

## РОЗДІЛЕННЯ ВХІДНИХ ЗМІННИХ ДЛЯ ОПТИМІЗАЦІЇ СХЕМИ АВТОМАТА МІЛІ

**Вступ.** Один із найважливіших блоків цифрових систем – це пристрій управління (ПУ) [1, 2]. У практиці інженерного проектування поведінка ПУ часто визначається за допомогою моделі мікропрограмного автомата (МПА) Мілі [3, 4]. Методи синтезу МПА залежать від особливостей цієї моделі. Особливість МПА Мілі – залежність систем булевих функцій (СБФ), що задають схему від входів та станів [4]. У цій статті ми враховуємо цю особливість для оптимізації характеристик схеми у базисі мікросхем FPGA (field-programmable logic arrays) [5, 6].

В даний час базис FPGA часто використовується для реалізації різноманітних цифрових систем [7, 8]. При використанні будь-якого базису виникає необхідність оптимізації характеристик схеми систем, що реалізуються. Для типових цифрових блоків (суматори, лічильники, зсувачі, помножувачі тощо) є стандартні бібліотечні рішення [4]. Однак схеми ПУ відрізняються залежністю від виду реалізованого алгоритму управління [9, 10]. Тому для кожного алгоритму схеми ПУ необхідно синтезувати заново. Це викликає необхідність розробки ефективних методів синтезу, які дозволяють оптимізувати схему ПУ. Методи оптимізації схем ПУ мають змінюватись зі зміною елементного базису [2, 3].

У цій роботі пропонується метод зменшення апаратних витрат у схемі МПА Мілі, що реалізується у базисі FPGA. При цьому розглядається завдання реалізації схеми у змішаному елементному базисі. Під змішаним базисом розуміється спільне використання елементів табличного типу LUT (look-up table) та блоків пам'яті ЕМВ (embedded memory blocks) [11]. Така ситуація цілком можлива, оскільки блоки ЕМВ інтенсивно використовуються при реалізації різних операційних блоків цифрових систем [6, 12].

### Реалізація схеми МПА у базисі FPGA

У базисі FPGA схеми будуються з урахуванням взаємозалежних блоків CLB (configurable logic blocks). Схеми МПА мають дві особливості: описуються системами булевих функцій і мають послідовний характер,

*Запропоновано метод, що дозволяє використовувати блок вбудованої пам'яті для заміни вхідних елементів автомата Мілі. Метод орієнтований на базис FPGA та змішану реалізацію схеми автомата. Метод заснований на розділенні вхідних змінних так, що частина входів замінюється з використанням елементів табличного типу LUT. Показано умови застосування запропонованого методу. Наведено результати досліджень ефективності запропонованого методу для стандартних автоматів із використанням мікросхем сімейства Virtex-7 та промислового пакету Vivado.*

**Ключові слова:** мікропрограмний автомат синтезу, FPGA, ЕМВ, LUT, заміна входів.

тобто мають у своєму складі пам'ять [2, 3]. Як зазначалося, для реалізації схем МПА використовуються CLB, які складаються з LUT і EMB [13, 14]. Розглянемо особливості цих блоків на прикладі сімейства Virtex-7 фірми Xilinx [15].

Елемент LUT – це схема, що складається з однорозрядних осередків SRAM, пов'язаних із мультиплексорами. Для керування мультиплексорами використовуються SL управляючих входів. Один LUT може реалізувати довільну функцію алгебри логіки (ФАЛ), що залежить від не більше SL змінних.

У сімействі Virtex-7 використовуються елементи LUT, які мають  $S_L$  входів та один вихід, де  $S_L = 6$ . Кожен CLB складається з чотирьох елементів LUT, восьми програмованих тригерів, трьох спеціалізованих мультиплексорів та інших ресурсів [16]. Вихід елемента LUT зв'язується із тригером, що дозволяє організувати пам'ять схеми. На вихід CLB можуть передаватися як виходи LUT (комбінаційна схема), так і виходи тригерів (запам'ятовуюча схема). У середині CLB є система швидких міжз'єднань, що дозволяє реалізувати два еквівалентні елементи LUT, що мають  $S_L = 7$ , або один LUT із  $S_L = 8$  [17]. Таке швидкісне поєднання практично не впливає на такт циклу схеми, що реалізується [17, 18].

Блоки EMB це блоки пам'яті, що конфігуруються, в яких можна змінювати число адресних входів  $S_A$  і розрядність осередків  $t_F$ . Місткість пам'яті  $V_0$  залишається незмінною [11] і дорівнює  $V_0 = 2^{S_A} \times t_F$ .

Для EMB фірми AMD Xilinx пари  $\langle S_A, t_F \rangle$  вибрані для  $V_0 = 32$  Кбіт [15]. Будемо називати кожен таку пару конфігурацією. Для Virtex-7 є такі конфігурації:  $\langle 15, 1 \rangle$ ,  $\langle 14, 2 \rangle$ ,  $\langle 13, 4 \rangle$ , ...,  $\langle 9, 64 \rangle$ . Таким чином, при зменшенні  $S_A$  на 1 число виходів подвоюється. Очевидно, що один блок EMB може реалізувати СБФ, що залежить від  $S_A$  змінних елементів, що мають до  $t_F$ .

Автомат Мілі може бути представлений вектором  $S = \langle A, X, Y, \delta, \lambda, a_1 \rangle$  [2], де  $A = \{a_1, \dots, a_M\}$  – множина внутрішніх станів,  $X = \{x_1, \dots, x_L\}$  – множина вхідних змінних,  $Y = \{y_1, \dots, y_N\}$  – множина виходів. Функція переходів  $\delta$  ставить у відповідність кожній парі  $\langle$  поточний стан, вхідний сигнал  $\rangle$  стан переходу  $a_s \in A$ . Функція виходів  $\lambda$  ставить у відповідність подібній парі набір виходів  $Y_q \subseteq Y$  ( $q \in \{1, \dots, Q\}$ ). Функції  $\delta$  і  $\lambda$  визначають  $H$  подібних пар. У початковий момент часу автомат знаходиться у початковому стані  $a_1 \in A$ .

Ми розглядаємо випадок представлення автомата його таблицею переходів [4]. Ця таблиця має стовпці: поточний стан ( $a_m \in A$ ); стан переходу ( $a_s \in A$ ); вхідний сигнал  $X_h$ , що визначає перехід  $\langle a_m, a_s \rangle$ ; набір вихідних сигналів (НВС) ( $Y_h$ ), що формується на переході  $\langle a_m, a_s \rangle$ ;  $h$  – номер рядка. Рядок  $h$  таблиці визначає  $h$ -й перехід МПА ( $h = \{1, H\}$ ). Функції  $\delta$  і  $\lambda$  знаходять своє відображення у СБФ

$$\Phi = \Phi(T, X); \quad (1)$$

$$Y = Y(T, X). \quad (2)$$

У СБФ (1) – (2) множина  $T = \{T_1, \dots, T_R\}$  складається з змінних  $a_m \in A$ , які кодуються кодами  $K(a_m)$ . Мінімальна розрядність кодів визначається як

$$R = \lceil \log_2 M \rceil. \quad (3)$$

Коди зберігаються у пам'яті МПА, що складається з  $R$  тригерів. Як правило [9], тригери мають інформаційні входи типу  $D$ . Для зміни стану пам'яті використовуються функції збудження пам'яті (ФЗП), що утворюють множину  $\Phi = \{D_1, \dots, D_R\}$ .

Пам'ять МПА це регістр  $RG$ , входи якого пов'язані з ФЗП  $D_r \in \Phi$ . У початковий момент до  $RG$  записується нульовий код стану  $a_1 \in A$ . Для цього використовується одиночний імпульс  $Start$ . Момент запису коду  $RG$  визначається імпульсом синхронізації  $Clock$ . Системи (1) – (2) визначають МПА  $S_1$  (рис.1).

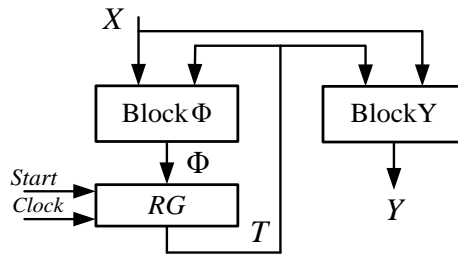


РИС. 1. Структурна схема МПА  $S_1$

У МПА  $S_1$  BlockΦ реалізує СБФ (1), BlockY – систему (2). Регістр  $RG$  має входи  $\Phi$ ,  $Start$ ,  $Clock$  і виходи  $T$ . Для реалізації схем блоків BlockΦ і BlockY використовуються різні методи оптимізації [19, 20], Особливості цих методів визначаються характеристиками елементного базису. Далі розглянемо методи, спрямовані на оптимізацію числа LUT.

#### Оптимізація схем МПА у базисі FPGA

Оптимізація схем на базі LUT пов'язана зі зменшенням їхньої кількості, міжз'єднань і рівнів логіки [11]. Нехай функція  $f_i \in \Phi \cup Y$  залежить від  $NA(f_i)$  аргументів. Поки для всіх функцій виконується умова

$$NA(f_i) \leq S_L \tag{4}$$

схема МПА має один рівень та  $R+N$  елементів. Якщо умова (4) виконується лише для деяких функцій, то схема є багаторівневою.

Для оптимізації багаторівневих схем використовують методи функціональної декомпозиції (ФД) [21]. При цьому функція  $f_i$  сприймається як сукупність часткових функцій  $f_i^1, \dots, f_i^K$ , композиція яких є вихідною функцією. Для зменшення числа LUT необхідно мінімізувати величину  $K$  [21]. Такий підхід призводить до багаторівневої схеми зі складною системою міжз'єднань [11].

Радикальним методом оптимізації є заміна всіх елементів LUT одним блоком ЕМВ [22]. Такий підхід можливий за наявності такої конфігурації  $\langle S_0, t_0 \rangle$ , для якої виконуються умови  $S_0 \geq L + R$  і  $t_0 \geq N + R$ .

Зазначимо, що ці умови можливі для край простих автоматів [2, 3]. При цьому домінує умова (6), тому що для сімейства ЕМВ Virtex-7 максимальне значення  $S_0 = 15$ . Однак у цьому випадку  $t_0 = 1$ .

Число елементів LUT можна зменшити за рахунок використання методів структурної декомпозиції (СД) [11]. Тут схема представляється як композиція кількох логічних блоків, кожен із яких має унікальну систему входів і виходів. Один із найбільш популярних це метод заміни вхідних змінних (ЗВЗ) [2]. Розглянемо цей метод.

Нехай переходи зі стану  $a_m \in A$  залежать від  $L_m$  змінних  $x_l \in X$ . Як показують дослідження [23], значення  $L_m$  знаходиться у межах від 0 (безумовні переходи) до трьох. Знайдемо кількість змінних, достатню для ЗВЗ:

$$G = \max(L_1, \dots, L_M). \quad (5)$$

Тепер входи  $x_l \in X$  можна замінити змінними  $p_g \in P$ , де  $P = \{p_1, \dots, p_G\}$ . Для здійснення ЗВЗ необхідно побудувати СБФ

$$P = P(T, X). \quad (6)$$

Заміна входів призводить до заміни СБФ (1), (2) наступними системами:

$$\Phi = \Phi(T, P); \quad (7)$$

$$Y = Y(T, P). \quad (8)$$

Системи (6) – (8) визначають структурну схему МПА  $S_2$  (рис. 2, а).

У схемі рис. 2, а символ LUTerP означає блок, що складається з елементів LUT. Блок LUTerP реалізує СБФ (6). Блок LUTerTY реалізує СБФ (7), (8). На рис. 2, а показано особливість блоків CLB, заснованих на LUT: реєстр кодів станів  $RG$  є розосередженим [11].

Схема LUTerP складається з  $G$  мультиплексорів, реалізованих за допомогою LUT і вбудованих мультиплексорів CLB. Функція  $p_g \in P$  залежить від  $L(p_g)$  вхідних змінних. При виконанні умови  $L(p_g) + R > S_L$  схема блоку LUTerP – багаторівнева.

Ми розглядаємо випадок, коли для реалізації схеми МПА розробник може використовувати лише один блок ЕМВ. Якщо є комбінація  $\langle S_0, t_0 \rangle$ , для якої виконуються умови

$$S_0 = L + R, \quad t_0 \geq G, \quad (9)$$

то для заміни входів доцільно використати ЕМВ. Це призводить до МПА  $S_3$  (рис. 2, б).

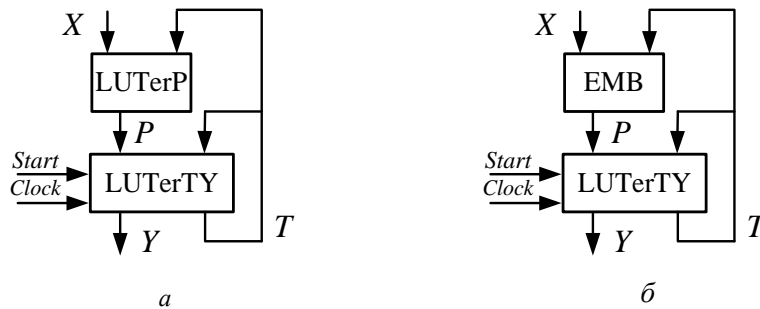


РИС. 2. Структурні схеми МПА: а –  $S_2$  та б – МПА  $S_3$

Зазначимо, що  $(t_0 - G)$  функцій  $y_n \in Y$  також можуть бути реалізовані на ЕМВ. Це зменшує кількість LUT у блоці LUTerTY.

Параметри ЕМВ визначають граничні значення суми  $L + R$ , перевищення яких призводить до неможливості застосування моделі  $S_3$ . Якщо  $G = 4$ , то для Virtex-7 необхідно вибрати конфігурацію  $\langle 13, 4 \rangle$ . Таким чином,  $L + R$  має не перевищувати 13. Наш аналіз бібліотеки [24] показує,

що умова  $L + R \leq 13$  виконується лише для 34 % автоматів цієї бібліотеки. Зазначимо, що у бібліотеці [24] зібрані реальні МПА. Таким чином, модель  $S_3$  можна застосувати далеко не завжди.

У цій роботі ми пропонуємо модифікацію моделі  $S_3$ . Модифікована модель передбачає використання ЕМВ для заміни лише частини входів  $x_i \in X$ .

**Основна ідея запропонованого методу**

Якщо число входів ЕМВ є недостатнім для заміни всіх змінних,  $q$  з них мають бути замінені з використанням елементів LUT. Нехай для конфігурації  $\langle S_0, t_0 \rangle$  виконуються такі умови:  $S_0 + q = L + R$ ,  $G \leq t_0$ .

Розіб'ємо множину  $X$  на підмножини  $X^1$ ,  $X^2$ , де  $|X^1| = L - q$ ,  $|X^2| = q$ . Входи  $x_i \in X^1$  можна замінити з використанням ЕМВ, а входи  $x_i \in X^2$  – з використанням LUT.

СБФ (6) можна представити у вигляді системи

$$P^1 = P^1(T, X^1); \tag{10}$$

$$P^2 = P^2(T, X^2). \tag{11}$$

Очевидно, що  $P = P^1 \cup P^2$ . У кращому випадку ми отримуємо співвідношення

$$|P^1 \cap P^2| = 0. \tag{12}$$

У цьому випадку ми пропонуємо МПА  $S_4$  (рис. 3).

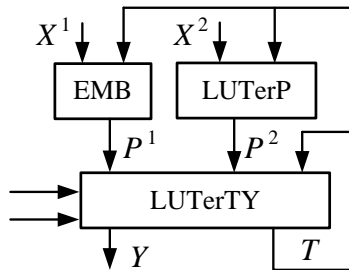


РИС. 3. Структурна схема МПА  $S_4$

У МПА  $S_4$  блок ЕМВ реалізує СБФ (10), а LUTerP – СБФ (11). Блок LUTerTY реалізує СБФ (7) – (8). Схема має два рівні логіки, якщо  $G + R \leq S_L$ .

Якщо умова (12) порушується, то до множин  $P^1$  і  $P^2$  входить до  $G$  однакових змінних  $p_g \in P$ . Нехай ці змінні утворюють множину  $P^0$ . Позначимо символом  $P^3$  множину  $P^1 \setminus P^0$ , а символом  $P^4$  множину  $P^2 \setminus P^0$ . Змінні формуються як блоком ЕМВ, так і блоком LUTerP. Це призводить до моделі МПА  $S_5$  (рис. 4).

У МПА  $S_5$  додано блок LUTerOR, що реалізує СБФ

$$P^0 = P^0(P_E^0, P_L^0). \tag{13}$$

Тут  $P_E^0$  – змінні  $p_g \in P^0$ , які формуються блоком ЕМВ, а  $P_L^0$  – блоком LUTerP. Вочевидь,  $P = P^0 \cup P^3 \cup P^4$ . У кращому випадку схема МПА  $S_5$  має три рівні логіки.

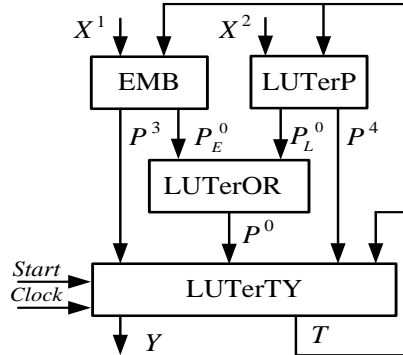


Рис. 4. Структурна схема МПА  $S_5$

Модель  $S_5$  має більш загальний характер. Тому в цій статті пропонується метод синтезу МПА Мілі  $S_5$ . Метод включає такі етапи:

1. Формування таблиці заміни змінних.
2. Кодування станів  $a_m \in A$  кодами  $K(a_m)$ .
3. Формування множин  $X^1, X^2, P^1, P^1, P^2, P^0, P^3, P^4$  з урахуванням показників блоку ЕМВ.
4. Формування таблиці блоку ЕМВ.
5. Формування СБФ (11), що задає блок LUTerP.
6. Формування СБФ(13), що задає блок LUTerOR.
7. Формування таблиці блоку LUTerTY та СБФ (7), (8).
8. Реалізація схеми МПА у заданому елементному базисі.

Ми вважаємо, що МПА Мілі задано таблицею переходів. Очевидно, при використанні будь-якої іншої форми завдання поведінки автомата необхідно здійснити перехід до цієї таблиці. Розглянемо приклад використання моделі під час синтезу схеми МПА А1.

#### Приклад синтезу схеми МПА

Нехай автомат А1 задано таблицею переходів (табл. 1). Для реалізації схеми МПА використовується тільки один блок ЕМВ і довільне число блоків LUT з числом входів  $S_L = 5$ . Можливі конфігурації блоку ЕМВ визначаються парами  $\langle 11, 1 \rangle, \langle 10, 2 \rangle, \dots, \langle 8, 8 \rangle$ .

З табл. 1 можна знайти такі множини:  $A = \{a_1, \dots, a_7\}$ ,  $X = \{x_1, \dots, x_9\}$ ,  $Y = \{y_1, \dots, y_6\}$ , що дає  $M = 7$ ,  $L = 9$  і  $N = 6$ . Аналіз підтаблиць для кожного зі станів дає  $L_1 = L_2 = L_4 = L_6 = 2$ ,  $L_3 = L_5 = 1$ ,  $L_7 = 0$ . З формули (5) отримуємо  $G = 2$ ,  $P = \{p_1, p_2\}$ .

Таблиця заміни вхідних змінних МПА А1 формується за допомогою методики [2]. Рядки таблиці ЗВЗ позначені змінними  $p_g \in P$ , а стовпці – станами  $a_m \in A$ . Якщо змінна  $p_g \in P$  замінює вхід  $x_l \in X$  у стані  $a_m \in A$ , то на перетині рядка  $p_g$  і стовпця  $a_m$  записується  $x_l$ . Один з варіантів заміни наведений у табл. 2.

Для  $M = 7$  формули (3) маємо  $R = 3$ . Це дає множини  $T = \{T_1, T_2, T_3\}$  і  $\Phi = \{D_1, D_2, D_3\}$ . Закодуємо множини тривіальним чином:  $K(a_1) = 000, \dots, K(a_7) = 110$ .

Оскільки  $G = 2$ , то з (9) випливає, що потрібно вибрати конфігурацію  $\langle 11, 2 \rangle$ . Маємо  $R+L = 12$ , отже  $q = 1$  і один із входів  $x_i \in X$  потрібно перетворити на блоці LUTerP. Виберемо  $x_8$  як вхід. Тоді отримаємо:  $X^1 = \{x_1, \dots, x_7, x_9\}$ ,  $X^2 = \{x_8\}$ ,  $P^1 = \{p_1, p_2\}$  і  $P^2 = \{p_2\}$ . Маємо  $P^1 \cap P^2 = \{p_2\}$ , а це призводить до необхідності формування множин  $P^3 = \{p_1\}$ ,  $P_E^0 = P_L^0 = \{p_2\}$ ,  $P^4 = \emptyset$ .

ТАБЛИЦЯ 1. Таблиця переходів МПА А1

$a_m$	$a_s$	$X_h$	$Y_h$	$h$
$a_1$	$a_2$	$x_1$	$y_1 y_2$	1
	$a_3$	$\overline{x_1 x_2}$	$y_3$	2
	$a_3$	$\overline{x_1 x_2}$	$y_4$	3
$a_2$	$a_3$	$x_3$	$y_3 y_4$	4
	$a_4$	$\overline{x_3 x_4}$	$y_5$	5
	$a_5$	$\overline{x_3 x_4}$	$y_1 y_2$	6
$a_3$	$a_4$	$x_5$	$y_6$	7
	$a_6$	$\overline{x_5}$	$y_3 y_5$	8
$a_4$	$a_2$	$x_6$	$y_2$	9
	$a_5$	$\overline{x_6 x_7}$	$y_1 y_4$	10
	$a_6$	$\overline{x_6 x_7}$	$y_6$	11
$a_5$	$a_6$	$x_8$	$y_2 y_6$	12
	$a_1$	$\overline{x_8}$	—	13
$a_6$	$a_7$	$x_1$	$y_1 y_3 y_5$	14
	$a_3$	$\overline{x_1 x_9}$	$y_2 y_6$	15
	$a_4$	$\overline{x_1 x_9}$	$y_4$	16
$a_7$	$a_1$	1	$y_3 y_5$	17

ТАБЛИЦЯ 2. Таблиця ЗВЗ МПА А1

$a_m \backslash p_g$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$a_7$
$p_1$	$x_1$	$x_3$	$x_5$	$x_6$	—	$x_1$	—
$p_2$	$x_2$	$x_4$	—	$x_7$	$x_8$	$x_9$	—

Змінні  $x_i \in X^1$  перетворюються блоком ЕМВ на змінні  $p_g \in P^1 = P$  (для нашого прикладу). Таблиця блоку ЕМВ має  $R + |X^1|$  стовпців, що визначають адресу комірки пам'яті, і  $G$  стовпців, що

визначають її вміст. Таблиця має  $H_E = 2^{R+|X^1|}$  рядків. Кожен стан представлений за допомогою  $H_E(a_m) = 2^{|X^1|}$  рядків. Таким чином, блок ЕМВ це таблиця істинності [19] системи (6).

У нашому випадку  $H_E = 2^{11} = 2048$  і  $H(a_m) = 2^8 = 256$ . Таблиця блоку ЕМВ формується тривіальним чином. Перші 8 рядків для прикладу наведені в табл. 3. Маємо  $K(a_m) = 000$ , тобто представлено заміну для стану  $a_1 \in A$ . Столпець  $v$  включає адресу комірки.

ТАБЛИЦЯ 3. Таблиця блоку ЕМВ МПА А1

$T_1$	$T_2$	$T_3$	$T_1$	$x_8$	$x_7$	$x_6$	$x_5$	$x_4$	$x_3$	$x_2$	$x_1$	$p_1$	$p_2$	$v$
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
0	0	0	0	0	0	0	0	0	0	1	0	0	1	2
0	0	0	0	0	0	0	0	0	0	1	1	1	1	3
0	0	0	0	0	0	0	0	0	1	0	0	0	0	4
0	0	0	0	0	0	0	0	0	1	0	1	1	0	5
0	0	0	0	0	0	0	0	0	1	1	0	0	1	6
0	0	0	0	0	0	0	0	0	1	1	1	1	1	7

Як впливає з табл. 3, для рядків 0 – 7 маємо  $p_1 = x_1$ ,  $p_2 = x_2$ . Інші входи – несуттєві й їх значення ігнорується.

Система (11) формується з табл. 2. Для оптимізації цієї формули можна використовувати код 110, оскільки стан  $a_7$  не бере участі у ЗВЗ. Можна також використовувати набір 111, який не використовується для кодування станів  $a_m \in A$ . В результаті маємо рівняння  $p_2^L = T_1 T_2 x_8$ . Індекс "L" у цій формулі додано, щоб підкреслити, що рівняння реалізовано за допомогою блоку LUTerP.

У таблиці блоку ЕМВ розглянуто ЗВЗ для всіх станів  $a_m \in A$ . У нашому випадку необхідно для всіх 256 рядків, що відповідають коду  $K(a_s)$ , записати 0 у стовпці  $p_2$ . Зазначимо, що блок ЕМВ формує змінну  $p_2^E$ , де індекс "E" показує цей факт.

Для нашого прикладу схема для функції  $p_2^L = T_1 T_2 x_8$  реалізується на одному елементі LUT. Цього можна досягнути і без оптимізації, оскільки  $|X^2| + R \leq S_L$ . Однак оптимізація зменшує кількість між'єднань у схемі МПА [18].

Блок LUTerOR реалізує СБФ (13). З урахуванням введених позначень, маємо  $P_E^0 = \{p_2^E\}$  і  $P_L^0 = \{p_2^L\}$ . Таким чином, блок LUTerOR може бути представленим рівнянням  $p_2 = p_2^E \vee p_2^L$ .

Для автоматів будь-якої складності кожне з рівнянь (13) має лише два аргументи. При цьому блок LUTerOR складається з  $|P^0|$  елементів LUT.

Блок LUTerTY має  $G+R$  входів. У нашому випадку  $G=2$ ,  $R=3$ ,  $S_L=6$ . Таким чином, блок LUTerTY реалізується однорівневою схемою. Таблиця блоку LUTerTY будується з урахуванням початкової таблиці переходів, таблиці заміни входів і кодів станів  $K(a_m)$ .

Таблиця блоку LUTerTY має стовпці  $a_m, K(a_m), a_s, K(a_s), P_h, \Phi_h, Y_h, h$  і по суті це є модифікована пряма структурна таблиця [2]. У ній вхідний сигнал представляється кон'юнкцією змінних



$p_g \in P$ . Наприклад, для  $a_1 \in A$  маємо  $p_1 = x_1$  і  $p_2 = x_2$  (табл. 2). При цьому вхідний сигнал  $X_1 = x_1$  (рядок 1, табл. 1) замінюється на  $P_1 = p_1$ , сигнал  $X_2 = \overline{x_1 x_2}$  – на  $P_2 = \overline{p_1 p_2}$ , сигнал  $X_3 = x_1 x_2$  – на  $P_3 = p_1 p_2$ . Аналогічно замінюються всі рядки зі стовпця  $X_h$  табл.1.

Стовпець  $\Phi_h$  містить набір ФЗП, рівних 1 для запису в  $RG$  коду стану переходу. Як відомо [3], якщо біт номер  $r$  ( $r \in \overline{1, R}$ ) дорівнює одиниці в кодї  $K(a_s)$ , то для відповідного переходу маємо  $D_r = 1$ . З урахуванням цих особливостей формується табл. 4, що представляє блок LUTerTY.

ТАБЛИЦЯ 4. Таблиця блоку LUTerTY МПА А1

$a_m$	$K(a_m)$	$a_s$	$K(a_s)$	$P_h$	$\Phi_h$	$Y_h$	$h$
$a_1$	000	$a_2$	001	$p_1$	$D_3$	$y_1 y_2$	1
		$a_3$	010	$\overline{p_1 p_2}$	$D_2$	$y_3$	2
		$a_3$	010	$\overline{p_1 p_2}$	$D_2$	$y_4$	3
$a_2$	001	$a_3$	010	$p_1$	$D_2$	$y_3 y_4$	4
		$a_4$	011	$\overline{p_1 p_2}$	$D_2 D_3$	$y_5$	5
		$a_5$	100	$\overline{p_1 p_2}$	$D_1$	$y_1 y_2$	6
$a_3$	010	$a_4$	011	$p_1$	$D_2 D_3$	$y_6$	7
		$a_6$	101	$p_2$	$D_1 D_3$	$y_3 y_5$	8
$a_4$	011	$a_2$	001	$p_1$	$D_3$	$y_2$	9
		$a_5$	100	$\overline{p_1 p_2}$	$D_1$	$y_1 y_4$	10
		$a_6$	101	$\overline{p_1 p_2}$	$D_1 D_3$	$y_6$	11
$a_5$	100	$a_6$	101	$p_2$	$D_1 D_3$	$y_2 y_6$	12
		$a_1$	000	$\overline{p_2}$	—	—	13
$a_6$	101	$a_7$	110	$p_1$	$D_1 D_2$	$y_1 y_3 y_5$	14
		$a_3$	010	$\overline{p_1 p_2}$	$D_2$	$y_2 y_6$	15
		$a_4$	011	$\overline{p_1 p_2}$	$D_2 D_3$	$y_4$	16
$a_7$	110	$a_1$	000	1	—	$y_3 y_5$	17

Системи (7), (8) включають терми  $F_h = A_m X_h$ , де  $A_m$  – кон'юнкція змінних  $T_r \in T$ , що відповідає коду  $K(a_m)$ . Так, код 000 відповідає коду  $A_1 = \overline{T_1 T_2 T_3}$ , код 001 відповідає  $A_2 = \overline{T_1 T_2} T_3$  і так далі. У формулах для функцій (7), (8) терми можуть склеюватись [4]. Будемо показувати такі терми у квадратних дужках. Наведемо лише дві з 9 функцій:

$$D_2 = [F_2 \vee F_3] \vee F_4 \vee F_5 \vee F_7 \vee [F_{14} \vee F_{15} \vee F_{16}] = \overline{T_1 T_2 T_3} p_1 \vee \overline{T_1 T_2} T_3 p_1 \vee \overline{T_1 T_2} T_3 p_2 \vee \overline{T_1 T_2} T_3 p_1 \vee T_1 T_3;$$

$$y_6 = F_7 \vee F_{11} \vee F_{12} \vee F_{15} = \overline{T_1 T_3} \overline{T_4} p_1 \vee \overline{T_1 T_2} T_3 \overline{p_1 p_2} \vee T_1 \overline{T_2} T_3 p_2 \vee T_1 T_2 T_3 \overline{p_1 p_2}.$$

Для виконання останнього кроку необхідно функції, що реалізуються, представити у вигляді таблиць істинності [4]. Це відбувається під час виконання етапу технологічного відображення

(technology mapping) [11] схеми МПА з використанням ресурсів FPGA. Для цього етапу необхідно використовувати різноманітні САПР. Для мікросхем AMD Xilinx використовуються пакети Vivado [25] та Vitis [26].

Оцінимо апаратні витрати у схемі МПА А1. Очевидно, у схемі використаний один блок ЕМВ. Для функції  $p_2^L = T_1 T_2 x_8$  виконується умова (4), отже, схема блоку LUTerP складається із одного елемента LUT. Блок LUTerOR описується формулою  $p_2 = p_2^E \vee p_2^L$ . Цей вираз є функцією двох аргументів. Тому, схема блоку LUTerOR складається з одного елемента LUT. Як випливає з табл. 4, блок LUTerTY представляється СБФ, що включає дев'ять функцій. Максимальна кількість аргументів у цих функціях не перевищує  $G + R = 5 = S_L$ . Отже, схема блоку LUTerTY складається із 9 елементів LUT.

Цей аналіз показує, що на основі моделі МПА  $S_3$  схема автомата А1 складається з одного ЕМВ і 11 елементів LUT. Схема має три рівні логіки. Зазначимо, що без розділення вхідних змінних на ЕМВ можна було б реалізувати лише одну із змінних  $p_g \in P$ . Це призвело б до збільшення числа LUT у схемі LUTerP та дублювання змінних  $x_i \in X$  у блоках ЕМВ та LUTerP, що ускладнює систему міжз'єднань.

**Висновок.** Блоки вбудованої пам'яті ЕМВ – це ефективний засіб для реалізації систем булевих функцій [13]. Для досить складних автоматів умова (4) порушується практично для всіх функцій систем (1), (2), що представляють схему МПА. Для зменшення числа елементів LUT у схемі та регуляції системи міжз'єднань доцільно замінити вхідні змінні  $x_i \in X$  додатковими змінними [4]. Для реалізації такої заміни ідеально підходять блоки ЕМВ.

Проте блоки ЕМВ широко використовуються для реалізації різних табличних функцій. Тому їх кількість, що є в наявності, може бути меншою за необхідну для реалізації схеми, що замінює входи. У цьому випадку ми пропонуємо використовувати метод поділу вхідних змінних, представлений у цій статті.

Дослідження ефективності запропонованого методу проводилося з використанням стандартних МПА з бібліотеки [24], еволюційної платформи [16] САПР Vivado [25]. Як засвідчили дослідження, наш метод дозволяє реалізувати ЗВЗ всіх автоматів бібліотеки [24]. При цьому кількість елементів LUT у схемі ЗВЗ зменшується від 100 % (якщо для ЗВЗ достатньо одного ЕМВ) до 82 %. Зазначимо, що з 37 % автоматів [24] метод ЗВЗ можна застосувати лише разом із розділенням вхідних змінних.

Запропонований метод універсальний і не залежить від типу МПА. У подальших дослідженнях ми плануємо розробити методи синтезу автоматів Мура та суміщених МПА [27, 28] з використанням запропонованого підходу.

#### Список літератури

1. Skliarova I., Sklyarov V., Sudnitson A. Design of FPGA-based circuits using hierarchical finite state machines. Tallinn: TUT Press, 2012. 240 p. <https://doi.org/10.1109/IranianCEE.2013.6599683>
2. Baranov S. Logic and System Design of Digital Systems. Tallinn: TUT Press, 2008.
3. Baranov S. Finite State Machines and Algorithmic State Machines. Amazon, 01 2018.
4. DeMicheli G. Synthesis and Optimization of Digital Circuits. McGraw–Hill, 1994. 576 p.
5. Wolf W. FPGA-Based System Design. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2004. 548 p.
6. Maxfield C. FPGAs: Instant access. Newnes, 2008. 204 p.
7. Trimberger S. Three ages of FPGA: A retrospective on the first thirty years of FPGA technology. *IEEE Proceedings Solid-State Circuits Magazine*. 2018. Vol. 10, No. 2. P. 16–29. <http://dx.doi.org/10.1109/mssc.2018.2822862>

8. 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>
9. 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](https://doi.org/10.1007/978-3-319-04708-9_6).
10. Barkalov A., Titarenko L. Logic synthesis for FSM-based control units. Berlin: Springer, 2009. 233 p.
11. Barkalov A., Titarenko L., Mielcarek K., Chmielewski S. Logic Synthesis for FPGA-Based Control Units - Structural Decomposition in Logic Design. *Lecture Notes in Electrical Engineering*. Berlin: Springer. 2020. <https://doi.org/10.1007/978-3-030-38295-7>
12. Marwedel P. Embedded System Design: Embedded Systems Foundations of Cyber-Physical Systems, and the Internet of Things. 3rd ed. Springer International Publishing, 2018. <https://doi.org/10.1007/978-3-030-60910-8>
13. Garcia-Vargas I., Senhadji-Navarro R., Jim Bnez-Moreno G., Civit-Balcells A., Guerra-Gutierrez P. ROM-based finite state machines implementation in low cost FPGAs. *IEEE Intern. Simp. on Industrial Electronics (ISIE'07)* (Vigo, 2007). 2007. P. 2342–2347. <https://doi.org/10.1109/ISIE.2007.4374972>
14. Garcia-Vargas L., Senhadji-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. <https://api.semanticscholar.org/CorpusID:14095082>
15. Xilinx. <https://www.xilinx.com/products/silicon-devices.html> (звернення: 01.01.2023)
16. Xilinx. Virtex-7 Family Overview. PDF, Xilinx Corporation. 2021. [http://www.xilinx.com/support/documentation/data\\_sheets/ds183\\_Virtex\\_7\\_Data\\_Sheet.pdf](http://www.xilinx.com/support/documentation/data_sheets/ds183_Virtex_7_Data_Sheet.pdf) (звернення: 01.01.2023)
17. Chapman K. Multiplexer Design Techniques for Datapath Performance with Minimized Routing Resources. Xilinx All Programmable, 2014. <https://api.semanticscholar.org/CorpusID:61416418> (звернення: 01.01.2023)
18. Feng W., Greene J., Mishchenko A. Improving FPGA Performance with a S44 LUT Structure. In Proceedings of the 2018 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays (New York: NY, USA, 02 2018), FPGA'18, Association for Computing Machinery. 2018. P. 61–66. <http://dx.doi.org/10.1145/3174243.3174272>
19. Islam M.M., Hossain M.S., Shahjalal M., Hasan M.K., Jang Y.M. Area-time efficient hardware implementation of modular multiplication for elliptic curve cryptography. *IEEE Access*. 2020. Vol. 8. P. 73898–73906. <http://dx.doi.org/10.1109/ACCESS.2020.2988379>
20. Machado L., Cortadella J. Support-Reducing Decomposition for FPGA Mapping. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*. Vol. 39, No. 1. P. 213–224. <http://dx.doi.org/10.1109/TCAD.2018.2878187>
21. Czerwinski R., Kania D. Finite state machines logic synthesis for complex programmable logic devices. *Bulletin of the polish academy of sciences technical sciences*. 2010. Vol. 58, No. 4. 172 p. <http://dx.doi.org/10.2478/v10175-010-0067-6>
22. Kolopienczyk M., Titarenko L., Barkalov A. Design of EMB-based Moore FSMs. *Journal of Circuits, Systems and Computers*. 2017. Vol. 26, No. 7. P. 1–23. <https://api.semanticscholar.org/CorpusID:22752093>
23. Tiwari A., Tomko K.A. 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. <https://doi.org/10.1109/DATE.2004.1269007>
24. Yang S. Logic synthesis and optimization benchmarks user guide. Version 3.0. Techn. Rep. Microelectronics Center of North Carolina, 1991. 43 p.
25. Vivado. <https://www.xilinx.com/products/design-tools/vivado.html> (звернення: 01.01.2023)
26. Xilinx. Vitis Platform. <https://www.xilinx.com/products/design-tools/vitis/vitis-platform.html> (звернення: 01.01.2023)
27. 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>
28. 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 30.09.2023

**Баркалов Олександр Олександрович,**  
 доктор технічних наук, професор  
 Університету Зеленогурського (Польща),  
<https://orcid.org/0000-0002-4941-3979>  
[A.Barkalov@iee.uz.zgora.pl](mailto:A.Barkalov@iee.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.Titareno@ie.uz.zgora.pl](mailto:L.Titareno@ie.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](mailto: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](mailto:avmatv@ukr.net)

**Сабурова Світлана Олександрівна,**  
 кандидат технічних наук, доцент  
 Харківського національного університету радіоелектроніки.  
<https://orcid.org/0000-0001-6286-1648>  
[sabsvet@gmail.com](mailto:sabsvet@gmail.com)

УДК 004.274

**О.О. Баркалов<sup>1</sup>, Л.О. Тітаренко<sup>1,2</sup>, О.М. Головін<sup>3</sup>, О.В. Матвієнко<sup>3</sup>, С.О. Сабурова<sup>2</sup>**

## **Розділення вхідних змінних для оптимізації схеми автомата Мілі**

<sup>1</sup> Університет Зеленогурський, Зелена Гура, Польща

<sup>2</sup> Харківський національний університет радіоелектроніки, Харків, Україна

<sup>3</sup> Інститут кібернетики імені В.М. Глушкова НАН України, Київ

\* Листування: [avmatv@ukr.net](mailto:avmatv@ukr.net)

**Вступ.** Одним із найважливіших блоків практично будь-якої цифрової системи є пристрій управління (ПУ), оскільки його характеристики значною мірою визначають характеристики системи загалом.

У практиці інженерного проектування поведінка ПУ часто визначається за допомогою моделі мікропрограмного автомата (МПА) Мілі. Особливість МПА Мілі це залежність систем булевих функцій, що задають схему, від входів та станів. Ця особливість враховується при оптимізації характеристик схеми в базисі мікросхем FPGA.

Під час розробки схеми автомата необхідно оптимізувати його характеристики, такі як швидкодія та апаратні витрати.

FPGA – один із найпоширеніших базисів при побудові цифрових систем. Для реалізації схеми ПУ достатньо таких компонентів FPGA, як логічні елементи табличного типу (LUT), програмовані тригери, вбудовані блоки пам'яті (ЕМВ) і програмовані між'єднання.

**Мета роботи.** У цій роботі пропонується метод зменшення апаратних витрат у схемі МПА Мілі, що реалізується у базисі FPGA. Розглянуто завдання реалізації схеми у змішаному елементному базисі. Під змішаним базисом розуміється спільне використання елементів табличного типу LUT та блоків пам'яті ЕМВ. Розглядається ситуація, коли кількість доступних блоків ЕМВ вкрай обмежена. Це цілком можливо, оскільки блоки ЕМВ широко використовуються реалізації різних операційних блоків цифрових систем.

Основний недолік елементів LUT це мала кількість входів. Сучасні цифрові системи можуть генерувати сигнали логічних умов, які поступають до ПУ, і кількість яких у десятки разів перевищує число входів LUT. Така невідповідність між характеристиками алгоритму керування та кількістю входів еле-

ментів LUT призводить до багаторівневих схем ПУ з нерегулярною структурою міжз'єднань. Для оптимізації багаторівневих схем може бути застосованим метод заміни вхідних змінних та спільне використання елементів LUT та блоків ЕМВ.

**Результати.** Аналіз ефективності запропонованого методу проводився за допомогою бібліотек стандартних МПА платформи САПР Vivado. Як показали дослідження, запропонований метод дозволяє зменшити кількість елементів LUT у діапазоні від 100 % до 82 %. Для 37 % автоматів метод заміни вхідних змінних можна застосувати лише разом із розділенням вхідних змінних.

**Висновки.** Запропонований метод дозволяє зменшити апаратні витрати (кількість елементів LUT та їх міжз'єднань), час затримки та споживану потужність. У статті наведено умови застосування запропонованого методу. Наведено результати досліджень ефективності запропонованого методу для стандартних автоматів із використанням мікросхем сімейства Virtex-7 та промислового пакету Vivado.

**Ключові слова:** мікропрограмний автомат, синтез, FPGA, ЕМВ, LUT, заміна входів.

UDC 004.274

Alexandr Barkalov<sup>1</sup>, Larysa Titarenko<sup>1,2</sup>, Oleksandr Golovin<sup>3</sup>, Oleksandr Matvienko<sup>3</sup>, Svitlana Saburova<sup>2</sup>

## Separation of Input Variables for Optimization of the Mealy FSM Circuit

<sup>1</sup> *University of Zielona Gora, Poland*

<sup>2</sup> *Kharkiv National University of Radio Electronics, Kharkiv, Ukraine*

<sup>3</sup> *V.M. Glushkov Institute of Cybernetics of the NAS of Ukraine, Kyiv*

\* *Correspondence: [avmatv@ukr.net](mailto:avmatv@ukr.net)*

**Introduction.** One of the most important blocks of almost any digital system is the control device (CU), since the characteristics of the CU largely determine the characteristics of the system as a whole.

In the practice of engineering design, the behavior of the CU is often specified using the Mealy finite state machine (FSM) model. A feature of Mealy FSM is the dependence of systems of Boolean functions that define the circuit on FSM inputs and states. In this article, this feature is taken into account when optimizing the characteristics of FSM circuit in the basis of FPGA chips.

When developing FSM circuits, it is necessary to optimize its characteristics, such as performance and hardware costs.

FPGAs are one of the most common logic bases for implementing digital systems. FPGA components such as look-up table (LUT) elements, programmable flip-flops, embedded memory blocks (EMBs), and programmable interconnects are sufficient to implement the CU circuit.

**The purpose of the article.** In this paper, we propose a method for reducing hardware costs in the Mealy FSM circuit implemented in the FPGA basis. In this case, the problem of implementing a circuit in a mixed elemental basis is considered. A mixed basis is understood as the joint use of LUTs and EMBs. The situation is considered when the number of available EMBs is extremely limited, which is quite possible, since EMBs are widely used to implement various operating blocks of digital systems.

The main disadvantage of LUTs is the small number of inputs. Modern digital systems can generate signals of logical conditions entering the CU, the number of which is tens of times greater than the number of LUT inputs. This discrepancy between the characteristics of the control algorithm and the number of inputs of the LUTs leads to multilevel CU circuits with an irregular structure of programmable interconnections. To optimize multilevel schemes, the method of replacing input variables is used with the joint use of LUTs and EMB blocks.

**Results.** The analysis of the effectiveness of the proposed method was carried out using the libraries of standard benchmarks FSMs and the Vivado CAD platform. Studies have shown that the proposed method makes it possible to reduce the number of LUTs in the range from 100% to 82%. For 37% of automata, the method of replacing input variables can be applied only in conjunction with the separation of input variables.

**Conclusions.** The proposed method makes it possible to reduce hardware costs (the number of LUTs and their interconnections), delay time, and power consumption. The article shows the conditions for applying the proposed method. The results of studies of the effectiveness of the proposed method for standard automata using chips of the Virtex-7 family and the Vivado industrial package are presented.

**Keywords:** finite state machine, synthesis, FPGA, EMB, LUT, input replacement.