthesis of Operational Devices*. — Donetsk: DNTU, (in Russian).

De Micheli G. (1994): *Synthesis and Optimization of Digital Circuits*. — New York: McGraw Hill.

Devadas S. and Newton R. (1991): *Exact algorithms for output encoding, state assignment, and four-level Boolean optimization*. — IEEE Trans. Comp. Aided Design, Vol. 10, No. 1, pp. 13–27.

Kania D. (2003): *Efficient approach to synthesis of multioutput Boolean functions on PAL-base devices*. — IEE Proc. Comp. Digit. Techn., Vol. 150, No. 3, pp. 143–149.

Lahtinen V., Kuasilinna K. and Hamalainen T. (2002): *Optimizing finite state machine for system-on-chip communication*. — Proc. IEEE Int. Symp. *Circuits and Systems*, Milan, Italy, Vol. 1, pp. 485–488.

Novikov G. (1974): *About one approach for finite-state-machines research*. — Contr. Syst. Mach., No. 2, pp. 70–75, (in Russian).

Singh M. and Nowick S.M. (2002). *Synthesis for logic initializability of synchronous finite state machines*. — IEEE Trans. VLSI Syst., Vol. 24, No. 5, pp. 542–557.

Solovjev V.V. (1996): *Design of the Functional Units of Digital Systems Using Programmable Logic Devices*. — Minsk: Bestprint, (in Russian).

Solovjev V.V. (2001): *Design of Digital Systems Using the Programmable Logic Integrate Circuits*. — Moscow: Hotline–Telecom, (in Russian).

Received: 28 April 2004
Revised: 20 May 2004