R. M. Babakov, ph. d, Assistant Professor

ACCOUNTING OF PROBABILITY OF STATES IN MICROPROGRAM FINITE-STATE MACHINE WITH DATAPATH OF TRANSITIONS

An approach for determining the comparative effectiveness of the variants of synthesis of a microprogrammable finite-state machine with datapath of transitions is proposed. The approach is to determine the average number of transitions to each state of the finite-state machine with further calculation of the probability of states and the average number of cycles in one cycle of the finite-state machine functioning. This approach is useful when using transit states and allows to choose one of several synthesis variants, in which the average number of clock cycles of a finite-state machine turns out to be the smallest.

Key words: *microprogrammable finite-state machine, datapath of transitions, transit states, probabilities of states.*

Introduction

An important element of modern computing systems is the control unit (CU), which coordinates functionality of all nodes of the system and largely determines its characteristics [1]. One of the ways of realizing a CU is a microprogram finite-state machine (FSM), in which relatively high speed is combined with considerable hardware expenses [2]. In order to reduce the hardware expences for the implementation of the logical circuit of the finite-state machine, the FSM can be realized in the form of an MPA with datapath of transitions (FSM with DT) [3].

In paper [4] it is shown that the use of additional (transit) states contributes to the reduction of hardware expenses in the logical circuit of FSM with DT. This increases the average number of FSM transitions performed per execution of the algorithm implemented by the FSM. Since for the same FSM in general a set of solutions using transit states can be obtained, the problem of choosing a solution, in which the increase in the average number of automatic transitions is minimal, is actualized. In this paper, it is proposed to estimate the increase in the average number of clock cycles of an FSM on the basis of known probabilities of the values of the signals of logical conditions analyzed during the FSM work cycle.

The purpose of research is to reduce the average time of one cycle of the finite-state machine with datapath of transitions. The problem solved in this paper is to investigate the influence of the probability of states of FSM with DT on the average number of clock cycles in one FSM work cycle.

Analysis of research and publications

A flow-chart of the algorithm can be used as the initial data for the synthesis of FSM with DT [5]. Today, a number of methods for optimization of the circuit of FSM, based on the transformation of the initial flow-chart, the result of which is an increase in the number of states of the FSM, are known [2, 5]. In the case of FSM with DT, one of such methods is the use of transit states, which in some cases reduces the number of FSM transitions realized canonically by the system of equations [4].

Let us use transition operations (TO) O_1-O_3, defined by the expressions (1)-(3), in which $K(a^t)$ is the current state code, $K(a^{t+1})$ – code of next state, "mod 16" – the operation of obtaining the remainder of the integer division by 16, in the process of synthesis of the FSM with DT according to the flow-chart *Г*, indicated by the states of the Moore FSM (Fig. 1).

$$O_1: \quad K(a^{t+1}) = (K(a^t) + 13) \bmod 16, \tag{1}$$

$$O_2: \quad K(a^{t+1}) = K(a^t) \div 4\,, \tag{2}$$

$$O_3: \quad K(a^{t+1}) = \overline{K(a^t)}\,. \tag{3}$$

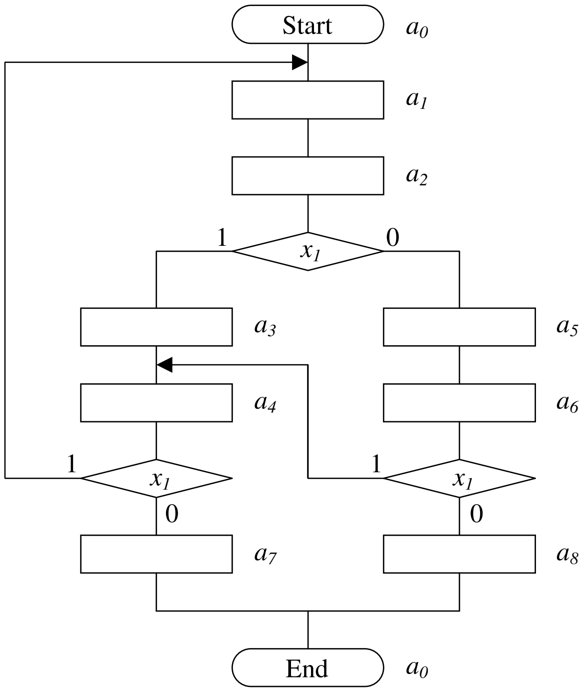


Fig. 1. Flow-chart Γ

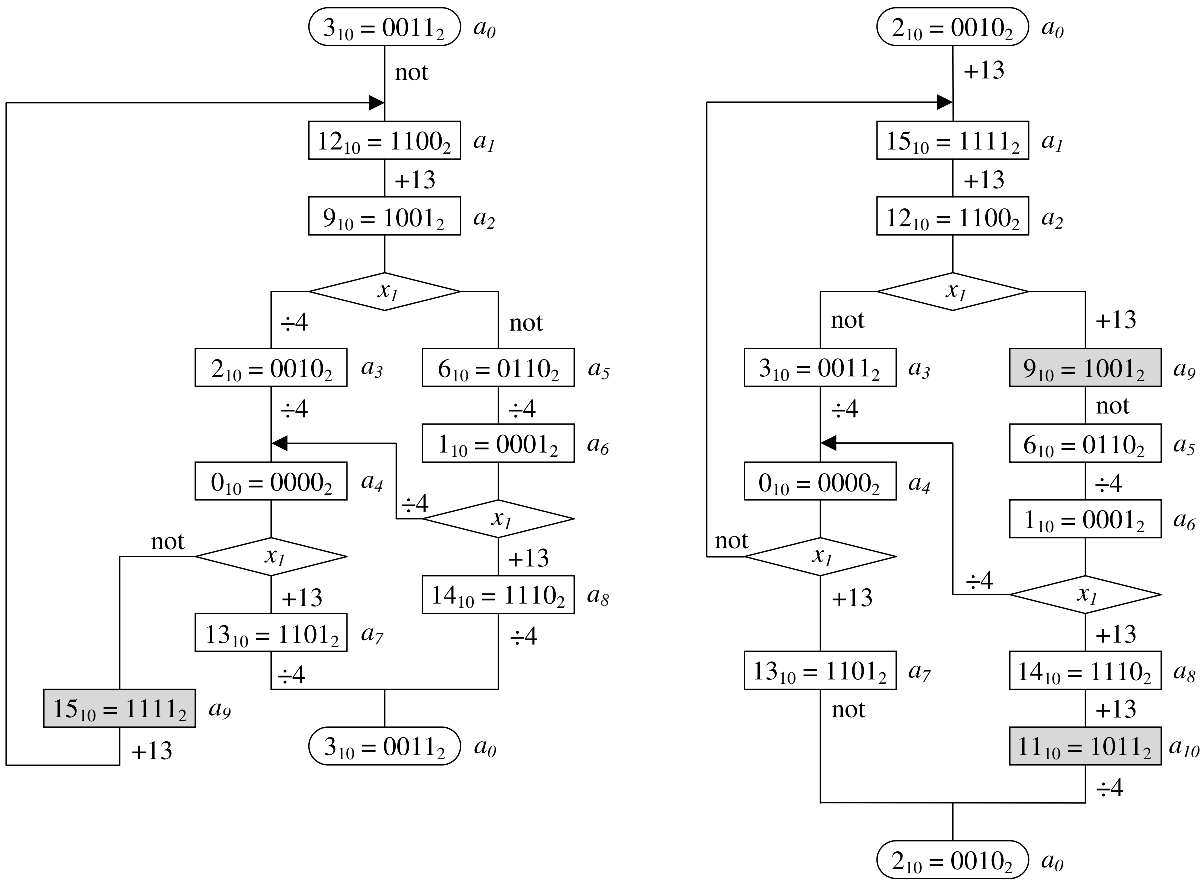
Since the output function of the FSM does not have a direct effect on the transition function, microoperations in operational nodes are not indicated in Fig. 1.

In Fig. 2 two flow-charts Γ_1 and Γ_2 obtained by transforming the initial flow-chart Γ and being different variants of the synthesis of FSM with DT by flow-chart Γ, are shown. The initial, final, and operational nodes contain binary state codes and their equivalent decimal values. Each branch is marked with one of the specified TOs: "+13" corresponds to O_1, «÷4» to O_2, "not" to O_3. Flow-chart Γ_1 (Figure 2, *a*) contains one transit state *a*9, flow-chart Γ_2 (Figure 2, *b*) – two transit states a_9 and a_{10}. At the same time, in this flow-charts transit states are located in different branches, have different codes, and different TOs are used to implement transitions from transit states.

If in this case the synthesis variant represented by flow-chart Γ_2 contains one more state than the variant represented by Γ_1, in general the difference in the number of states can be larger. Since increase in the number of states leads to increase in the functioning time of the FSM, the question arises: is the functioning time of an FSM synthesized by a flow-chart with a larger number of states exceeding the functioning time of an FSM synthesized by a flow-char with a smaller number of states. The answer to this question would allow choosing a variant of synthesis of FSM with DT, leading to the least loss in the execution time of the algorithm implemented by the FSM.

Accounting for the probability of the truth of logical conditions

In some cases, for each input signal, the probabilities of the appearance of this signal at the input of the FSM at an arbitrary instant of time are known. When the FSM is given by a flow-chart, the input signals are encoded by vectors formed by the values of the structural (binary) variables of the logical conditions (LC) x_1, ..., x_L [1, 2]. In this case, the probabilities of the appearance of input signals can be determined based on the probabilities of the truth of the LC at each moment of time.



a) *b)*

Fig. 2. Transformed flow-charts Γ_1 (a) и Γ_2 (b)

Suppose that for all logical conditions x_1, ..., x_L contained in a given flow-chart, the probabilities $p(x_1)$, ..., $p(x_L)$ of the truth of the corresponding LC at each moment of time are known. This makes it possible to determine for each operational node the average number of cycles spent on its execution per pass of the flow-chart corresponding to one cycle of the FSM functioning. Since the states in FSM are associated with operational nodes or their outputs, the average number of cycles spent on executing the operational node will correspond to the average number of FSM transitions to the corresponding state during single pass of the flow-chart.

We denote by the $q(a_i)$ average number of transitions to the state performed in a single pass of the flow-chart. Then the average number of cycles Q spent on the execution of a microprogram is determined by the sum of the values of q for all states of a given flow-chart:

$$Q = \sum_{i=0}^{M-1} q(a_i)\,. \qquad (4)$$

To determine Q for known values of $p(x_1)$, ..., $p(x_L)$ the technique given in [6] can be used. Its application to flow-chart Γ (Figure 1), containing a single LC x_1, allows us to determine the values of q for different values of probability $p(x_1)$. The calculation results are given in Table. 1.

We consider the flow-chart Γ_1 (Fig. 2, a). In it, the transit state a_9 is added after the state a_4, and the transition $a_4 \rightarrow a_9$ is a single incoming transition for a_9. This allows us to determine the value $q(a_9)$ by the following expression:

$$q(a_9) = q(a_4) \cdot p(x_1)\,. \tag{5}$$

Table 1

Values of $q(a_i)$ for different probabilities $p(x_1)$

$p(x_1)$	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9
$q(a_0)$	1	1	1	1	1	1	1	1	1
$q(a_1)$	1,02	1,08	1,18	1,34	1,60	2,02	2,75	4,31	9,17
$q(a_2)$	1,02	1,08	1,18	1,34	1,60	2,02	2,75	4,31	9,17
$q(a_3)$	0,1	0,22	0,35	0,54	0,80	1,21	1,93	3,45	8,26
$q(a_4)$	0,19	0,39	0,6	0,86	1,20	1,69	2,51	4,14	9,08
$q(a_5)$	0,92	0,86	0,83	0,81	0,80	0,81	0,83	0,86	0,92
$q(a_6)$	0,92	0,86	0,83	0,81	0,80	0,81	0,83	0,86	0,92
$q(a_7)$	0,17	0,31	0,42	0,52	0,60	0,68	0,75	0,83	0,91
$q(a_8)$	0,83	0,69	0,58	0,48	0,40	0,32	0,25	0,17	0,09
Q	6,17	6,49	6,97	7,7	8,8	10,56	13,6	19,93	39,52

This value corresponds to the increment ΔQ_1 of the Q value when using the flow-chart Γ_1 instead of the flow-chart Γ_1.

By analogy with (5), we will form expressions (6) and (7) for the transit states of the flow-chart Γ_2.

$$q(a_9) = q(a_2) \cdot (1 - p(x_1))\,. \tag{6}$$

$$q(a_{10}) = q(a_8) \cdot 1 = q(a_8)\,. \tag{7}$$

In this case

$$\Delta Q_2 = q(a_9) + q(a_{10})\,. \tag{8}$$

Let us compare the values of ΔQ_1 and ΔQ_2 for different values of probability, for which we will compile Table. 2.

Dividing ΔQ_1 and ΔQ_2 by the corresponding values of Q from Table. 1, we obtain the relative values of ΔQ_1 and ΔQ_2, presented in Table. 3 as a percentage.

Table 2

Values of ΔQ_1 and ΔQ_2 for different probabilities $p(x_1)$

$p(x_1)$		0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9
Γ_1	$q(a_9)$	0,02	0,08	0,18	0,34	0,6	1,01	1,76	3,31	8,17
	ΔQ_1	0,02	0,08	0,18	0,34	0,6	1,01	1,76	3,31	8,17
Γ_2	$q(a_9)$	0,92	0,86	0,83	0,81	0,80	0,81	0,83	0,86	0,92
	$q(a_{10})$	0,83	0,69	0,58	0,48	0,40	0,32	0,25	0,17	0,09
	ΔQ_2	1,75	1,55	1,41	1,29	1,2	1,13	1,08	1,03	1,01

Table 3

Relative values of ΔQ_1 and ΔQ_2

$p(x_1)$	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9
ΔQ_1, %	0,3	1,2	2,6	4,4	6,8	9,6	12,9	16,6	20,7
ΔQ_2, %	28,3	23,9	20,2	16,8	13,6	10,7	7,9	5,1	2,6

Analysis of Tables 1-3 allows us to draw the following conclusions.

1. The influence of the probabilities of the truth of the logical conditions on the average number of cycles of microprogram execution depends on the structure of the flow-chart. The application of known methods makes it possible to determine the value of Q before and after the addition of transit states, which allows to evaluate the feasibility of adding transit states in terms of increasing the average time of one cycle of the FSM functioning.

2. Using the example of flow-charts Γ_1 and Γ_2, it can be seen that a smaller number of transit states used in the case of Γ_1 compared to Γ_2 is not a guarantee of a higher efficiency with respect to the execution time of the algorithm. The structure of the flow-chart is such that, for small values $p(x_1)$, the value of ΔQ_2 exceeds the value of ΔQ_1, while for $p(x_1)$>0,7 the situation changes to the opposite.

Conclusions

Researches have shown that a larger number of states do not necessarily lead to a longer FSM run time. Knowledge of the probabilities of the truth values of the logical conditions of a given flow-chart makes it possible to choose a variant of synthesis of FSM with DT more effective from the point of view of the average execution time of the implemented algorithm, as well as to estimate the increase in the average time of one cycle of FSM functioning and the expediency of using transit states in each specific case. The approach considered in this paper can be taken into account in the development and algorithmization of methods for synthesizing of FSM with DT as part of specialized CAD of digital control devices.

REFERENCE

1. Glushkov V. M. Syntesis of digital mashines / V. M. Glushkov. – M. : Phizmatgiz, 1962. – 476 p. (Rus).

2. Baranov S. I. Synthesis of microprogramming mashines / S. I. Baranov. – L. : Energy, 1979. – 232 p. (Rus).

3. Barkalov A. A. Operational formation of state codes in microprogramming machines / A. A. Barkalov, R. M. Babakov // Cybernetics and system analysis. – 2011. – № 2. – P. 21 – 26. (Rus).

4. Babakov R. M. Using of transit states in microprogramming machine with operation transition machine / R. M. Babakov, I. V. Yarosh // Proceedings of Donetsk National Technical University. Series: «Computer engineering and automation». – Krasnoarmeisk: DonNTU, 2016. – Issue 1 (29). – P. 56 – 64. (Rus).

5. Barkalov A. A. Synthesis of control devices on programmable logic devices / A. A. Barkalov. – Donetsk, Donetsk National Technical University, 2002. – 262 p. (Rus).

6. Mayorov S. A. Structure of electronic computers / S. A. Mayorov, G. I. Novikov. – L. : Mashine-building, 1979. – 384 p. (Rus).

Babakov Roman **–** Cand.Sc,(Eng.), Assistant Professor, Assistant Professor with the Chair of Applied Mechanics and Computer Technologies, e-mail: r.babakov@donnu.edu.ua.

Vasyl' Stus Donetsk National University, Vinnytsia.