

EXTENSION OF MICROINSTRUCTION FORMAT IN COMPOSITIONAL MICROPROGRAM CONTROL UNIT WITH ELEMENTARIZATION OF OPERATIONAL LINEAR CHAINS

The modification of synthesis method of compositional microprogram control units is directed to decrease of hardware amount in scheme of control unit in FPGA basis. Reduction of complexity of block of microinstruction addressing is reached due to of field with pseudoequivalent operational linear chain class code. Conditions of proposed method usage possibility are given. The example of method implementation is shown.

Key words: compositional microprogram control unit, operational linear chain, microinstruction addressing, class of pseudoequivalence.

INTRODUCTION

Modern industrial production requires cheap, secure and productive devices as results of design process. That's why reduces of the complexity of developing systems causes the topicality of hardware reduction problem in digital circuits [1]. It is necessary to take into account peculiarities of device structural scheme and element basis features. Some of the peculiarities, that can be used, are pseudoequivalent states and linear type of control algorithm [2].

Compositional microprogram control unit (CMCU) is reasonable to use in case of linear (percentage of operational vertices is over 75 %) algorithms realization [2]. FPGA (Field-Programmable Gate Arrays) basis is widely used nowadays for realization of control unit circuit [3, 4]. Problem of hardware amount minimization is solved by decrease in complexity of device main structural elements by means of decrease in main interconnections widths [5]. One of ways of this problem solving is proposed in the article: control unit realization as compositional microprogram one with code sharing and elementarization of operational linear chains (OLC).

The main purpose of investigation is simplification of combinational part of CMCU via implementation in the microinstruction format of additional field containing pseudoequivalent operational linear chain (POLC) class code. The main task of investigation is development of CMCU synthesis method modification that let decrease number of LUT-elements in Block of Microinstruction Addressing (BMA). Control algorithms are represented as graph-schemes (GSA).

MAIN STATEMENTS

Graph-scheme of control algorithm consists of operational and conditional vertices, making sets E_1 and E_2 accordingly, and the set of arcs E . Let us begin vertex be marked as b_0 , end – b_E . Operational vertex $b_q \in E_1$ contain set of microinstructions $Y(b_q) \subseteq Y$, where $Y = \{y_1, \dots, y_N\}$ is the set of output signals of control unit. Conditional vertex $b_g \in E_2$ contains one elements $X(b_g)$ of the logical conditions set $X = \{x_1, \dots, x_L\}$. In case of operational vertices percentage is over 75 % from total number of vertices, we talk about linear GSA.

OLC is a sequence of operational vertices of graph-scheme of algorithm. Each OLC α_g has accidental number of inputs I_g^i and only one output Q_g . Formal definitions of OLC, its input and output one can find in [4]. OLC with only one input and one output is called elementary [2].

OLC, outputs of which are connected with the input of the same vertex are called pseudoequivalent operational linear chains (POLC). Such OLCs make the class B_i . All classes are packed into the set $B = \{B_1, \dots, B_I\}$ of POLC classes.

Let GSA contains G elementary OLC α_g that form the set C .

$$R_1 = \lceil \log_2 G \rceil \quad (1)$$

bits are enough for encoding elements of the set C . Number of components in OLC α_g is marked as F_g . Maximum length $Q = \max(F_1, \dots, F_G)$ of linear chain

determines number of bits R_2 in the code for encoding OLC components, where

$$R_2 = \lceil \log_2 Q \rceil. \quad (2)$$

Elements $\tau_r \in \tau$ and $T_r \in T$ are used for encoding elementary OLC and their components accordingly. It being known that $|\tau| = R_1$ and $|T| = R_2$. Encoding of components is performed in natural order, that is

$$K(b_{gi}) = K(b_{gi-1}) + 1, \quad (3)$$

where $g = 1, \dots, G, i = 1, \dots, F_g$.

Each operational vertex $b_q \in E_1$ corresponds to microinstruction MI_q storing in control memory (CM) in the cell with address $A(b_q) = A_q$. Code sharing is obtaining of the address A_q as concatenation of OLC code and its component code.

Structure of compositional microprogram control unit with elementary OLC and code sharing can be used for interpretation of graph-scheme of control algorithm (Fig. 1). Let us call this structure U_1 .

Block of microinstruction addressing in CMCU scheme realized function of memory excitation for register RG:

$$\Psi = \Psi(X, \tau). \quad (4)$$

When signal Start is coming initial microprogram address is loaded into RG, zero value is loaded into CT, and flip-flop TF is set to "1" that allows reading microinstructions from control memory. There are two additional internal signals: y_0 and y_E . In case of $y_0 = 1$ content of CT is incremented and next vertex of current operational linear chain is addressed. If $y_0 = 0$ then OLC output is reached and BMA prepares address of next OLC using code of current POLC class. Signal y_E is used at the end of microprogram to reset flip-flop TF. The value "0" of TF output stops access to CM.

Asynchronous reset of counter must be controlled by function $\text{Start} \vee \overline{y_0}$. Signal $\overline{y_0}$ ensures loading zero value to the CT when transition to another OLC performed.

Number of terms in BMA scheme can be decreased by implementation OLC code transformer into POLC class codes [2]. But such realization demands extra FPGA recourses.

In the article complexity of code transformer is proposed to decrease by using free recourses of embedded memory.

MAIN IDEA OF PROPOSED METHOD

In initial GSA the set C_j contains OLC α_g , which are not connected to the end vertex of GSA. All operational

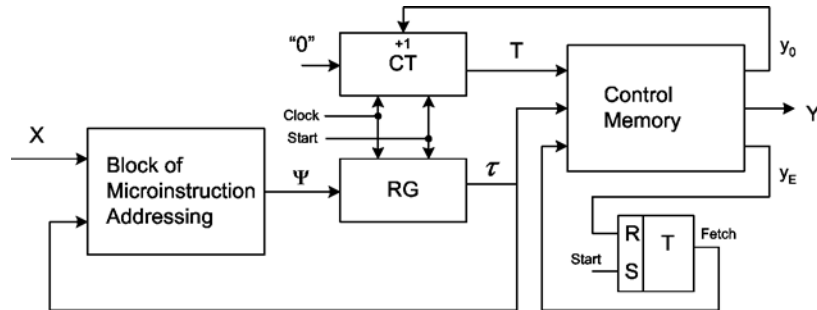


Fig. 1. Structure of compositional microprogram control unit with elementary OLC and code sharing

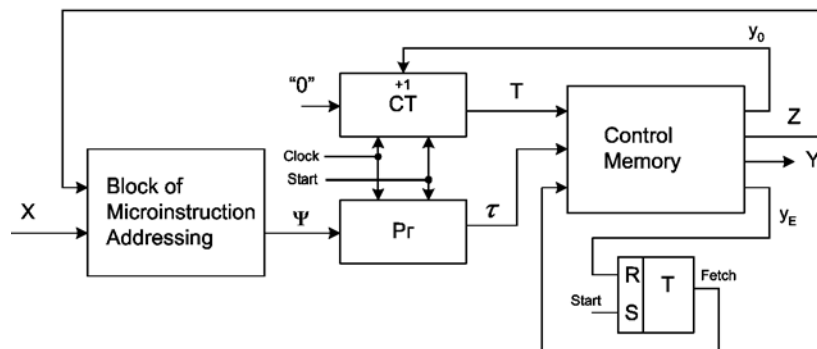


Fig. 2. Structure of compositional microprogram control unit with elementary OLC and code sharing after microinstruction format extension

linear chains are divided into classes $B_i \in \Pi_C$ of POLC. Binary code $K(B_i)$ of width R_3 is set to each class B_i , where

$$R = \lceil \log_2 I \rceil. \tag{5}$$

In (5) I is number of POLC classes. Let control memory of CMCU is realized on blocks of embedded memory with t output pins. Using unitary method of microinstructions encoding [2] we need

$$n_1 = N + 2, \tag{6}$$

bits in appropriate field, where $N = |Y|$, and constant 2 take into account internal signals y_0 and y_E . So, R_4 bits of the microinstruction may be free, where

$$R_4 = \left\lceil \frac{n_1}{t} \right\rceil * t - n_1. \tag{7}$$

If the condition

$$R_4 \geq R_3, \tag{8}$$

takes place, a field FB with the code of POLC class can be included into microinstruction format. Structure U_2 is obtained (Fig. 2).

In CMCU U_2 variables $z_r \in Z$, where $|Z| = R_3$, is bits of code $K(B_i)$. Block of microinstruction addressing performed function

$$\Psi = \Psi(Z, X). \tag{9}$$

Other blocks of CMCU U_2 perform corresponding functions to functions of CMCU U_1 blocks. Let us point out that structural elements BMA, CT, RG, TF is realized in LUT-elements, and CM is implemented in embedded memory.

The following method of CMCU U_2 synthesis is proposed in this article:

1. Construction of the sets C , C_1 , and Π_C for a GSA Γ .
2. Encoding of OLC, their components and classes $B_i \in \Pi_C$.
3. Construction of the content of control memory.
4. Construction of CMCU transition table and $\Psi = \Psi(Z, X)$ functions.
5. Synthesis of CMCU logic circuit.

EXAMPLE OF METHOD USING

Let GSA Γ_1 (Fig. 3) be characterized by next sets: $C = \{\alpha_1, \dots, \alpha_5\}$ – elementary OLC, $C_1 = C \setminus \alpha_5$ OLC without connection to the end vertex, $\Pi_C = \{B_1, \dots, B_3\}$ – classes of pseudoequivalent operational linear elementary chains, where $B_1 = \{\alpha_1\}$, $B_2 = \{\alpha_2, \alpha_3\}$, $B_3 = \{\alpha_4\}$. Number of OLC $G = 5$, $R_1 = 3$ bits from the set $\tau = \{\tau_1, \tau_2, \tau_3\}$ are enough for their encoding. Maximum length of OLC is $Q = 3$,

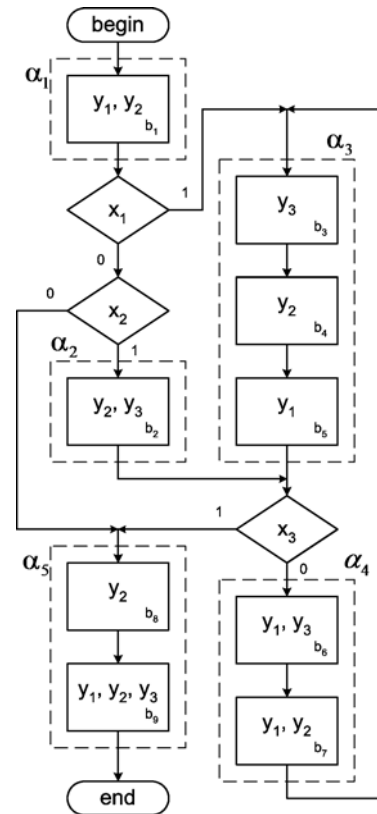


Fig. 3. Initial GSA Γ_1

let us use $R_2 = 2$ variables from the set $T = \{T_1, T_2\}$ for OLC components encoding. Total number of operational vertices is $M = 9$, this number demands $R = 4$ bit of address in CM. For encoding $I = 3$ classes $B_i \in \Pi_C$ of POLC $R_3 = 2$ bits are used.

Let us encode OLC $\alpha_g \in C$ and their components in arbitrary manner (3). Adresse $A(b_q)$ of CMCU $U_2(\Gamma_1)$ microinstruction are shown in Table 1. Here and after symbol $U_i(\Gamma_j)$ means, that CMCU U_i interprets GSA Γ_j .

Table 1. Addresses of CMCU $U_2(\Gamma_1)$ microinstruction

$T_1 T_2$ \ τ_1, τ_2, τ_3	000	001	010	011	100
00	b_1	b_2	b_3	b_6	b_8
01	–	–	b_4	b_7	b_9
10	–	–	b_5	–	–
11	–	–	–	–	–

From Table 1 one can obtain addresses, for example: $A(b_6) = 01100$, $A(b_9) = 10001$ and so on.

Codes of classes $B_i \in \Pi_C$ are set as $K(B_1) = 00, \dots, K(B_3) = 10$. Microinstruction format of CMCU U_2 includes fields y_0, y_E, FY, FB , where field FY contains

code of micro-operation set, FB – code of class $B_i \in \Pi_C$. If $y_0 = 1$, contents of FB field is ignored.

Let GSA Γ_1 includes $N = 3$ different microoperation y_n , and memory blocks with $t = 4$ output are used for realization of control memory in FPGA basis [6, 7]. In this case formula (7) gives us $R_4 = 3$ free bits. Because of $R_4 > R_3$, usage of proposed method is possible. So, in example $Z = \{z_1, z_2\}$. Contents of CMCU $U_2(\Gamma_1)$ control memory is shown in Table 2.

Table 2. Contents of CMCU $U_2(\Gamma_1)$ control memory

$A(b_q)$	y_0	FY	y_E	FB
$A(b_1)$	0	y_1, y_2	0	00
$A(b_2)$	0	y_2, y_3	0	01
$A(b_3)$	1	y_3	0	–
$A(b_4)$	1	y_2	0	–
$A(b_5)$	0	y_1	0	01
$A(b_6)$	1	y_1, y_3	0	–
$A(b_7)$	0	y_1, y_2	0	10
$A(b_8)$	1	y_2	0	–
$A(b_9)$	0	y_1, y_2, y_3	1	–

If vertex $b_q \in E_1$ is not an output of current OLC $\alpha_g \in C_1$, in memory cell with address $A(b_q)$ microoperation y_0 is written. In opposite case in FB field of this cell code $K(B_i)$ is written, where $\alpha_g \in B_i$. If vertex $b_q \in E_1$ is connected with the end of GS A than in memory cell with address $A(b_q)$ internal microoperation y_E is written.

Transitions from outputs of OLC $\alpha_g \in C_1$ are expressed by next system of formulae [2]:

$$\begin{aligned} B_1 &\rightarrow x_1 b_3 \vee \overline{x_1} x_2 b_2 \vee \overline{x_1} \overline{x_2} b_8; \\ B_2 &\rightarrow x_3 b_8 \vee \overline{x_3} b_6; \\ B_3 &\rightarrow b_3. \end{aligned} \tag{10}$$

Such system is the base for CMCU U_2 transition table formation. This table consists of next columns: $B_i, K(B_i), b_q, A(b_q), X_h, \Psi_h, h$. Their purpose became clear from Table 3.

Table 3. Fragment of CMCU $U_2(\Gamma_1)$ transition table

B_i	$K(B_i)$		b_q	$A(b_q)$			X_h	Ψ_h	h
	z_2	z_1		τ_1	τ_2	τ_3			
B_1	0	0	b_2	0	0	1	$\overline{x_1} x_2$	D_3	1
			b_3	0	1	0	x_1	D_2	2
			b_8	1	0	0	$\overline{x_1} \overline{x_2}$	D_1	3

Addresses of microinstruction is taken from Table 1. Let us point out, that system of memory excitation functions Ψ includes functions $\{D_1, D_2, D_3\}$. Total number of rows $H_2(\Gamma_j)$ in transition table of CMCU $U_2(\Gamma_j)$ is equal to number of terms in system transition formulae. In our example, $H_2(\Gamma_1) = 6$.

System (9) is formed according to transition table. Fragments of system Ψ can be found from Table 3:

$$\begin{aligned} D_1 &= \overline{z_1} \overline{z_2} \overline{x_1} \overline{x_2}; \quad D_2 = \overline{z_1} \overline{z_2} x_1; \\ D_3 &= \overline{z_1} \overline{z_2} x_1 x_2. \end{aligned} \tag{11}$$

For minimization of terms number in (9) classes $B_i \in \Pi_C$ may be encoded with the help of EXPRESSO algorithm, for example.

Realization of logical circuit of CMCU U_2 reduces to implementation of system (9) in base of integrated circuit (FPGA) and realization of control memory on blocks of embedded or external memory. Modern CAD systems or methods [1, 2] can be used for this purpose.

CONCLUSIONS

Proposed method of microinstruction format extension for compositional microprogram control unit is oriented to LUT-elements decrease in the block of microinstruction addressing. Number of memory blocks in device and its working time are the same as for base structure CMCU U_1 with code sharing.

Disadvantage of proposed method is in its usage limitation (8).

Term number decrease in memory excitation functions can lead to decrease number of circuit levels in combinational part of device, that can increase speed of work.

Scientific novelty of proposed method modification is in usage of POLC classes and free recourses of control memory for LUT-elements number decrease in block of microinstruction addressing. Practical meaning is in chip parameters decrease. It allows realization of device with less cost.

Our future work is directed to development of CAD system for synthesis of compositional microprogram control units [4].

LIST OF REFERENCES

1. Соловьев В. В. Проектирование цифровых схем на основе программируемых логических интегральных схем. – М. : Горячая линия – ТЕЛЕКОМ, 2001. – 636 с.
2. Баркалов А. А. Синтез устройств управления на программируемых логических устройствах. – Донецк : ДНТУ, 2002. – 262 с.
3. Грушвицкий Р. И., Мурсаев А. Х., Узрюмов Е. П. Проектирование систем с использованием микросхем программируемой логики. – СПб : БХВ-Петербург, 2002. – 608 с.

4. Баркалов А. А., Тутаренко Л. А. Синтез микропрограммных автоматов на заказных и программируемых СБИС. – Донецк : УНИТЕХ, 2009. – 336 с.
5. De Micheli G. Synthesis and Optimization of Digital Circuits. – NY : McGraw-Hill, 1994. – 636 pp.
6. Virtex-6 Family Overview [Электронный ресурс] : Advance Product Specification / XILINX. – Электрон. дан. (1 файл). – [S. l.] : XILINX, 2010. – Режим доступа: http://www.xilinx.com/support/documentation/data_sheets/ds150.pdf, свободный. – Загл. с экрана. – Англ. яз.
7. Stratix III FPGA: Lowest Power, Highest Performance 65-nm FPGA [Электронный ресурс] / Altera. – Электрон. дан. – [S. l.] : Altera, 2010. – Режим доступа: <http://www.altera.com/products/devices/stratix-fpgas/stratix-iii/st3-index.jsp>, свободный. – Загл. с экрана. – Англ. яз.

Надійшла 5.04.2010

Баркалов О. О., Зеленёва І. Я., Мірошкін О. М.

РОЗШИРЕННЯ ФОРМАТУ МІКРОКОМАНД У КОМПОЗИЦІЙНОМУ МІКРОПРОГРАМНОМУ ПРИСТРОЇ КЕРУВАННЯ ІЗ ЕЛЕМЕНТАРІЗАЦІЄЮ ОПЕРАТОРНИХ ЛІНІЙНИХ ЛАНЦЮГІВ

Модифікація методу синтезу композиційного мікропрограмного пристрою керування спрямована на зменшення апаратних витрат при реалізації у FPGA базисі. Зменшення складності блоку адресації мікрокоманд досягається завдяки

полю, що містить код класу псевдоеквівалентного операторного лінійного ланцюга. Приведені умови доцільності та приклад використання модифікації методу синтезу.

Ключові слова: композиційне мікропрограмне устройство управления, операторная линейная цепь, адресация микрокоманд, класс псевдоеквивалентности.

Баркалов А. А., Зеленёва И. Я., Мирошкин А. Н.

РАСШИРЕНИЕ ФОРМАТА МИКРОКОМАНД В КОМПОЗИЦИОННОМ МИКРОПРОГРАММНОМ УСТРОЙСТВЕ УПРАВЛЕНИЯ С ЭЛЕМЕНТАРИЗАЦИЕЙ ОПЕРАТОРНЫХ ЛИНЕЙНЫХ ЦЕПЕЙ

Модификация метода синтеза композиционного микропрограмного устройства управления направлена на уменьшение аппаратных затрат при реализации в FPGA базисе. Уменьшение сложности блока адресации микрокоманд достигается за счет поля, содержащего код класса псевдоеквивалентной операторной линейной цепи. Приведены условия целесообразности и пример применения предложенной модификации метода синтеза.

Ключевые слова: композиционный микропрограммный пристрой керування, операторний лінійний ланцюг, адресация мікрокоманд, клас псевдоеквівалентності.

УДК 002.53+681.3(075.8)

Бойченко О. В.

Канд. техн. наук, професор Кримського юридичного інституту Одеського державного університету внутрішніх справ (м. Сімферополь)

КООРДИНАЦІЯ НЕЧІТКИХ РІШЕНЬ В БАГАТОРІВНЕВІЙ ІЄРАРХІЧНІЙ СИСТЕМІ

Проаналізовано методологічні засади керування складними багаторівневими інформаційними системами в умовах швидкої зміни порядку їх застосування. Запропоновано застосування комплексного системного підходу щодо координації прикладного (сеансового) та базового рівнів складних розгалужених інформаційних систем для оптимізації їхнього функціонування в умовах можливості виникнення нечітких рішень.

Ключові слова: багаторівневі ієрархічні системи, координація нечітких рішень, математичне моделювання та прогноз.

ПОСТАНОВКА ПРОБЛЕМИ

Аналіз практики застосування в діяльності організацій та установ розподілених інформаційно-телекомунікаційних систем визначає можливість виникнення обставин відсутності координації між базовим та прикладним рівнями управлінської ієрархії, що зменшує ефективність оперативного та достовірного обміну даними в діяльності закладу та потребує розробки відповідних заходів оптимізації функціонування комп'ютерної системи через застосування сучасних методів математичного моделювання та прогнозу з метою вирішення проблеми координації нечітких рішень в багаторівневій ієрархічній системі.

© Бойченко О. В., 2010

АНАЛІЗ ОСТАННІХ ДОСЛІДЖЕНЬ ТА ПУБЛІКАЦІЙ

Вирішенню проблем оптимізації складних ієрархічних систем управління стосовно координації нечітких рішень через застосування рекурентної процедури та процесу корекції в теорії нечітких множин присвячені роботи таких видатних фахівців, як Белман Р., Заде Л., Месарович М., Такахара Я., Моїсєєв Н. та інших [1–3]. Разом з тим, наявність низки проблем практики застосування управлінських ієрархічних систем визначає необхідність проведення подальших наукових досліджень з метою розробки відповідних заходів для їх вирішення. Наукова новизна запропонованої праці полягає в формуванні шляхів до