

інформацію, яка буде використана. Додаткові поля інших мікрокоманд не використовуються. З цього виходить, що одному ОЛЛ відповідає тільки одне використовуване значення з КП. В роботі пропонується для реалізації зазначеної відповідності використовувати окремий перетворювач кодів, який задається функцією

$$K(B_i) = f(\tau). \tag{10}$$

Апаратні витрати на реалізацію КМПК пропонується додатково зменшити шляхом застосування методики елементаризації ОЛЛ [16]. Елементаризація полягає в розбитті ОЛЛ з декількома входами на такі, що мають лише один вхід. Використання таких ОЛЛ при умовних або безумовних переходах дозволяє формувати тільки код ОЛЛ. Код компоненти ОЛЛ при цьому завжди еквівалентний десятковому нулю.

Функцію блоків вбудованої пам'яті можуть виконувати LUT елементи, які можна вважати блоками пам'яті з S_L входами та одним виходом. З цього виходить, що один LUT елемент еквівалентний 2^{S_L} коміркам пам'яті.

Якщо для ГСА Г виконується відношення $S_L \geq R$,

$$S_L \geq R, \tag{11}$$

то для її інтерпретації пропонується використовувати модель КМПК U_2 з перетворювачем кодів ОЛЛ на коди класів псевдоеквівалентних елементарних операторних лінійних ланцюгів (ПЕОЛЛ) (рис. 2).

В КМПК U_2 схема адресації мікрокоманд реалізується у вигляді сукупності блоків PLA. Схема КП та перетворювач кодів реалізуються з сукупності елементів LUT. Блоки СТ та RG також реалізовані на LUT елементах. Для кодування класів $B_i \in \Pi_C$ використовуються змінні z_r з множини $Z = \{z_1, \dots, z_{R_i}\}$.

Схема адресації мікрокоманд реалізує функції

$$\Psi = \Psi(Z, X). \tag{12}$$

Решта функцій реалізована за допомогою LUT елементів:

$$Y = Y(\tau, T); \tag{13}$$

$$y_E = y_E(\tau, T); \tag{14}$$

$$y_0 = y_0(\tau, T). \tag{15}$$

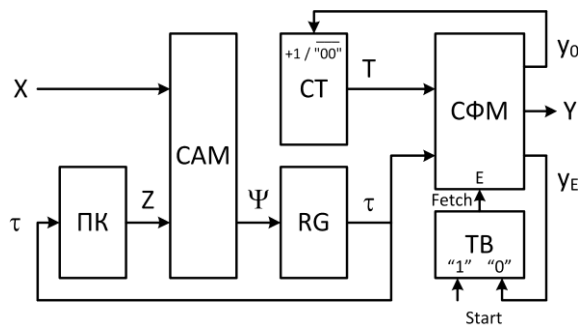


Рисунок 2 – Структурна схема КМПК U_2

Методика синтезу КМПК U_2 передбачає наступні етапи:

1. Формування множини C для ГСА Г.
2. Застосування методики елементаризації.
3. Формування множин C_1 та Π_C .
4. Кодування ОЛЛ, їх компонент та класів $B_i \in \Pi_C$.
5. Формування систем функцій (13)-(15) для СФМ.
6. Формування системи функцій (12) для САМ.
7. Формування системи функцій (10) для ПК.
8. Синтез логічної схеми КМПК.

Приклад застосування запропонованої методики

Для демонстрації запропонованої методики виконаємо синтез композиційного мікропрограмного пристрою керування з використанням кодів ПЕОЛЛ для формування адрес переходів на прикладі тестової ГСА Г₁, яка наведена на рис. 3. Зазначена ГСА Г₁ містить $|B_1| = 17$ операторних вершин, $|B_2| = 3$ умовних вершини, що визначає її як лінійну.

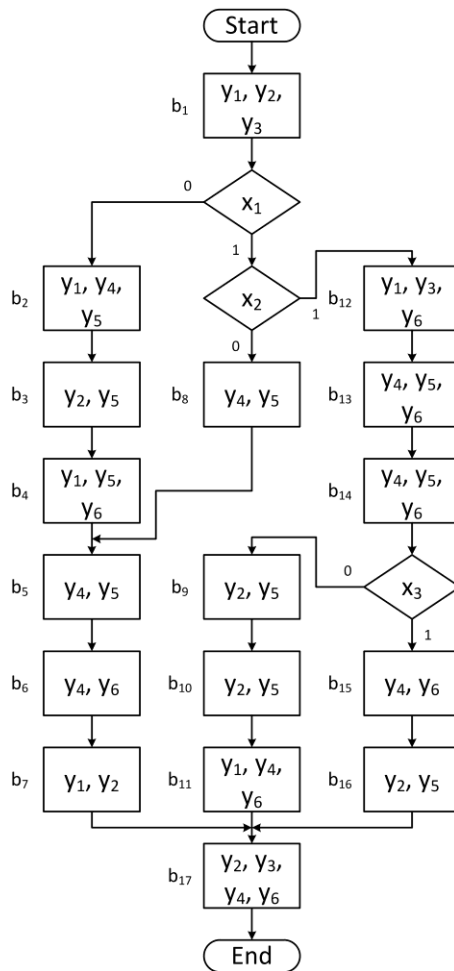


Рисунок 3 – Тестова ГСА Г

1. Формування множини C для ГСА G . За результатом роботи алгоритму формування ОЛЛ, наведеного в роботі [6], отримаємо множину з $G = 6$ ОЛЛ: $C = \{\alpha_1, \dots, \alpha_6\}$, де $\alpha_1 = \langle b_1 \rangle$, $\alpha_2 = \langle b_2, b_3, b_4 \rangle$, $\alpha_3 = \langle b_8, b_5, b_6, b_7 \rangle$, $\alpha_4 = \langle b_9, b_{10}, b_{11} \rangle$, $\alpha_5 = \langle b_{12}, b_{13}, b_{14} \rangle$, $\alpha_6 = \langle b_{15}, b_{16}, b_{17} \rangle$.

2. Застосування методики елементаризації призведе до збільшення кількості ОЛЛ до $G' = 8$ через розбиття α_3 на $\alpha_3^1 = \langle b_8 \rangle$ та $\alpha_3^2 = \langle b_5, b_6, b_7 \rangle$, а також α_6 на $\alpha_6^1 = \langle b_{15}, b_{16} \rangle$ та $\alpha_6^2 = \langle b_{17} \rangle$. Згідно з визначенням, елементарним є такий ОЛЛ, який має тільки один вхід. Якщо ОЛЛ α_i має P входів, відбувається його розбиття на ОЛЛ $\alpha_i^1, \alpha_i^2, \dots, \alpha_i^P$, при цьому кожен з P входів попереднього ОЛЛ стає входом відповідного нового елементарного ОЛЛ (ЕОЛЛ).

3. Формування множин C_1 та Π_C . ОЛЛ $\alpha_6^2 \notin C_1$ через те, що $e(b_{17}, b_E) \in E$. В множині C_1 за ознакою псевдоеквівалентності сформовані $I = 4$ класи ПОЛЛ: $\Pi_C = \{B_1, B_2, B_3, B_4\}$, де $B_1 = \{\alpha_1\}$, $B_2 = \{\alpha_2, \alpha_3^1\}$, $B_3 = \{\alpha_5\}$, $B_4 = \{\alpha_3^2, \alpha_4, \alpha_6^1\}$. Нагадаємо, що ОЛЛ $\{\alpha_i, \alpha_j\} \subseteq C$ називаються псевдоеквівалентними, якщо їх виходи зв'язані зі входом однієї вершини ГСА G .

4. Кодування ОЛЛ, їх компонент та класів $B_i \in \Pi_C$. Відповідно до (2) для кодування $G' = 8$ ЕОЛЛ достатньо $R_1 = 3$ змінних з множини $\tau = \{\tau_1, \tau_2, \tau_3\}$, де G' – кількість ЕОЛЛ. Максимальна кількість компонент $F_{\max} = 3$, для їх кодування відповідно до (3) необхідно $R_2 = 2$ змінних з множини $T = \{T_1, T_2\}$. У загальному випадку для кодування $M = 17$ операторних вершин відповідно до (1) достатньо $R = 5$ двійкових розрядів, отже умова (5) виконується, та застосування

методики розділення кодів доцільно. При цьому для кодування $I = 4$ класів ПЕОЛЛ відповідно до (9) необхідно $R_I = 2$ змінні, які формують множину $Z = \{z_1, z_2\}$.

Виконаємо кодування ЕОЛЛ $\alpha_g \in C$ та їх класів у довільному порядку: $K(\alpha_1) = 000, \dots, K(\alpha_6^2) = 111$; $K(B_1) = 00, \dots, K(B_4) = 11$. Для виконання умови (4) призначимо першій компоненті кожного ОЛЛ $\alpha_g \in C$ код 00, другій – 01, третій – 10. Це дозволить визначити адреси $A(b_q)$ мікрокоманд КМПК $U_2(\Gamma_1)$, наведені в табл. 1. Запис $U_i(\Gamma_j)$ означає КМПК U_i , що інтерпретує ГСА Γ_j .

Адреса мікрокоманди визначається як конкатенація коду ЕОЛЛ та коду відповідної компоненти, наприклад, з табл. 1 маємо $A(b_4) = 001 * 10 = 00110$, $A(b_{15}) = 110 * 00 = 11000$ та ін., де «*» - операція конкатенації.

5. Формування систем функцій (13)-(15) для СФМ. Формат мікрокоманди КМПК U_2 містить поля y_0, y_E, FY , де поле FY містить інформацію про мікрооперації, що підлягають формуванню (принципи та способи кодування зазначеної інформації в даній статті не розглядаються).

Вміст СФМ КМПК $U_2(\Gamma_1)$ наведений у табл. 2. Принцип формування його вмісту тривіальний. До мікрокоманди з адресою $A(b_q)$ заноситься інформація про мікрооперації, які підлягають формуванню. Якщо вершина $b_q \in B_1$ не є виходом ЕОЛЛ $\alpha_g \in C$, то до поля y_0 заноситься «1», у протилежному випадку – «0». Якщо $(\exists e_i \in E): e_i = (b_q, b_E)$, то до поля y_E заноситься «1», інакше – «0».

Таблиця 1- Таблиця для формування адрес мікрокоманд КМПК $U_2(\Gamma_1)$

α_i $\tau_1\tau_2\tau_3$	α_1 000	α_2 001	α_3^1 010	α_3^2 011	α_4 100	α_5 101	α_6^1 110	α_6^2 111
T_1T_2								
00	b_1	b_2	b_8	b_5	b_9	b_{12}	b_{15}	b_{17}
01	*	b_3	*	b_6	b_{10}	b_{13}	b_{16}	*
10	*	b_4	*	b_7	b_{11}	b_{14}	*	*

Таблиця 2 - Умовне відображення вмісту СФМ КМПК $U_2(\Gamma_1)$

α_i $\tau_1\tau_2\tau_3$	α_1 000	α_2 001	α_3^1 010	α_3^2 011	α_4 100	α_5 101	α_6^1 110	α_6^2 111
T_1T_2								
00	$Y(b_1)$	$y_0, Y(b_2)$	$Y(b_8)$	$y_0, Y(b_5)$	$y_0, Y(b_9)$	$y_0, Y(b_{12})$	$y_0, Y(b_{15})$	$y_E, Y(b_{17})$
01	*	$y_0, Y(b_3)$	*	$y_0, Y(b_6)$	$y_0, Y(b_{10})$	$y_0, Y(b_{13})$	$Y(b_{16})$	*
10	*	$Y(b_4)$	*	$Y(b_7)$	$Y(b_{11})$	$Y(b_{14})$	*	*

Таблиця 3 - Таблиця для формування системи функцій САМ КМПК $U_2(\Gamma_1)$

B_i	$K(B_i)$		b_q	$A(\alpha_g), (b_q \in \alpha_g)$			X_h	Ψ_h	h
	z_1	z_2		τ_1	τ_2	τ_3			
B_1	0	0	b_2	0	0	1	\bar{x}_1	D_3	1
			b_8	0	1	0	$x_1\bar{x}_2$	D_2	2
			b_{12}	1	0	1	x_1x_2	D_1D_3	3
B_2	0	1	b_5	0	1	1	–	D_2D_3	4
B_3	1	0	b_9	1	0	0	\bar{x}_3	D_1	5
			b_{15}	1	1	0	x_3	D_1D_2	6
B_4	1	1	b_{17}	1	1	1	–	$D_1D_2D_3$	7

При виконання умови (10) кожна функція з систем (13)–(15) реалізується на одному елементі LUT. Таке рішення є оптимальним. При цьому таблиця для формування вмісту СФМ розглядається як таблиця істинності функцій (13)–(15).

6. Формування системи функцій (12) для САМ. Для формування систем САМ визначимо систему узагальнених формул переходів з виходів ЕОЛЛ $\alpha_g \in C_1$:

$$\begin{aligned} B_1 &\rightarrow \bar{x}_1b_2 \vee x_1\bar{x}_2b_8 \vee x_1x_2b_{12}; \\ B_2 &\rightarrow b_5; \quad B_3 \rightarrow \bar{x}_3b_9 \vee x_3b_{15}; \\ B_4 &\rightarrow b_{17}. \end{aligned} \quad (16)$$

На базі системи (16) побудуємо таблицю для формування системи функцій САМ (табл. 3). Адреси ЕОЛЛ визначаються за табл. 1. Зазначимо, що $\Psi = \{D_1, D_2, D_3\}$. Загальна кількість строк $H_2(\Gamma_j)$ в таблиці блоку КМПК $U_2(\Gamma_j)$ співпадає з кількістю термів у системи узагальнених формул переходів. В нашому прикладі, $H_2(\Gamma_1) = 7$. Зазначимо, що $H_1(\Gamma_1) = 10$, де $H_i(\Gamma_j)$ визначає кількість строк в таблиці схеми адресації мікрокоманд КМПК U_i , що інтерпретує ГСА Γ_j .

Для синтезу схеми адресації мікрокоманд необхідно визначити систему функцій (12). В системі (11) немає потреби, оскільки застосування методики елементаризації ОЛЛ призвело до того, що код компоненти-входу ОЛЛ дорівнює десятковому нулю.

Система (12) формується за таблицею переходів. Так, за табл. 3 після деяких спрощень маємо:

$$\begin{aligned} D_1 &= \bar{z}_2x_1x_2 \vee z_1; \\ D_2 &= \bar{z}_1\bar{z}_2x_1\bar{x}_2 \vee \bar{z}_1z_2 \vee z_1\bar{z}_2x_3 \vee z_1z_2; \\ D_3 &= \bar{z}_1\bar{x}_1 \vee \bar{z}_1x_2 \vee z_2. \end{aligned} \quad (17)$$

При виконанні умов

$$S \geq L + R_l; \quad (18)$$

$$t \geq R_l; \quad (19)$$

$$q \geq H_2(\Gamma_j), \quad (20)$$

САМ тривіальним чином може бути реалізована за допомогою одного макроосередка PLA. Якщо ці співвідношення порушуються, то необхідно декілька макроосередків. Для зменшення кількості макроосередків PLA в САМ можна застосувати відомі методики [17].

7. Формування системи функцій (10) для ПК виконується на базі табл. 4, яка містить співвідношення між ОЛЛ та класами, до яких вони входять.

Таблиця 4. Таблиця для формування вмісту ПК

α_g	τ_1	τ_2	τ_3	B_j	z_1	z_2
α_1	0	0	0	B_1	0	0
α_2	0	0	1	B_2	0	1
α_3^1	0	1	0	B_2	0	1
α_3^2	0	1	1	B_4	1	1
α_4	1	0	0	B_4	1	1
α_5	1	0	1	B_3	1	0
α_6^1	1	1	0	B_4	1	1
α_6^2	1	1	1	–	*	*

З табл. 4 маємо систему функцій для ПК:

$$\begin{aligned} z_1 &= \tau_1 \vee \tau_2\tau_3; \\ z_2 &= \tau_1\bar{\tau}_3 \vee \bar{\tau}_1\tau_3 \vee \tau_2. \end{aligned} \quad (21)$$

Якщо значення R_l не перевищує кількість входів LUT-елементу, для реалізації перетворювача кодів знадобиться R_l додаткових LUT-елементів, кожен з яких сформує окремий розряд коду класу ПЕОЛЛ. В нашому випадку достатньо $R_l = 2$ таких елементи.

8. Синтез логічної схеми КМПК у базисі гібридних FPGA виконується за допомогою будь-якої САПР цифрових пристроїв (Xilinx ISE, Xilinx Vivado, Altera Quartus та ін.) та в даній статті детально не розглядається.

Висновки

Запропонована методика адресації мікрокоманд з використанням кодів класів псевдоеквівалентних елементарних операторних лінійних ланцюгів орієнтований на зменшення кількості макроосередків PLA в схемі адресації мікрокоманд. При цьому кількість тактів інтерпретації алгоритму керування співпадає з відповідним значенням для базової структури КМПК U_1 з розділенням кодів.

Зменшення кількості термів в ДНФ функції збудження пам'яті може привести до зменшення кількості рівнів в комбінаційній частині пристрою керування. Це в свою чергу призводить збільшення швидкодії обчислювальної системи в цілому. Розглянутий приклад показав, що кількість макроосередків PLA в залежності від їх параметрів зменшується на значення до 30 % у

порівнянні з КМПК U_1 . Нагадаємо, що застосування запропонованої методики доцільно лише для лінійних ГСА за умови виконання відношення (5).

Наукова новизна запропонованої методики полягає у використанні кодів класів псевдоеквівалентних елементарних ОЛЛІ для зменшення кількості макроосередків PLA в схемі адресації мікрокоманд. Практична значимість методики полягає у зниженні вартості схеми та у деяких випадках збільшенню швидкодії у порівнянні з аналогами.

Зазначимо, що при порушенні відношень (18)-(20) використання запропонованої методики неможливе, а структура КМПК та відповідний метод синтезу вимагають модифікації, на що й спрямовані подальші дослідження.

Список використаної літератури

1. Barkalov A., Titarenko L. Logic Synthesis for Compositional Microprogram Control Unit – Berlin: Springer, 2008. – 272 pp.
2. Synthesis of compositional microprogram control unit with dedicated area of inputs / Alexander Barkalov, Larisa Titarenko, Jacek Bieganski, A.N. Miroshkin // W: Design of digital systems and devices / eds M. Adamski, A. Barkalov, M. Wegrzyn. – Berlin: Springer-Verlag, 2011. (Lecture Notes in Electrical Engineering; 79) – pp. 193–214.
3. Hardware reduction in FPGA-based compositional microprogram control units / Alexander Barkalov, Larisa Titarenko, A. N. Miroshkin // W: Proceedings of IEEE East-West Design & Test Symposium - EWDTS '09. Moscow, Russia, 2009. - Moscow: The Institute of Electrical and Electronics Engineers, Inc., 2009, 21-26 pp.
4. Реализация композиционных микропрограммных устройств управления на FPGA-микросхемах / А.А. Баркалов, Л.А. Титаренко, А.Н. Мирошкин // Радиоэлектроника и информатика. – 2011, январь-март. – № 1(52). – С. 54-58.
5. Модификация метода синтеза композиционного устройства управления для реализации в базисе FPGA / А.А. Баркалов, А.А. Красичков, А.Н. Мирошкин // Научные труды Донецкого национального технического университета. Серия "Проблемы моделирования и автоматизации проектирования динамических систем" (МАП-2010). – 2010. – Вып. 8 (168). – С. 57-67.
6. Мірошкін О.М. Методи синтезу композиційних мікропрограмних пристроїв керування з модифікацією системи адресації мікрокоманд : дис. на здобуття наукового ступеня канд. техн. наук : спец 05.13.05 «Комп'ютерні системи і компоненти» / Мірошкін Олександр Миколайович ; ДВНЗ «Донецький національний технічний університет». – Донецьк, 2013. – 155 с.
7. Синтез устройства управления с разделением кодов и модификацией операторных линейных цепей / А.А. Баркалов, А.А. Красичков, А.Н. Мирошкин // Наукові праці Донецького національного технічного університету. Серія "Інформатика, кібернетика і обчислювальна техніка" (КОТ-2008). – 2008. – Вип. 9 (132). – С. 183-187.
8. Расширение формата микрокоманд в устройстве управления с разделением кодов / А.А. Баркалов, С.А. Ковалёв, И.Я. Зеленёва и др. // Известия ТТИ ЮФУ-ДонНТУ. Материалы Девятого Международного семинара "Практика и перспективы развития партнерства в сфере высшей школы": в 3-х кн. – Таганрог: Изд-во ТТИ ЮФУ, 2008. – Кн. 3. – № 8. – С. 4-12.
9. Баркалов А.А. Синтез устройств управления на программируемых логических устройствах / А.А. Баркалов. – Донецк: ДНТУ, 2002. – 262 с.
10. Соловьев В.В. Проектирование цифровых схем на основе программируемых логических интегральных схем / В.В. Соловьев. – М.: Горячая линия-ТЕЛЕКОМ, 2001. – 636 с.
11. Баркалов А.А. Синтез композиционных микропрограммных устройств управления / А.А. Баркалов, Л.А. Титаренко. – Х.: Коллегиум, 2007. – 302 с.
12. Kabiani A., Brown S. The Hybrid Field Programmable Architecture. – IEEE Design & Test of Computers. – 1999, V.16, №4. – pp. 74–83.

13. Singh, S.K., Singh, R.K. Design flow of reconfigurable embedded system architecture using LUTs/PLAs. 2nd IEEE International Conference on Parallel Distributed and Grid Computing (PDGC), 6-8 Dec. 2012, pp. 385–390. ISBN 978-1-4673-2922-4.
14. Altera Corporation APEX20K PLD Family Data Sheet (2004) [электронный ресурс]. – Режим доступа: www.altera.com/literature/ds/apex.pdf
15. Баркалов А.А. Расширение формата микрокоманд в микропрограммном устройстве управления с разделением кодов / А.А. Баркалов, Л.А. Титаренко, А.Н. Мирошкин // Радиоэлектроника и информатика. – 2008, апрель-июнь. – № 2 (41). – С. 40-45.
16. Баркалов А.А. Расширение формата микрокоманд в композиционном микропрограммном устройстве управления с элементаризацией операторных линейных цепей / А.А. Баркалов, А.А. Красичков, А.Н. Мирошкин // Радиоелектрон. і комп'ют. системи. – 2010. – № 7. – С. 301–305.
17. Баранов С.И. Цифровые устройства на проектируемых БИС с матричной структурой / С.И. Баранов, В.А. Складов. – М.: Радио и связь, 1986. – 272 с.

Надійшла до редакції 17.03.2015

А.Н. МИРОШКИН

Донецкий национальный технический университет, г. Красноармейск

АДРЕСАЦИЯ МИКРОКОМАНД НА БАЗЕ ПСЕВДОЭКВИВАЛЕНТНЫХ ЭЛЕМЕНТАРНЫХ ОПЕРАТОРНЫХ ЛИНЕЙНЫХ ЦЕПЕЙ В УСТРОЙСТВАХ УПРАВЛЕНИЯ В БАЗИСЕ ГИБРИДНЫХ FPGA

В работе предлагается методика синтеза композиционных микропрограммных устройств управления на гибридных FPGA. Методика направлена на уменьшение аппаратных затрат в семе адресации микрокоманд за счет использования кодов классов псевдоэквивалентных элементарных операторных линейных цепей и учитывает особенности строения гибридного базиса. Методика позволяет уменьшить количество необходимых для реализации устройства управления ресурсов и снизить его стоимость.

Ключевые слова: композиционное микропрограммное устройство управления, операторная линейная цепь, элементаризация, гибридный базис, FPGA, снижение аппаратных затрат.

O. MIROSHKIN

Donetsk National Technical University, Krasnoarmiisk

MICROINSTRUCTION ADDRESSING BASED ON PSEUDOEQUIVALENT ELEMENTARY OPERATIONAL LINEAR CHAINS IN CONTROL UNITS IN HYBRID FPGA

In this paper synthesis method for compositional microprogram control unit is proposed. The method is oriented on control unit implementation in hybrid FPGA and aimed at hardware amount reducing in control unit circuit. Operating of compositional microprogram control unit is based on operational linear chains in flow-chart of control algorithm. Unconditional transitions between microinstructions mostly are provided with nature addressing and next address is generated on a counter. Addresses for conditional and conflicted unconditional transitions are provided with special subcircuit, based on PLA parts of hybrid FPGA. Microoperations to be generated are stored in control memory, which is implemented on LUT-elements as on basic simple memory blocks with several address inputs and one output.

Decreasing of hardware amount can be reached due to number of input or outputs reducing in circuit of microinstruction addressing. Number of inputs can be decreased with the help of pseudoequivalent operational linear chains class codes, and number of outputs can be decreased with elementarization approach implementation.

An example of proposed synthesis methodic is shown. According to the example hardware amount decreasing up to 30 % has been reached.

Proposed method can be used only if given in the paper relations between algorithm parameters are satisfied. Algorithm is represented as graph-scheme and last is the linear one.

Future research is aimed at new synthesis methods development for algorithms with parameters, which do not satisfy given in the paper equations.

Keywords: compositional microprogram control unit, operational linear chain, elementarization, hybrid basis, FPGA, hardware amount reducing.