

ОПТИМИЗАЦИЯ СХЕМЫ КОМПОЗИЦИОННОГО МИКРОПРОГРАММНОГО УСТРОЙСТВА УПРАВЛЕНИЯ С РАЗДЕЛЕНИЕМ КОДОВ

А.А. Баркалов, Р.В.Мальчева, А.А.Красичков, Халед Баракат

Зеленогурский университет (Польша),
Донецкий национальный технический университет
A.Barkalov@iie.uz.zgora.pl

Аннотация

Предлагается метод синтеза КМУУ с разделением кодов, позволяющий сохранить минимальную емкость управляющей памяти при произвольных характеристиках реализуемого алгоритма управления. Также приведен пример применения предлагаемого метода.

1. Постановка задачи и цели исследования

Устройство управления (УУ) любой цифровой системы может быть реализовано в виде композиционного микропрограммного устройства управления (КМУУ) [1,2]. В настоящее время для реализации схем устройств управления широко используются программируемые логические устройства типа CPLD и FPGA [3,4]. При этом традиционно актуальной является задача оптимизации аппаратных затрат в схеме устройства управления [4,5]. В случае применения CPLD или FPGA эта задача может решаться путем уменьшения числа термов в системе функций, реализуемых УУ [6]. Одним из методов решения этой задачи в случае КМУУ является метод разделения кодов [2]. Однако этот метод применим только при выполнении определенных условий, нарушении которых ведет к резкому увеличению емкости управляющей памяти КМУУ [2]. В настоящей работе предлагается метод синтеза КМУУ с разделением кодов, позволяющий сохранить минимальную емкость управляющей памяти при произвольных характеристиках реализуемого алгоритма управления.

2. Основные определения и общая постановка задачи

Пусть алгоритм управления цифровой системы представлен в виде граф-схемы алгоритма (ГСА) Γ [2] с начальной вершиной b_0 , конечной вершиной b_E , операционными вершинами из множества V_1 и условными вершинами из множества V_2 . Вершины ГСА Γ образуют множество $V = V_1 \cup V_2 \cup \{b_0, b_E\}$, связанных дугами $\langle b_i, b_q \rangle$, образующими множество дуг E . Введем ряд определений, используя результаты работы [1].

Определение 1. Операторной линейной цепью (ОЛЦ) ГСА Γ называется конечная последовательность операторных вершин $\alpha_g = \langle b_{g1}, \dots, b_{gF_g} \rangle$, такая, что существует дуга $\langle b_{gi}, b_{gi+1} \rangle \in E$ для каждой пары соседних компонент вектора α ($i=1, \dots, F_g-1$).

Определение 2. Операторная вершина $b_q \in D^g$, где D^g – множество вершин, входящих в ОЛЦ α_g , называется входом ОЛЦ α_g , если существует дуга $\langle b_t, b_q \rangle \in E$, такая, что $b_t = b_0$ или $b_t \in V_2$ или $b_t \notin D^g$.

Определение 3. Операторная вершина $b_q \in D^g$ называется выходом ОЛЦ α_g , если существует дуга $\langle b_t, b_q \rangle \in E$, такая, что $b_t = b_E$ или $b_t \in V_2$ или $b_t \notin D^g$.

Пусть для ГСА Γ сформировано множество ОЛЦ $C = \{\alpha_1, \dots, \alpha_G\}$, удовлетворяющее условию

$$\begin{aligned} D^i \cap D^j &= \emptyset \quad (i \neq j; i, j \in \{1, \dots, G\}); \\ D^1 \cup D^2 \cup \dots \cup D^G &= B_1; \\ G &\Rightarrow \min. \end{aligned} \quad (1)$$

При выполнении условия (1) каждая операторная вершина исходной ГСА входит ровно в одну ОЛЦ $\alpha_g \in C$. Пусть каждой вершине $b_q \in V_1$ соответствует двоичный код $A(b_q)$ и для компонент любой ОЛЦ $\alpha_g \in C$ выполняется условие

$$\begin{aligned} A(b_{g+1}) &= A(b_{g_i}) + 1 \\ (i &= 1, F_g - 1; g = 1, G). \end{aligned} \quad (2)$$

В этом случае для интерпретации ГСА Γ может быть использовано КМУУ U_1 (рис.1), называемое КМУУ с общей памятью [2].

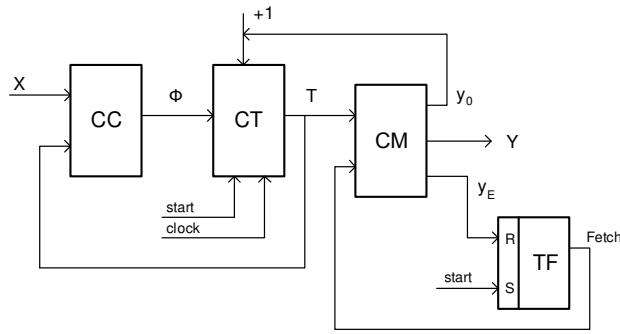


Рис. 1. Структурная схема КМУУ с общей памятью

Здесь комбинационная схема CC формирует систему функций возбуждения счетчика CT

$$\Phi = \Phi(X, T), \quad (3)$$

где $X = \{x_1, \dots, x_L\}$ – множество логических условий, записанных в условных вершинах ГСА Γ , $T = \{T_1, \dots, T_R\}$ – множество внутренних переменных, используемых для адресации микрокоманд, хранящихся в управляющей памяти CM ,

$$R = \lceil \log_2 M \rceil, \quad (4)$$

где $M = |B_1|$. Система (3) формирует адрес $A(I_g^j)$ j -го входа ОЛЦ $\alpha_g \in C$. Управляющая память CM содержит микрооперации $y_n \in Y$, где $Y = \{y_1, \dots, y_N\}$ – множество микроопераций, записанных в операторных вершинах ГСА Γ . Сигнал y_0 используется для управления счетчиком CT : если $y_0 = 1$, то выполняется операция $CT := CT + 1$; если $y_0 = 0$, то содержимое CT определяется функциями (3). Сигнал y_E используется для организации режима останова U_1 .

Устройство управления U_1 функционирует следующим образом. По сигналу $start=1$ в CT записывается адрес первой микрокоманды интерпретируемой микропрограммы. Одновременно триггер выборки TF устанавливается в единицу и сигнал $Fetch=1$ разрешает выборку микрокоманды из CM . Если в CT находится адрес $A(b_q)$ микрокоманды $Y(b_q)$ и $b_q \neq O_g$, где O_g – выход ОЛЦ $\alpha_g \in C$, то CM формирует соответствующие микрооперации и сигнал y_0 . Таким образом происходят переходы между микрокомандами, соответствующими компонентам одной ОЛЦ. Если $b_q = O_g$, то $y_0 = 0$ и адрес перехода формируется системой (3). При достижении микрокоманды $Y(b_i)$, такой, что существует дуга $\langle b_i, b_E \rangle \in E$, формируется сигнал y_E , триггер TF обнуляется и выборка микрокоманд прекращается.

Устройство управления U_1 является автоматом Мура, память которого реализована на счетчике. При этом содержимое CT интерпретируется как код состояния, совпадающий с адресом выхода ОЛЦ. Если число R_1 переменных, требуемых для кодирования ОЛЦ $\alpha_g \in C$

$$R_1 = \lceil \log_2 G \rceil, \quad (5)$$

меньше R , то аппаратурные затраты в схеме CC можно уменьшить, применив метод разделения кодов [2], приводящий к КМУУ U_2 (рис.2).

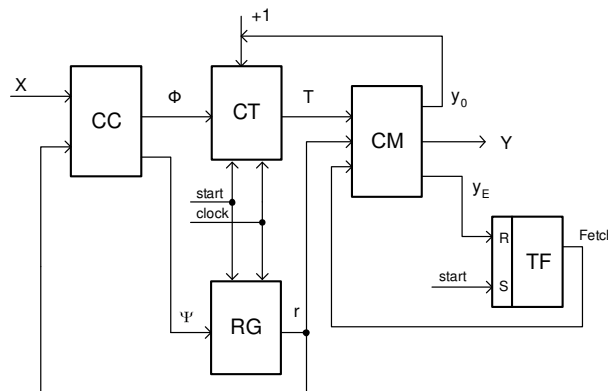


Рис. 2. Структурная схема КМУУ с разделением кодов

Здесь адрес $A(b_q)$ микрокоманды $Y(b_q)$ определяется как конкатенация

$$A(b_q) = K(\alpha_g) * K(b_q), \quad (6)$$

где $K(\alpha_g)$ – код ОЛЦ $\alpha_g \in C$ разрядности R_1 , $K(b_q)$ – код вершины $b_q \in D^g$ ($g=1, \dots, G$) разрядности

$$R_2 = \lceil \log_2 F_{\max} \rceil \quad (7)$$

где F_{\max} – максимальная длина ОЛЦ $F_{\max} = \max(F_1, \dots, F_G)$, * – знак конкатенации. Принципы функционирования КМУУ U_1 и U_2 идентичны, но комбинационная схема КМУУ U_2 формирует системы функций

$$\Phi = \Phi(\tau, X), \quad (8)$$

$$\Psi = \Psi(\tau, X), \quad (9)$$

где Ψ – множество функций возбуждения регистра RG, хранящего код $K(\alpha_g)$, представленный переменными τ , где $|\tau| = R_1$. Кроме того, условие (2) превращается в условие

$$K(b_{g_{i+1}}) = K(b_{g_i}) + 1 \\ (i = 1, F_g - 1; g = 1, G). \quad (10)$$

Такой подход позволяет использовать известные методы оптимизации автомата Мура [7] для уменьшения аппаратных затрат в схеме СС. Но при нарушении условия

$$R_1 + R_2 = R \quad (11)$$

емкость управляющей памяти КМУУ U_2 резко увеличивается по сравнению с соответствующим параметром КМУУ U_1 . В настоящей работе предлагается метод, позволяющий использовать метод разделения кодов при нарушении условия (11).

3. Метод решения задачи

Пусть для ГСА Γ условие (11) нарушается, то есть

$$R_1 + R_2 > R. \quad (12)$$

Введем в схему КМУУ U_2 преобразователь адреса АТ и получим КМУУ U_3 (рис.3).

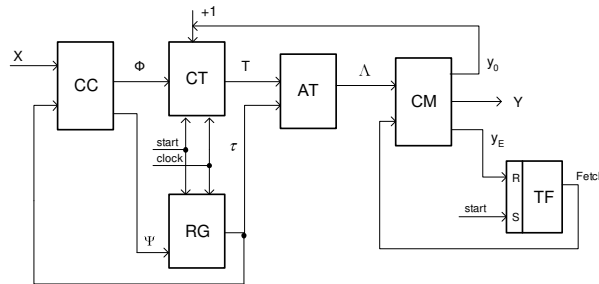


Рис. 3. Структурная схема КМУУ с разделением кодов и преобразователем адреса

Здесь преобразователь АТ реализует систему функций

$$\Lambda = \Lambda(T, \tau), \quad (13)$$

где $\Lambda = \{\lambda_1, \dots, \lambda_R\}$ – множество переменных, кодирующих адреса микрокоманд в управляющей памяти. Функции остальных структурных элементов схемы КМУУ U_3 и U_2 совпадают.

В настоящей работе предлагается метод синтеза КМУУ U_3 , включающий следующие этапы:

1. Преобразование исходной ГСА.
2. Формирование удовлетворяющего (1) множества ОЛЦ С.
3. Кодирование операторных линейных цепей $\alpha_g \in C$.
4. Кодирование операторных вершин преобразованной ГСА.
5. Адресация микрокоманд.
6. Формирование содержимого управляющей памяти.
7. Формирование таблицы переходов КМУУ.
8. Формирование таблицы преобразователя адресов.
9. Формирование систем функций Φ, Ψ, Λ .
10. Реализация схем КМУУ в заданном элементе базисе.

4. Пример применения предложенного метода

Рассмотрим применение предложенного метода на примере ГСА Γ_1 (рис.4).

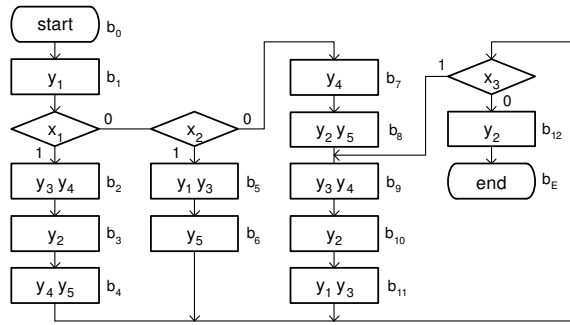


Рис. 4. Исходная граф-схема алгоритма Γ_1

Преобразование исходной ГСА. Преобразование сводится к введению сигнала u_E в операторные вершины b_q , такие, что $\langle b_q, b_E \rangle \in E$. Если существует дуга $\langle b_i, b_E \rangle \in E$, такая, что $b_i \in B_2$, то вводится дополнительная вершина b_q , содержащая u_E , и дуги $\langle b_i, b_q \rangle, \langle b_q, b_E \rangle$ заменяют исходную дугу $\langle b_i, b_E \rangle$. Если существует дуга $\langle b_0, b_q \rangle \in E$, такая, что $b_q \in B_2$, то вводится дополнительная пустая вершина b_i , содержащая u_E , и дуги $\langle b_0, b_i \rangle$ и $\langle b_i, b_q \rangle$ заменяют исходную дугу $\langle b_0, b_q \rangle$.

В случае ГСА Γ_1 преобразование сводится к введению сигнала u_E в вершину b_{12} .

Формирование множества ОЛЦ. Для этого используется стандартная процедура из [1]. В нашем случае $C = \{\alpha_1, \dots, \alpha_5\}$, где $\alpha_1 = \langle b_1 \rangle$, $\alpha_2 = \langle b_2, b_3, b_4 \rangle$, $\alpha_3 = \langle b_5, b_6 \rangle$, $\alpha_4 = \langle b_7, \dots, b_{11} \rangle$, $\alpha_5 = \langle b_{12} \rangle$. $I_1^1 = O_1 = b_1$, $I_2^1 = b_2$, $O_2 = b_4$, $I_3^1 = b_5$, $O_3 = b_6$, $I_4^1 = b_7$, $I_4^2 = b_9$, $O_4 = b_{11}$, $I_5^2 = O_5 = b_{12}$.

Анализ множества C показывает, что $G=5$, $F_1=F_5=1$, $F_2=3$, $F_3=2$, $F_4=5$, $F_{\max}=5$. Следовательно, для кодирования ОЛЦ $\alpha_g \in C$ достаточно $R_1=3$ разряда, для кодирования компонент ОЛЦ достаточно $R_2=3$. Для адресации микрокоманд достаточно $R=4$ адресных разряда. Следовательно, условие (11) нарушается и применение модели КМУУ U_3 целесообразно.

Кодирование операторных линейных цепей. Этот этап выполняется аналогично оптимальному кодированию псевдоэквивалентных состояний автомата Мура [7]. Напомним, что ОЛЦ являются эквивалентными, если их выходы связаны с входом одной и той же вершины ГСА. Для оптимизации аппаратных затрат коды псевдоэквивалентных ОЛЦ должны входить в один обобщенный интервал R_1 – мерного булева пространства [2].

В нашем случае разбиение Π_C множества ОЛЦ C на классы псевдоэквивалентных ОЛЦ включает блоки $V_1 = \{\alpha_1\}$, $V_2 = \{\alpha_2, \alpha_3, \alpha_4\}$, $V_3 = \{\alpha_5\}$. Закодируем ОЛЦ $\alpha_g \in C$ следующим образом: $K(\alpha_1) = 000$, $K(\alpha_2) = 001$, $K(\alpha_3) = 011$, $K(\alpha_4) = 101$, $K(\alpha_5) = 010$. Теперь с учетом неопределенностей классу V_1 соответствует код $K(V_1) = *00$, классу V_2 – $K(V_2) = **1$, классу V_3 – $K(V_3) = *10$.

Кодирование операторных вершин ГСА. Кодирование выполняется так, чтобы выполнялось условие (10). Используя метод [2], получим $K(b_1) = 000$, $K(b_2) = 001$, $K(b_3) = 011$, $K(b_4) = 101$, $K(b_5) = 010$, $K(b_6) = 000$, $K(b_7) = 001$, $K(b_8) = 011$, $K(b_9) = 101$, $K(b_{10}) = 011$, $K(b_{11}) = 100$, $K(b_{12}) = 000$.

Адресация микрокоманд. Адреса микрокоманд могут быть присвоены в произвольном порядке, так как они формируются преобразователем АТ. При этом микрокомандам $Y(b_q)$ и $Y(b_i)$ могут быть присвоены одинаковые адреса, если содержимое вершин b_q и b_i совпадает. Это может привести к оптимизации емкости управляющей памяти при ее реализации на FPGA.

Пусть в нашем случае $A(b_1) = 0000$, $A(b_2) = 0001$, ..., $A(b_{12}) = 1011$

Формирование содержимого управляющей памяти. Содержимое СМ представляется таблицей, каждая строка которой содержит адрес микрокоманды и входящие в нее микрооперации. В микрокоманде $Y(b_q)$, где $b_q \neq O(\Gamma)$, вводится сигнал y_0 , здесь $O(\Gamma)$ – множество выходов ОЛЦ $\alpha_g \in C$. Например, микрокоманде $Y(b_1)$ соответствует строка с адресом 0000 и содержимым y_1 , микрокоманде $Y(b_2)$ – с адресом 0001 и содержимым y_0, y_3, y_4 и т.д.

Формирование таблицы переходов КМУУ. Для формирования этой таблицы необходимо построить систему формул перехода вида

$$B_i \rightarrow \bigvee X_i(I_g^j) I_g^j \quad (B_i \in \Pi_C'), \quad (14)$$

где $X_i(I_g^j)$ – конъюнкция логических условий, определяющая переход из выхода любой ОЛЦ $\alpha_g \in V_i$ в операторную вершину, соответствующую входу I_g^j , $\Pi_C' \in \Pi_C$ – множество классов ОЛЦ, выходы которых не связаны с конечной вершиной ГСА.

В случае ГСА Γ $\Pi_C' = \{V_1, V_2\}$ и формулы (14) имеют следующий вид:

$$B_1 \rightarrow x_1 I_2^1 \vee \overline{x_1} x_2 I_3^1 \vee \overline{x_1} \overline{x_2} I_4^1,$$

$$B_2 \rightarrow x_3 I_4^2 \vee \overline{x_3} I_5^1.$$

Таблица переходов КМУУ U_3 включает столбцы: B_i , $K(B_i)$, I_g^j , $A(I_g^j)$, X_h , Φ_h , Ψ_h , h , где $A(I_g^j)$ – адрес входа I_g^j ; $X_h \equiv X_i(I_g^j)$; Φ_h – набор функций возбуждения СТ, принимающих единичное значение для формирования в СТ кода компоненты ОЛЦ α_g , соответствующей входу I_g^j ; Ψ_h – набор функций возбуждения RG кода ОЛЦ $\alpha_g \in C$; $h = \overline{1, N}$ – номер перехода. В случае ГСА Γ_1 таблица переходов КМУУ U_3 имеет $N=5$ строк (табл.1)

Таблица 1

Таблица переходов КМУУ U_3

B_i	$K(B_i)$	I_g^j	$A(I_g^j)$	X_h	Φ_h	Ψ_h	h
B_1	*00	I_2^1	001000	x_1	–	D_3	1
		I_3^1	011000	$\overline{x_1} x_2$	–	D_2 D_3	2
		I_4^1	101000	$\overline{x_1} \overline{x_2}$	–	D_1 D_3	3
B_2	**1	I_4^2	101010	x_3	D_5	D_1 D_3	4
		I_5^2	010000	$\overline{x_3}$	–	D_2	5

Как видно из таблицы 1, регистр RG и счетчик СТ имеют информационные входы D-типа: $\Psi = \{D_1, D_2, D_3\}$, $\Phi = \{D_4, D_5, D_6\}$.

Формирование таблицы преобразователя адреса. Эта таблица отображает систему функций (13) и имеет столбцы: b_q , $K(\alpha_g)$, $K(b_q)$, $A(b_q)$, λ_h , h . Здесь λ_h – набор адресных разрядов, равных единице для h -й строки таблицы ($h = \overline{1, N_1}$). Для ГСА Γ_1 таблица AT имеет $N_1=12$ строк и $N_1 = |B_1|$. Фрагмент таблицы AT приведен в таблице 2.

Таблица 2 – Фрагмент таблицы преобразователя адреса

b_q	$K(\alpha_g)$	$K(b_q)$	$A(b_q)$	λ_h	h
b_1	000	000	0000	–	1
b_2	001	000	0001	λ_4	2
b_3	001	001	0010	λ_3	3
b_4	001	010	0011	$\lambda_3 \lambda_4$	4

Формирование систем функций Φ, Ψ, λ . Функции Φ и Ψ формируются по таблице переходов КМУУ.

Например, $D_1 = \overline{\tau_2} \overline{\tau_3} x_1 x_2 \vee \tau_3 x_3$, что соответствует третьей и четвертой строкам табл.1. Функции $\lambda_r \in \lambda$ формируются по таблице AT. Например, из табл.2 имеем $\lambda_3 = \overline{\tau_1} \overline{\tau_2} \tau_3 \overline{T_1} \overline{T_2} T_3 \vee \overline{\tau_1} \tau_2 \tau_3 \overline{T_1} T_2 T_3 = \overline{\tau_1} \overline{\tau_2} \tau_3 \overline{T_1} T_2$, что соответствует третьей и четвертой строкам табл.2.

Реализация схемы КМУУ в заданном элементном базисе. Схема СС реализуется на FPGA или CPLD по системам (8) – (9), схема преобразователя адресов AT реализуется на FPGA или CPLD по системе (13), управляющая память СМ, как правило, реализуется с использованием памяти, входящей в состав кристаллов с FPGA и CPLD [4,5]. Вопросы реализации подобных схем достаточно полно рассмотрены в литературе [4,8] и выходят за рамки данной статьи.

5. Заключение и результаты исследований

Применение метода преобразования адресов позволяет использовать метод разделения кодов при синтезе композиционных микропрограммных устройств управления при любом сочетании разрядности кодов операторных линейных цепей, их компонент и минимальной разрядности адреса микрокоманды. Научная новизна предлагаемого метода заключается в том, что сохраняется возможность оптимального кодирования псевдоэквивалентных ОЛЦ, что позволяет уменьшать аппаратные затраты в схеме, формирующей коды ОЛЦ и их компонент. Практическое значение метода заключается в том, что применение преобразователя адресов

позволяет уменьшить емкость управляющей памяти за счет произвольной адресации по сравнению с известными методами реализации устройств-аналогов. К недостаткам предложенного метода относится увеличение времени цикла КМУУ за счет введения схемы АТ. Исследования авторов показали, что при выполнении условия (12), оптимальном кодировании операторных линейных цепей и уменьшении емкости управляющей памяти предлагаемый метод позволяет на 12-16% уменьшить аппаратные затраты по сравнению с КМУУ с общей памятью. Применение метода является перспективным и для КМУУ с элементарными ОЛЦ.

Литература:

1. Баркалов А.А., Палагин А.В. Синтез микропрограммных устройств управления. – Киев: ИК НАН Украины, 1997. – 135 с.
2. Баркалов А.А. Синтез устройств управления на программируемых логических устройствах. – Донецк: ДонНТУ, 2002. – 262 с.
3. Грушвицкий Р.И., Мурсаев А.Х., Угрюмов Е.П. Проектирование систем с использованием микросхем программируемой логики. – Петербург: БХВ, 2002.
4. T.Luba. Synteza ukladow logicznych. – Warszawa: WSIZ, 2001. – 238 s.
5. Z.Salcic. VHDL and FPLDs in Digital systems Design, prototyping and Customization. – Kluwer Academic Publishers, 1998. – 412 pp.
6. T. Sasao. Switching Theory for Logic Synthesis. – Kluwer Academic Publishers, 1999. – 316 pp.
7. Баркалов А.А. Принципы оптимизации логической схемы микропрограммного автомата Мура. / Кибернетика и системный анализ. – 1998, №1. – с. 65 – 72.
8. Соловьев В.В. Проектирование цифровых схем на основе программируемых логических интегральных схем. – М.: Горячая линия-ТЕЛЕКОМ, 2001. – 636 с.

Поступила в редколлегию 06.12.05

Рецензент:

Баркалов Александр Александрович, д.т.н., профессор, профессор кафедры ЭВМ ДонНТУ, профессор Университета Зеленогурского (Польша). Цифровые устройства управления. Научная работа, спорт. 83122, г.Донецк, ул. Артема, д.204А, кв.105. (+38062)301-07-35

Мальчева Раиса Викторовна, к.т.н., доцент, доцент кафедры ЭВМ ДонНТУ. Компьютерная графика. Научная работа, туризм. 83000, г.Донецк, ул. Артема, 58, каф. ЭВМ, к.36. (+38062)301-07-35.

Красичков Алексей Александрович, к.т.н, доцент каф ЭВМ ДонНТУ. Цифровые устройства управления на ПЛУ. Научная работа, спорт, музыка. 83005, г.Донецк, ул. Куприна, д.62, кв.33. (+38062)301-07-23.

Халед Баракат, аспирант каф ЭВМ ДонНТУ. Цифровые устройства управления на ПЛУ. Научная работа. 83000, г.Донецк, ул. Артема, 58, каф. ЭВМ, к.36.

УДК 681.324

Оптимізація схеми композиційного мікропрограмного пристрою керування з розділенням кодів / О.О. Баркалов, Р.В.Мальчева, О.О.Красічков, Халед Баракат // Радіоелектроніка та інформатика. 2006. № 00. С.00-00.

В роботі запропоновано використовувати розділення кодів станів АЖЛ в схемі КМПК. Розглядаються умови використання перетворювача адрес мікрокоманд. Приведено приклад синтезу схеми КМПК з перетворювачем адреси. Проведені дослідження показали, що метод зменшує апаратні витрати на 12-16%.

Табл. 02. Іл. 04. Бібліогр.: 08 назв.

UDC 681.324

Optimization of the compositional microprogram control unit's circuit with codes sharing / A.A. Barkalov, R.V.Malcheva, A.A.Krasichkov, Khaled Barakat // Radioelektronika i informatika. 2006. N 00. P.000-000.

The sharing of the FSM state's codes in the CMCU circuit is offered in the article. The conditions of address transformer usage for addressing of microinstructions are considered. The example of the CMCU circuit synthesis with address transformer is given. The carried out researches have shown that the method reduces hardware expenses on 12-16%.

Tab. 02. Fig. 04. Ref.: 08 items.