

Запропоновано метод зменшення кількості елементів табличного типу у схемі мікропрограмного автомата (МПА) Мілі. Метод заснований на використанні вбудованого блоку пам'яті, який здійснює заміну вхідних змінних та кодування виходів автомата. Розглянуто випадок наявності лише одного блоку пам'яті. Показано умови доцільності застосування запропонованого методу. Ці умови враховують характеристики МПА і елементного базису. Розглянуто приклад синтезу МПА із застосуванням запропонованого методу. Результати досліджень свідчать про позитивний ефект запропонованого методу.

Ключові слова: автомат Мілі, FPGA, синтез, кодування вхідів, кодування наборів виходів.

© О.О. Баркалов, Л.О. Тітаренко,
О.М. Головін, О.В. Матвієнко, 2023

УДК 004.274

DOI:10.34229/2707-451X.23.3.8

О.О. БАРКАЛОВ, Л.О. ТІТАРЕНКО, О.М. ГОЛОВІН, О.В. МАТВІЄНКО

ОПТИМІЗАЦІЯ СХЕМИ АВТОМАТА МІЛІ У ЗМІШАНОМУ БАЗИСІ

Вступ. Модель мікропрограмного автомата (МПА) Мілі [1] – одна з фундаментальних моделей, що використовується для синтезу послідовних схем [1, 2]. До таких схем належать, наприклад, пристрої керування [3] або інтерфейсні перетворювачі вбудованих систем [4]. При синтезі схем МПА з'являється ряд оптимізаційних завдань [5, 6]. Одне з найважливіших – завдання зменшення площі кристала, яку займає схема МПА [7, 8]. Розв'язання цього завдання дозволяє зменшити споживану потужність та час циклу МПА [6].

Методи вирішення цього завдання багато в чому залежать від особливостей елементного базису, що використовується для реалізації схем МПА. Нині більшість цифрових систем реалізуються у базисі надвеликих інтегральних схем типу FPGA (field-programmable logic array) [9, 10]. Наприклад, у роботі [11] наводиться близько 1700 прикладів застосування FPGA. На основі цього аналізу ми орієнтуємо нашу статтю на базис FPGA.

Основні виробники FPGA це фірми Intel (Altera) [9] та AMD Xilinx [10]. При цьому компанія AMD Xilinx домінує на цьому ринку. Тому як елементний базис ми вибрали FPGA фірми Xilinx. Подібні мікросхеми складаються з конфігурованих логічних блоків (КЛБ), програмованої матриці міжз'єднань, дерева синхронізації та програмованих вхідів-виходів. Для реалізації схем МПА можна використовувати КЛБ двох типів. Один з них – це табличні логічні елементи (ТЛЕ), пов'язані з програмованими тригерами та мультиплексами. Другий тип КЛБ – це вбудовані блоки пам'яті (ВБП), які мають властивість реконфігурації.

Схема МПА може бути реалізована з використанням ТЛЕ або ВБП. Однак найкращі результати можуть бути досягнуті при використанні змішаного базису [7, 12]. Саме таке завдання ми розглядаємо у цій роботі. При цьому запропонований метод спрямований на випадок, коли розробник має лише один блок ВБП. Такий випадок цілком можливий через те,

що ВБП широко використовуються для реалізації різних операційних блоків цифрових систем [6].

Реалізація схеми МПА у базисі FPGA. Автомат Мілі характеризується трьома множинами та двома функціями [1, 2]. Входи МПА утворюють множину $I = \{i_1, \dots, i_L\}$, виходи об'єднані у множину $O = \{o_1, \dots, o_N\}$, а стани входять у множину $S = \{s_1, \dots, s_M\}$. Функція переходів δ задає залежність між станами та входами: $\delta: S \times I \rightarrow S$. Функція виходів λ визначає залежність між усіма трьома множинами: $\lambda: S \times I \rightarrow O$. Ми розглядаємо автомати з явно виділеним початковим станом $s_1 \in S$, що дозволяє однозначно визначити початкові умови функціонування пристрою [1, 2].

Найчастіше автомати задаються у вигляді графів або таблиць [1, 2]. У даній статті ми обрали табличну форму, коли МПА визначається таблицею переходів, що має стовпці: s_m, s_t, I_h, O_h, h [1, 2]. У таблиці використані такі позначення: s_m – початковий стан; s_t – стан переходу ($s_m, s_t \in S$); I_h – вхідний сигнал, що визначає перехід і рівний кон'юнкції деяких входів (або їх інверсій); O_h – набір виходів (НВ), що формуються на переході $\langle s_m, s_t \rangle$; h – номер переходу ($h \in \{1, \dots, H\}$).

Для синтезу схеми МПА необхідно закодувати стани $s_m \in S$ двійковими кодами $C(s_m)$, які мають R розрядів [6]. Кодуючі змінні утворюють множину $T = \{T_1, \dots, T_R\}$. Кожна змінна є виходом тригера. Як правило, ці тригери мають інформаційні входи типу D [13]. Зміною кодів тригерів керують функції збудження пам'яті (ФЗП), що утворюють множину $D = \{D_1, \dots, D_R\}$.

Після кодування станів вихідна таблиця переходів перетворюється на пряму структурну таблицю (ПСТ) [1]. Ця таблиця містить три додаткові стовпці: $K(s_m), K(s_t)$ і D_h . Стовпець D_h містить набори ФЗП, рівних 1 для зміни $K(s_m)$ на $K(s_t)$. Тригери утворюють регістр коду стану (РКС), який керується сигналами синхронізації Clock та Start.

Ми розглядаємо автомат із мінімальною кількістю тригерів. Це число визначає формула [2]

$$R = \lceil \log_2 M \rceil. \tag{1}$$

Слід зазначити, що запропонований метод не залежить від R .

Використовуючи ПСТ, можна знайти дві системи булевих функцій (СБФ):

$$D = D(T, I); \tag{2}$$

$$O = O(T, I). \tag{3}$$

Системи (2) – (3) визначають структурну схему МПА Мілі A_1 (рис. 1).

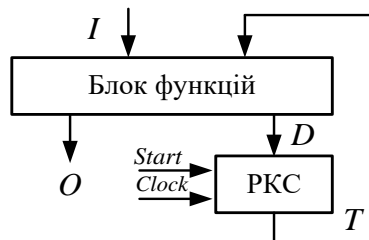


РИС. 1. Структурна схема МПА Мілі A_1

В автоматі A_1 блок функцій реалізує СБФ (2) – (3). У статті цей блок реалізується за допомогою внутрішніх ресурсів FPGA. Розглянемо їх особливості.

Елемент ТЛЕ має I_o входів та один вихід. При цьому ТЛЕ реалізує довільну функцію алгебри логіки, яка залежить від 1 до I_o аргументів. Якщо кількість аргументів перевищує I_o , то для реалізації схеми потрібно декілька елементів. Для реалізації подібних схем використовуються методи структурної декомпозиції (СД) [14 – 16]. Для найпотужніших FPGA сімейства Virtex-7 [14] $I_o = 6$. Вихід ТЛЕ може бути підключений до входу тригера, через внутрішній мультиплексор блоку КЛБ. Сукупність цих тригерів утворює прихований розподілений регістр кодів станів РКС [15]. На рис. 2, а показана структурна схема МПА A_1 на базі ТЛЕ.

У схемі на рис. 2, а блок виходів реалізує комбінаційну схему, що задається СБФ (3). Блок пам'яті складається з комбінаційної схеми, що реалізує СБФ (2) та регістр РКС. Регістром керують імпульси Clock та Start.

Елемент ВБП – конфігурована пам'ять, в якій може змінюватися кількість адресних входів (від 1 до R_A) і виходів (від 1 до t_F). При цьому ємність пам'яті V_o – незмінна і дорівнює

$$V_o = 2^{R_A} \cdot t_F. \tag{4}$$

Для FPGA фірми Xilinx $V_o = 32$ Кбіт [14] існує низка конфігурацій від $\langle 15, 1 \rangle$ до $\langle 9, 64 \rangle$. У цих парах на першому місці знаходиться величина параметра R_A , на другому – число виходів t_F , що відповідає цій кількості входів.

Блок ВБП має внутрішній вихідний регістр пам'яті з можливістю встановлення його в "0" [12]. Тому при реалізації схеми МПА на ВБП немає потреби в окремому регістрі.

Нехай виконуються наступні умови:

$$R_A \geq L + R; \tag{5}$$

$$t_F \geq N + R. \tag{6}$$

У цьому випадку МПА A_1 може бути представлений як один блок ВБП (рис. 2, б).

У схемі рис. 2, б виходи $o_n \in O$ – синхронні. Зміна виходів та коду стану відбувається одночасно за сигналом Clock. Далі виходи не змінюються протягом такту роботи МПА. Це дозволяє усунути додатковий регістр, необхідний для стабільної роботи операційного пристрою, яким керує МПА Мілі [1, 2].

Якщо немає пари $\langle R_A, t_F \rangle$, яка задовольняє умовам (4)–(6), то частина схеми МПА має бути реалізована за допомогою елементів ТЛЕ, тобто у змішаному базисі.

Нехай функція $f_i \in O \cup D$ залежить від $NA(f_i)$ аргументів. Якщо умова

$$NA(f_i) \leq I_o \tag{7}$$

виконується для всіх функцій з множини $O \cup D$, то схема МПА A_1 (рис. 2, а) має тільки $N + R$ елементів. Однак, умова (7) зазвичай виконується для дуже простих автоматів.

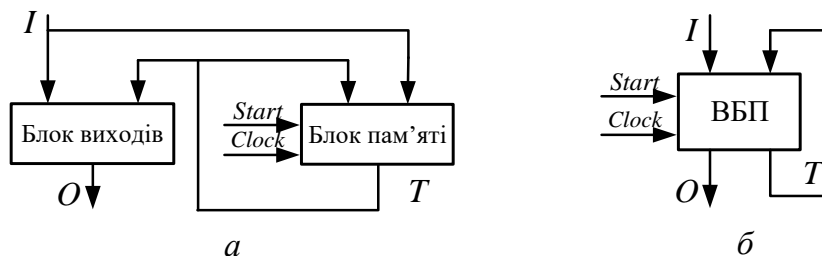


РИС. 2. Структурна схема МПА A_1 на базі: а –ТЛЕ; б – ВБП

У разі порушення умови (7) схема має кілька рівнів логіки. Для синтезу таких схем слід застосовувати методи функціональної декомпозиції (ФД) [4]. Недоліком таких схем є складна система між'єднань [16], яка збільшує час затримки сигналу та споживану потужність [16].

Характеристики багаторівневих схем можна покращити за рахунок застосування методів структурної декомпозиції [16]. Сенс цих методів зводиться до запровадження додаткових функцій, що спрощує схеми формування функцій $D_r \in D$ і $o_n \in O$ [1]. Два основних методи СД це кодування вхідних змінних (КВЗ) та кодування наборів виходів (КНВ).

Метод КВЗ зводиться до заміни входів $i_l \in I$ додатковими змінними з множини $P = \{p_1, \dots, p_G\}$, де $G \ll L$. Для заміни $I \rightarrow P$ використовується СБФ

$$P = P(T, I). \tag{8}$$

На переходах МПА формується Q наборів $O_q \subseteq O$. Кожен набір O_q кодується двійковим кодом $K(O_q)$, який має R_Q розрядів:

$$R_Q = \lceil \log_2 Q \rceil. \tag{9}$$

Для кодування використовуються змінні з множини $Z = \{z_1, \dots, z_{R_Q}\}$.

Для втілення системи (3) необхідно реалізувати наступні СБФ: $O = O(T, I)$

$$Z = Z(T, I); \tag{10}$$

$$O = O(Z). \tag{11}$$

У цій роботі ми розглядаємо випадок спільного застосування цих двох методів. При цьому схема МПА задається СБФ (8), (11) та

$$D = D(T, P); \tag{12}$$

$$Z = Z(T, P). \tag{13}$$

Для реалізації систем (8), (11) – (13) використовується МПА A_2 (рис. 3).

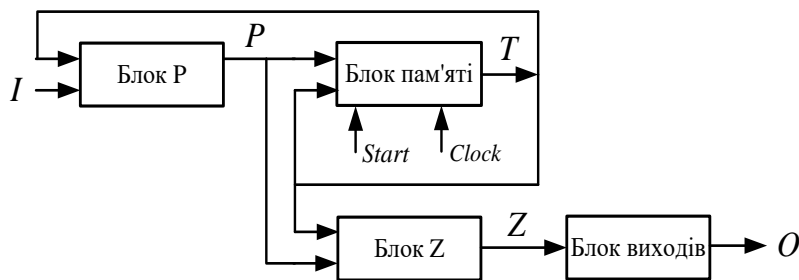


РИС. 3. Структурна схема МПА Мілі A_2

У МПА A_2 на рис. 3 Блок P реалізує СБФ (8), Блок пам'яті – СБФ (12), Блок Z – СБФ (13) та Блок виходів – СБФ (11). Як показано в [17], такий підхід призводить до схеми МПА з мінімальним числом елементів ТЛЕ, а також регулярною системою між'єднань і мінімальним споживанням пам'яті. Однак частота роботи МПА A_2 менша, ніж у МПА A_1 .

У роботі пропонується реалізувати частину схеми МПА A_2 на базі ВБП. Це дозволить зменшити кількість елементів ТЛЕ та їх між'єднань. У цьому разі можливо очікувати збільшення частоти роботи схеми МПА у порівнянні з її аналогом лише на елементах ТЛЕ.

Запропонований метод синтезу. У цій статті розглядається випадок, коли в розпорядженні розробника є лише один блок БВП і його недостатньо для реалізації схеми. Це означає, що немає конфігурації $\langle R_A, t_F \rangle$, для якої виконуються умови (5), (6). Крім того, порушено умову (7) і необхідно використовувати методи ФД.

Виконаємо кодування входів та наборів виходів, що дасть параметри G та R_Q . Нехай для БВП існує конфігурація $\langle R_o, t_o \rangle$, для якої виконуються умови:

$$R_o \geq L + R; \tag{14}$$

$$t_o \geq G + R_Q. \tag{15}$$

Для цього випадку пропонується реалізувати системи (8) та (10) за допомогою БВП. Решта схеми реалізується на елементах ТЛЕ. Це призводить до МПА A_3 (рис. 4).

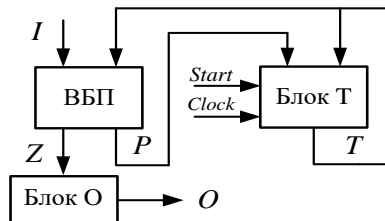


РИС. 4. Структурна схема МПА A_3

У МПА A_3 два блоки реалізуються на елементах ТЛЕ. При виконанні умови

$$R_Q \leq I_o \tag{16}$$

схема блоку виходів (Блок О) має точно N або менше елементів. При виконанні умови

$$G + R \leq I_o; \tag{17}$$

схема блоку ФЗП (блок Т) має точно R елементів. Умови (16), (17) визначають найкращу ситуацію для реалізації автомата A_3 . У разі порушення (16) схема блоку виходів – багаторівнева, а схема блоку ФЗП – багаторівнева за порушення умови (17).

В цій роботі пропонується метод синтезу МПА Мілі A_3 . Розглядається випадок, коли поведінка МПА задано таблицею переходів. Метод включає такі етапи:

1. Кодування станів $s_m \in S$ кодами $R_o \geq L + R = 9$, $t_o \geq G + R_Q = 6$.
2. Кодування наборів $O_q \subseteq O$ кодами $K(O_q)$.
3. Формування системи (11), яка задає Блок О.
4. Кодування входів та формування СБФ (8).
5. Формування таблиці блоку БВП.
6. Формування модифікованої таблиці ПСТ та СБФ (12).
7. Реалізація схеми МПА у заданому базисі.

Приклад синтезу автомата A_3 . Розглянемо приклад застосування запропонованого методу синтезу схеми МПА М1 (табл.1). Для реалізації схеми використовуються елементи ТЛЕ з числом входів $I_o = 5$ та блоки БВП з конфігураціями $\langle 12, 1 \rangle$, $\langle 11, 2 \rangle$, $\langle 10, 4 \rangle$, $\langle 9, 8 \rangle$, $\langle 8, 16 \rangle$.

Як впливає з табл. 1, автомат М1 має $M = 7$ станів, $L = 6$ входів та $N = 9$ виходів. Використовуючи (1), отримаємо $R = 3$ та множини $T = \{T_1, T_2, T_3\}$, $D = \{D_1, D_2, D_3\}$. Отже, $L + R = 9$ та $N + R = 12$. Очевидно, умови (5)–(6) не виконуються для жодної конфігурації ВБП. Отже, схема М1 має реалізовуватися в змішаному елементному базисі.

Виконаємо кодування станів тривіальним чином: $K(s_1) = 000$, $K(s_2) = 001, \dots, K(s_7) = 110$. У реальній практиці можна використати різні оптимізаційні алгоритми кодування [4]. Однак для ілюстрації методу синтезу це не має значення.

У табл. 1 є $Q = 9$ різних НВ. Це такі набори: $O_1 = \emptyset$, $O_2 = \{o_1, o_2, o_7\}$, $O_3 = \{o_1\}$, $O_4 = \{o_1, o_3, o_5, o_7\}$, $O_5 = \{o_4\}$, $O_6 = \{o_3\}$, $O_7 = \{o_2, o_4, o_6\}$, $O_8 = \{o_5\}$, $O_9 = \{o_1, o_3, o_5\}$. З формули (9) маємо $R_Q = 4$, $Z = \{z_1, z_2, z_3, z_4\}$. Використовуємо метод [4] та закодуємо НВ, як показано на рис. 5.

Використовуючи вміст НВ та коди виходів $K(O_q)$, побудуємо СБФ

$$\begin{aligned}
 o_1 &= O_2 \vee O_3 \vee O_4 \vee O_9 = \overline{z_1}z_2; & o_2 &= O_2 \vee O_7 = \overline{z_3}z_4; \\
 o_3 &= O_4 \vee O_6 \vee O_9 = \overline{z_2}z_3; & o_4 &= O_5 \vee O_7 = z_2\overline{z_3}; \\
 o_5 &= O_4 \vee O_8 \vee O_9 = z_2z_3; & o_6 &= O_7 = z_1z_4; \\
 o_7 &= O_2 \vee O_4 = z_2z_4.
 \end{aligned}
 \tag{18}$$

ТАБЛИЦЯ 1. Таблиця переходів автомата М1

s_m	s_t	I_h	O_h	h
s_1	s_2	i_1	$o_1o_2o_7$	1
	s_3	$\overline{i_1}i_2$	o_5	2
	s_4	$\overline{i_1}\overline{i_2}$	o_4	3
s_2	s_2	i_3i_4	o_3	4
	s_4	$i_3\overline{i_4}$	$o_1o_2o_7$	5
	s_5	$\overline{i_3}$	o_1	6
s_3	s_2	i_5	o_4	7
	s_1	$\overline{i_5}$	$o_1o_3o_5o_7$	8
s_4	s_5	i_6	o_3	9
	s_6	$\overline{i_6}i_4$	$o_1o_3o_5$	10
	s_7	$\overline{i_6}\overline{i_4}$	o_4	11
s_5	s_1	1	o_3	12
s_6	s_2	i_1	o_3	13
	s_4	$\overline{i_1}i_5$	o_4	14
	s_7	$\overline{i_1}\overline{i_5}$	$o_2o_4o_6$	15
s_7	s_5	1	—	16

		Z_1Z_2			
		00	01	11	10
Z_3Z_4	00	O_1	O_3	*	O_5
	01	*	O_2	*	O_7
	11	*	O_4	*	*
	10	O_6	O_9	O_8	*

РИС. 5. Кодування наборів виходів МПА М1

Система (18) має сумарно 14 літералів. Кожен літерал відповідає між'єднанню у схемі МПА. Між'єднання впливають на споживану потужність та швидкодію [16]. Тому навіть за виконання умови (16) доцільно оптимізувати СБФ (11).

Для виконання КВЗ необхідно [1].

1. Знайти числа L_m , які дорівнюють кількості входів, що визначають переходи зі станів $s_m \in S$.
2. Знайти кількість додаткових змінних

$$G = \max (L_1, L_2, \dots, L_M). \tag{19}$$

3. Побудувати таблицю заміни входів додатковими змінними $p_g \in P$.
4. Побудувати систему (8), використовуючи коди станів МПА.

З табл.1 маємо $L_1 = L_2 = L_4 = L_6 = 2, L_3 = 1, L_5 = L_7 = 1$. З формули (19) отримаємо $G = 2$ та $P = \{p_1, p_2\}$. Таблиця заміни (табл. 2) будується тривіально, використовуючи метод [1].

ТАБЛИЦЯ 2. Таблиця КВЗ автомата М1

s_m	s_1	s_2	s_3	s_4	s_5	s_6	s_7
p_1	i_1	i_3	—	i_6	—	i_1	—
p_2	i_2	i_4	i_5	i_4	—	i_5	—
$K(s_m)$	000	001	010	011	100	101	110

Терми системи (8) залежать від кон'юнкції змінних $T_r \in T$, відповідних кодів станів рядка $K(s_m)$. Друга складова цих термів це входи. Оскільки система (8) реалізується на ВБП, немає сенсу в зменшенні числа літералів у її термах. Насправді ВБП реалізують таблицю істинності СБФ [18]. У нашому випадку ця СБФ має такий вигляд:

$$\begin{aligned}
 p_1 &= \overline{T_1}\overline{T_2}\overline{T_3}i_1 \vee \overline{T_1}\overline{T_2}T_3i_3 \vee \overline{T_1}T_2\overline{T_3}i_6 \vee T_1\overline{T_2}T_3i_1; \\
 p_2 &= \overline{T_1}\overline{T_2}\overline{T_3}i_2 \vee \overline{T_1}\overline{T_2}T_3i_4 \vee \overline{T_1}T_2\overline{T_3}i_5 \vee \overline{T_1}T_2T_3i_4 \vee T_1\overline{T_2}\overline{T_3}i_5.
 \end{aligned} \tag{20}$$

Перевіримо можливість реалізації систем (2) та (8) за допомогою одного блоку вбудованої пам'яті. Маємо $R = 3, G = 2, L = 6$ та $R_Q = 4$. Це дає $R_o \geq L + R = 9, t_o \geq G + R_Q = 6$. За умовою, серед конфігурацій ВБП є варіант $<9, 8 > z t_o = 8 > G + R_Q$ і $R_o = 9 = L + R$. Умови (14), (15) виконуються і використання моделі A_3 у цьому випадку можливе. Крім того, виконуються умови (16), (17). При цьому кожна з функцій систем (11), (12) реалізується схемою, що складається з одного ТЛЕ.

Таблиця ВБП має стовпці $T_1, \dots, T_R, i_L, \dots, i_1$ як аргументи і стовпці $z_1, \dots, z_{R_Q}, p_1, \dots, p_G$ як функції. У загальному випадку ця таблиця має $H_o = 2^{R+L}$ рядків.

У нашому випадку $H_o = 2^9 = 512$ рядків. Частина цієї таблиці наведена у табл. 3.

ТАБЛИЦЯ 3. Фрагмент таблиці ЕМВ автомата М1

T_1	T_2	T_3	i_6	i_5	i_4	i_3	i_2	i_1	p_1	p_2	z_1	z_2	z_3	z_4	o_1	o_2	v	h
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	3
0	0	0	0	0	0	0	0	1	0	1	1	1	1	0	0	0	1	2
0	0	0	0	0	0	0	1	0	1	0	0	1	0	1	1	1	2	1
0	0	0	0	0	0	0	1	1	1	1	0	1	0	1	1	1	3	1
0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	4	3
0	0	0	0	0	0	1	0	1	0	1	1	1	1	0	0	0	5	2
0	0	0	0	0	0	1	1	0	1	0	0	1	0	1	1	1	6	1
0	0	0	0	0	0	1	1	1	1	1	0	1	0	1	1	1	7	1
0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	8	3
0	0	0	0	0	1	0	0	1	0	1	1	1	1	0	0	0	9	2
0	0	0	0	0	1	0	1	0	1	0	0	1	0	1	1	1	10	1
0	0	0	0	0	1	0	1	1	1	1	0	1	0	1	1	1	11	1

Табл. 3 задає переходи зі стану $s_1 \in S$. У стовпцях v показаний десятковий еквівалент адреси осередку пам'яті; у стовпці h показано відповідність між таблицями переходів та вмісту ВБП.

Модифікована ПСТ автомата A_3 включає стовпці $s_m, K(s_m), s_t, K(s_t), P_h, D_h, h$ (табл. 4).

ТАБЛИЦЯ 4. Модифікована ПСТ автомата М1

s_m	$K(s_m)$	s_t	$K(s_t)$	P_h	D_h	h
s_1	000	s_2	001	p_1	D_3	1
		s_3	010	$\overline{p_1 p_2}$	D_2	2
		s_4	011	$\overline{p_1 p_2}$	$D_2 D_3$	3
s_2	001	s_2	001	$p_1 p_2$	D_3	4
		s_4	011	$\overline{p_1 p_2}$	$D_2 D_3$	5
		s_5	100	$\overline{p_1}$	D_1	6
s_3	010	s_2	001	p_2	D_3	7
		s_1	000	$\overline{p_2}$	—	8
s_4	011	s_5	100	p_1	D_1	9
		s_6	101	$\overline{p_1 p_2}$	$D_1 D_3$	10
		s_7	110	$\overline{p_1 p_2}$	$D_1 D_2$	11
s_5	100	s_1	000	1	—	12
s_6	101	s_2	001	p_1	D_3	13
		s_4	011	$\overline{p_1 p_2}$	$D_2 D_3$	14
		s_7	110	$\overline{p_1 p_2}$	$D_1 D_2$	15
s_7	110	s_5	100	1	D_1	16

З цієї таблиці формується СБФ (12), яка для нашого прикладу має такий вигляд:

$$\begin{aligned}
 D_1 &= F_6 \vee [F_9 \vee F_{10} \vee F_{11}] \vee F_{15} \vee F_{16} = \overline{T_1 T_2 T_3 p_1} \vee T_2 T_3 \vee T_1 T_3 \overline{p_1 p_2} \vee T_1 T_2; \\
 D_2 &= [F_2 \vee F_3] \vee F_5 \vee F_{11} \vee [F_{14} \vee F_{15}]; \\
 D_3 &= F_1 \vee F_3 \vee [F_4 \vee F_5] \vee F_7 \vee F_{10} \vee F_{13} \vee F_{14}.
 \end{aligned}
 \tag{21}$$

До СБФ (21) входять терми F_h ($h \in \{1, \dots, H\}$). Кожен терм відповідає одному рядку модифікованої ПСТ. Терми складаються з кон'юнкцій, що відповідають кодам станів $s_m \in S$ та вхідних сигналів P_h . У квадратних дужках показані терми, які можна склеїти. Для FPGA таке склеювання має сенс лише, якщо якась із змінних виключається з усієї формули.

Останній пункт методу, що розглядається, виконується з використанням стандартних пакетів САПР. Для сімейства Virtex-7 використовується пакет Vivado [19]. При цьому кожен елемент схеми представляється таблицею істинності. Наприклад, при $I_o = 5$ така таблиця має 32 рядки.

У нашому прикладі для реалізації системи (21) всі п'ять входів ТЛЕ пов'язані зі змінними з множини $T \cup P$. Як видно з табл. 4, ПСТ має $H=16$ рядків.

Таким чином деякі рядки ПСТ дублюються в таблицях ТЛЕ.

У табл. 5 наведено вміст ТЛЕ, що реалізує функцію $D_r \in D$.

У табл. 5 є стовпець F_h , у якому показаний терм ПСТ, відповідний даному рядку. Наприклад, терм F_{12} повторюється 4 рази, а терм F_3 тільки раз. Оскільки код 111 не використовується для кодування станів, останні 4 рядки не відповідають ніяким термам ПСТ. Таким чином необхідно визначити кожен ТЛЕ.

ТАБЛИЦЯ 5. Таблиця істинності ТЛЕ для функції D1

T_1	T_2	T_3	p_1	p_2	D_1	F_h	T_1	T_2	T_3	p_1	p_2	D_1	F_h
0	0	0	0	0	0	3	1	0	0	0	0	0	12
0	0	0	0	1	0	2	1	0	0	0	1	0	12
0	0	0	1	0	0	1	1	0	0	1	0	0	12
0	0	0	1	1	0	1	1	0	0	1	1	0	12
0	0	1	0	0	1	6	1	0	1	0	0	1	15
0	0	1	0	1	1	6	1	0	1	0	1	0	14
0	0	1	1	0	0	5	1	0	1	1	0	0	13
0	0	1	1	1	0	4	1	0	1	1	1	0	13
0	1	0	0	0	0	8	1	1	0	0	0	1	16
0	1	0	0	1	0	7	1	1	0	0	1	1	16
0	1	0	1	0	0	8	1	1	0	1	0	1	16
0	1	0	1	1	0	7	1	1	0	1	1	1	16
0	1	1	0	0	1	11	1	1	1	0	0	0	*
0	1	1	0	1	1	10	1	1	1	0	1	0	*
0	1	1	1	0	1	9	1	1	1	1	0	0	*
0	1	1	1	1	1	9	1	1	1	1	1	0	*

Блок ВВП наведений у табл. 3, до якої додані стовпці $o_1, o_2 \in O$. Це можливо завдяки наявності двох вільних виходів: $t_o - (G + R_Q) = 2$. Як два виходи можна вибрати будь-яку пару виходів. Якщо умова (16) порушується, необхідно вибрати пару, що вимагає для реалізації найбільшого числа елементів. У нашому прикладі умова (16) виконується і тому можна вибрати два довільних виходи.

Таким чином, схема МПА М1 складається з одного блоку вбудованої пам'яті трьох елементів ТЛЕ, що реалізують ФЗП, і п'яти елементів ТЛЕ, що реалізують виходи $o_n \in O \setminus \{o_1, o_2\}$. Схема має два рівні логіки (рис. 6).

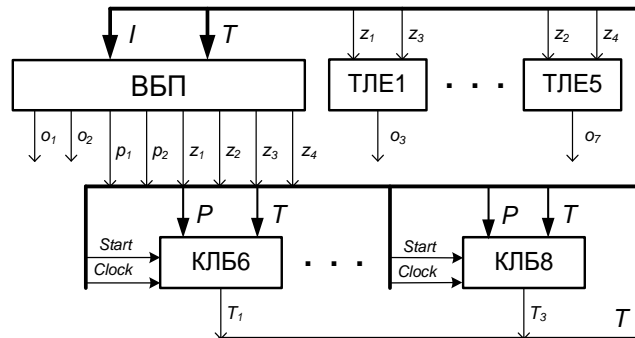


РИС. 6. Логічна схема МПА М1

У схемі (рис. 6) елементи ТЛЕ1 – ТЛЕ5 реалізують частину вихідних функцій. Змінні $T_r \in T$ формуються на виходах тригерів елементів КЛБ6 – КЛБ8. При цьому система (21) реалізується на елементах ТЛЕ6 – ТЛЕ8, що входять до складу КЛБ6 – КЛБ8 відповідно. Ці КЛБ включають регістр РКС, керований імпульсами Start та Clock. Зазначимо, що функції $o_n \in O$ і $D_r \in D$ формуються одночасно. За такої організації схеми необхідний додатковий регістр зберігання вихідних сигналів [18]. Він показаний на схемі (рис. 6).

Висновок. Для зменшення площі схем на базі FPGA необхідно використовувати вбудовані блоки пам'яті, які замінюють значну кількість елементів табличного типу [6]. Однак достатня кількість блоків ВВП може бути відсутньою через їх використання для реалізації різних табличних функцій. В цьому випадку схема будується у вигляді композиції блоків ВВП та елементів ТЛЕ. У цій роботі пропонується метод оптимізації МПА Мілі, що реалізується у змішаному елементному базисі.

Пропонований метод ґрунтується на спільному використанні двох методів структурної декомпозиції. Метод заміни вхідних змінних зменшує вимоги до входів елементів, що реалізують функції збудження пам'яті МПА. Метод кодування вихідних наборів зменшує вимоги до входів елементів, що реалізують систему вихідних функцій.

При виконанні певних умов схема МПА має два рівні логіки та включає $N + R$ елементів ТЛЕ. Ця кількість може бути зменшеною, якщо ВВП має вільні виходи.

Ми дослідили ефективність запропонованого методу для стандартних автоматів [20], що реалізуються в базисі мікросхем сімейства Virtex-7. Реалізація схеми проводилася за допомогою пакету Vivado [19]. Застосування блоку ВВП дозволило зменшити кількість блоків ТЛЕ у середньому на 14 % – 18 % порівняно зі схемами, що складаються лише з ТЛЕ. Для FPGA сімейства Virtex-7 число входів $I_o = 6$. Цієї величини було достатньо для однорівневої реалізації системи виходів. Для

складних автоматів ($R \geq 7$, $L \geq 19$) схеми формування ФЗП мали до трьох рівнів елементів ТЛЕ, що в середньому в два рази менше, ніж в еквівалентних автоматах, реалізованих без ВБП.

У подальших дослідженнях ми плануємо розробляти методи оптимізації схем МПА. Головна мета – зменшення кількості рівнів елементів у схемі. Для цього можна використовувати методи подвійного кодування станів [19, 20]. Крім того, ми плануємо застосувати ці методи до суміщених автоматів [21, 22].

Список літератури

1. Baranov S. Logic synthesis for control automata. Dordrecht: Kluwer Academic Publishers, 1994. 312 p.
2. DeMicheli G. Synthesis and optimization of digital circuits. New York: McGraw-Hill, 1994. 576 p.
3. Skliarova I., Sklyarov V., Sudnitson A. Design of FPGA-based circuits using hierarchical finite state machines. Tallinn: TUT Press, 2012. 240 p.
4. Czerwinski R., Kania D. Finite state machines logic synthesis for complex programmable logic devices. Berlin: Springer, 2013. 172 p.
5. Wiśniewski R., Bazydło G., Szcześniak P., Wojnakowski M. Petri net-based specification of cyber-physical systems oriented to control direct matrix converters with space vector modulation. IEEE Access, 2019. Vol. 7. 23407–23420.
6. Sklyarov V., Skliarova I., Barkalov A., Titarenko L. Synthesis and optimization of FPGA-based systems. Berlin: Springer, 2014. 432 p. https://doi.org/10.1007/978-3-319-04708-9_6.
7. Tiwari A., Tomko K. Saving power by mapping finite state machines into embedded memory blocks in FPGAs. Proc. Design, Automation and Test in Europe Conference and Exhibition (Paris, France, 6–20 Feb. 2004). 2004. Vol. 2. P. 916–921.
8. Rawski M., Tomaszewicz P., Borowski G., Luba T. Logic synthesis method of digital circuits designed for implementation with embedded memory blocks on FPGAs. In: Design of Digital Systems and Devices. Lecture Notes in Electrical Engineering. Adamski M., Barkalov A., Wegrzyn M. (Eds.). Vol. 79. Berlin: Springer, 2011. P. 121–144.
9. Maxfield C. The design warrior's guide to FPGAs. Orlando: Academic Press, 2004. 542 p.
10. Grout I. Digital systems design with FPGAs and CPLDs. Amsterdam: Elsevier, 2008. 784 p. <https://doi.org/10.1016/B978-0-7506-8397-5.X0001-3>.
11. Ruiz-Rosero J., Ramirez-Gonzalez G., Khanna R. Field Programmable Gate Array Applications – A Scientometric Review. Computation. 2019. 7 (4), 63. <https://doi.org/10.3390/computation7040063>
12. Garcia-Vargas L., Senhaji-Navarro R. Finite state machines with input multiplexing: A performance study. IEEE Transactions on CAD of Integrated Circuits and Systems. 2015. Vol. 34, Iss. 5. P. 867–871.
13. Sklyarov V. Synthesis and Implementation of RAM-based Finite States Machines in FPGAs. in Proceeding of Field-Programmable Logic and Applications: The Roadmap to Reconfigurable Computing. Villach: Springer-Verlag, 2000. P. 718–728.
14. Kuon I., Tessier R., Rose J. FPGA Architecture: Survey and Challenges. Foundations and Trends in Electronic Design Automation. 2008. Vol. 2, No. 2. P. 135–253.
15. Kubica M., Opara A., Kania D.. Technology Mapping for LUT- based. FPGA. Berlin: Springer, 2021.
16. Barkalov O., Titarenko L., and Barkalov Jr., A Structural Decomposition as a tool for the optimization of an FPGA–based implementation of a Mealy FSM. *Cybernetics and Systems Analysis*. 2012. Vol. 48, No. 2. P. 313–322.
17. Barkalov A., Titarenko L., Mielcarek K. Hardware reduction for LUT–based Mealy FSMs. *International Journal of Applied Mathematics and Computer Science*. 2018. P. 595–607. <https://doi.org/10.2478/amcs-2018-0046>
18. Barkalov A., Titarenko L., Mielcarek K. Improving characteristics of LUT–based Mealy FSMs. *International Journal of Applied Mathematics and Computer Science*. 2020. 30 (4). P. 745–759.
19. Vivado Design Suite. <https://www.xilinx.com/products/design-tools/vivado.html> (звернення: 15.01.2023)
20. Yang S. Logic synthesis and optimization benchmarks user guide. Version 3.0. Techn. Rep. Microelectronics Center of North Carolina, 1991. 43 p.
21. Баркалов А.А., Титаренко Л.А., Визор Я.Е., Матвиенко А.В., Горина В.В. Уменьшение числа LUT элементов в схеме совмещенного автомата. *Управляющие системы и машины*. 2016. № 3. С. 16–22. <https://doi.org/10.15407/usim.2016.03.016>
22. Баркалов А.А., Титаренко Л.А., Визор Я.Е., Матвиенко А.В. Уменьшение аппаратурных затрат в совмещенных автоматах. *Управляющие системы и машины*. 2017. № 4. С. 43–50. <https://doi.org/10.15407/usim.2017.04.043>

Одержано 20.07.2023

Баркалов Олександр Олександрович,

доктор технічних наук, професор Університету Зеленогурського (Польща),

<https://orcid.org/0000-0002-4941-3979>

A.Barkalov@iie.uz.zgora.pl

Тітаренко Лариса Олександрівна,

доктор технічних наук, професор Університету Зеленогурського (Польща),

професор Харківського національного університету радіоелектроніки,

<https://orcid.org/0000-0001-9558-3322>

L.Titarenko@iie.uz.zgora.pl

Головін Олександр Миколайович,

кандидат технічних наук, старший науковий співробітник

Інституту кібернетики імені В.М. Глушкова НАН України, Київ,

<https://orcid.org/0000-0002-0279-812X>

o.m.golovin.1@gmail.com

Матвієнко Олександр Володимирович,

науковий співробітник Інституту кібернетики імені В.М. Глушкова НАН України, Київ.

<https://orcid.org/0000-0003-1838-1422>

avmatv@ukr.net

УДК 004.274

О.О. Баркалов¹, Л.О. Тітаренко^{1,2}, О.М. Головін³, О.В. Матвієнко^{3*}

Оптимізація схеми автомата Мілі у змішаному базисі

¹ Університет Зеленогурський, Зелена Гура, Польща

² Харківський національний університет радіоелектроніки, Харків, Україна

³ Інститут кібернетики імені В.М. Глушкова НАН України, Київ

* Листування: avmatv@ukr.net

Вступ. Пристрій управління це один із найважливіших блоків будь-якої цифрової системи. Основна функція пристрою управління – координування взаємодії інших блоків системи. Тому характеристики схеми пристрою управління мають значний вплив на якість системи в цілому.

Для представлення закону функціонування пристрою керування використовуються моделі мікропрограмного автомата (МПА) Мура та Мілі. При синтезі схем МПА необхідно вирішити низку оптимізаційних завдань: зменшення апаратних витрат, підвищення швидкодії, мінімізація споживаної потужності, спільна оптимізація апаратних швидкісних характеристик. Методи вирішення цих завдань значною мірою залежать від використовуваного елементного базису. В даний час одним з основних базисів, в якому реалізуються сучасні цифрові системи, є базис FPGA.

Основні блоки у складі FPGA це логічні блоки, що конфігуруються, програмована матриця між'єднань, дерево синхронізації і програмовані входи-виходи. Для реалізації схем МПА можна використовувати логічні блоки, що конфігуруються, двох типів: табличні логічні елементи (ТЛЕ), і вбудовані блоки пам'яті (ВБП), що володіють властивістю реконфігурації. Проте ВБП широко використовуються реалізації різних операційних блоків цифрових систем. Тому розробник схеми пристрою управління може використовувати обмежену кількість таких блоків пам'яті.

Мета статті. У статті розглянуті питання синтезу МПА, коли є обмежена кількість "вільних" блоків ВБП. І тут схема мікропрограмного автомата представляється мережею з блоків ВБП і ТЛЕ. Запропоновано метод синтезу МПА з оптимізацією числа ТЛЕ, коли у схемі мікропрограмного автомата можна використовувати лише один ВБП.

Пропонований метод заснований на використанні вбудованого блоку пам'яті, який здійснює заміну вхідних змінних та кодування виходів автомата.

Результати. Дослідження ефективності запропонованого методу проводилися на стандартних автоматах. Як елементний базис використовувалися FPGA сімейства Virtex-7 фірми Xilinx. Для реалізації

МПА застосовано пакет Vivado. Результати досліджень показали, що використання блоку ВВП дозволило зменшити кількість блоків ТЛЕ в середньому на 14 % – 18 % порівняно зі схемами, що складаються лише з ТЛЕ. Для FPGA сімейства Virtex-7 було достатньо числа входів ТЛЕ $I_o=6$ для однорівневої реалізації системи виходів.

Висновки. Ефективність запропонованого методу дозволяє рекомендувати його для використання при синтезі мікропрограмних автоматів в умовах граничного обмеження числа БВП.

Ключові слова: автомат Мілі, синтез, кодування входів, кодування наборів виходів.

UDC 004.274

Alexander Barkalov¹, Larysa Titarenko^{1,2}, Oleksandr Golovin³, Oleksandr Matvienko^{3*}

Optimization of a Mealy Automaton Circuit in a Mixed Element Basis

¹ University of Zielona Gora, Poland

² Kharkiv National University of Radio Electronics, Ukraine

³ V.M. Glushkov Institute of Cybernetics of the NAS of Ukraine, Kyiv

* Correspondence: avmatv@ukr.net

Introduction. The control device is one of the most important blocks of any digital system. The main function of the control device is to coordinate the interaction of the remaining units of the system. Therefore, the characteristics of the control device circuit have a significant impact on the quality of the overall system.

To represent the law of functioning of the control device, the models of the microprogrammed automaton (MPA) by Moore and Mealy are used. When synthesizing MPA circuits, it is necessary to solve a number of optimization problems: reducing hardware costs, increasing performance, minimizing power consumption, and jointly optimizing hardware-time characteristics. Methods for solving these problems largely depend on the elemental basis used. Currently, one of the main bases in which modern digital systems are implemented is the FPGA.

The main blocks in the FPGA are configurable logic blocks, a programmable interconnect matrix, a timing tree, and programmable inputs and outputs. To implement MPA schemes, two types of configurable logic blocks can be used: tabular logic elements (TLE) and built-in memory blocks (VBP), which have the property of reconfiguration. However, VBPs are widely used to implement various operating blocks of digital systems. Therefore, the controller circuit designer can use a limited number of such memory blocks.

Purpose of the article. The article deals with the issues of MPA synthesis when there are a limited number of "free" blocks of EBP. In this case, the microprogram automaton circuit is represented by a network consisting of VBP and TLE blocks. A method for the synthesis of a microprogram automaton with optimization of the number of TLEs is proposed when only one VBP can be used in the microprogram automaton circuit.

The proposed method is based on the use of a built-in memory block that performs the replacement of input variables and the coding of the automaton outputs.

Results. Studies of the effectiveness of the proposed method were carried out on standard machines. FPGAs of the Virtex-7 family from Xilinx were used as the elemental basis. To implement the proposed MPA, the Vivado package was used. The results of the research showed that the use of the VBP block made it possible to reduce the number of SLE blocks by an average of 14 % – 18 % compared to schemes consisting only of SLE. For the Virtex-7 family FPGA, the number of TLE inputs $I_o=6$ was sufficient for a single-level implementation of the output system.

Conclusions. The effectiveness of the proposed method makes it possible to recommend it for use in the synthesis of microprogram automata under conditions of an extremely limited number of VBPs.

Keywords: Mealy automaton, synthesis, coding of inputs, coding of sets of outputs.