

КОМПОЗИЦИОННОЕ МИКРОПРОГРАММНОЕ УСТРОЙСТВО УПРАВЛЕНИЯ С РАСШИРЕНИЕМ ФОРМАТА МИКРОКОМАНД

Мирошкин А.Н., Зеленёва И.Я., Перкин П.В.

ДонНТУ, г. Донецк, Украина

miroshkinan@gmail.com, irina@cs.dgtu.donetsk.ua

Abstract: *In the article approach to hardware amount decreasing in the block of microinstruction addressing of compositional microprogram control unit is proposed. The approach is based on usage of pseudoequivalent operational linear chains and free recourses of FPGA chip. Average decreasing of hardware amount in logical circuits of compositional control units is 26 %. Time delay of circuit gets less.*

Key words: *compositional microprogram control unit, microinstruction format, hardware amount decreasing, FPGA, LUT, optimization.*

Введение

Непрерывное развитие средств вычислительной техники выдвигает высокие требования к разрабатываемым устройствам. Устройства должны обладать высокими показателями быстродействия, надежности, минимальными аппаратными затратами, низкой потребляемой и рассеиваемой мощностями. Кроме того, сам процесс проектирования устройств не должен занимать много времени, иначе устройство морально устареет еще до того, как сойдет с конвейера. В качестве базиса для реализации современных цифровых устройств широко используются программируемые логические интегральные схемы (ПЛИС) типа FPGA (Field-Programmable Gate Array). Совместное использование базиса ПЛИС и специализированных систем автоматизированного проектирования (САПР) позволяет значительно увеличить скорость и гибкость процесса проектирования цифровых устройств. Развитие технологии производства

интегральных схем и постоянное увеличение сложности цифровых систем определяет актуальность создания новых методов синтеза цифровых устройств.

В данной работе рассматривается управляющая часть цифровых устройств – микропрограммное устройство управления [1, 2]. Исходными данными для синтеза устройств управления зачастую служит граф-схема алгоритма (ГСА) [1, 3]. В случае если алгоритм носит линейный характер (доля операторных вершин в ГСА превышает 75%), целесообразно использование модели композиционного микропрограммного устройства управления (КМУУ) [4, 5]. Ряд работ посвящен реализации КМУУ в базисе ПЛИС [6-9].

Данная статья посвящена разработке и исследованию структуры КМУУ, ориентированной на уменьшение аппаратных ресурсов в логической схеме устройства и реализации в базисе ПЛИС типа FPGA. Отличием данной работы является то, что в процессе синтеза КМУУ используются такие особенности ГСА управления, как наличие классов псевдоэквивалентных операторных линейных цепей, а также наличие в современных микросхемах FPGA конфигурируемых блоков встроенной памяти, которые могут иметь свободные ресурсы.

Идея работы заключается в использовании незадействованных ресурсов блоков встроенной памяти микросхем FPGA для хранения кодов классов псевдоэквивалентных операторных линейных цепей, в результате чего уменьшается количество переменных и термов в схеме адресации устройства, а это, в свою очередь, ведет к уменьшению количества используемых ресурсов микросхемы.

Структура КМУУ с расширением формата микрокоманд

Анализ систем адресации КМУУ [1] показал, что одним из существующих подходов к размещению в управляющей памяти данных,

необходимых для формирования адреса перехода, является включение поля адреса наряду с операционной частью в одну микрокоманду (аналог комбинированной адресации).

На основе данного подхода предлагается подход к модификации системы адресации композиционного микропрограммного устройства управления: расширение формата микрокоманд. Предлагаемый подход имеет свои преимущества, однако, существуют условия, которые определяют целесообразность его применения.

Формат микрокоманды базовой структуры КМУУ содержит только операционную часть, что приводит к минимальному размеру микрокоманды и является одним из преимуществ КМУУ перед другими микропрограммными устройствами управления. Расширение формата микрокоманды подразумевает включение в микрокоманду дополнительного поля с кодом $K(B_i)$ класса ПОЛЦ, который будет использоваться схемой формирования адреса перехода. Расширенный формат микрокоманд представлен на рис. 1.

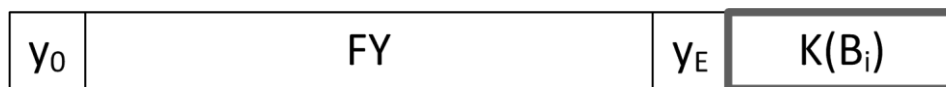


Рисунок 1. Расширенный формат микрокоманды КМУУ

Данная модификация системы адресации микрокоманд предлагается для структуры КМУУ с разделением кодов. При этом в формировании адреса перехода участвует не содержимое регистра P_r , а содержимое поля F_A микрокоманды. Полученную структуру композиционного устройства управления условимся далее называть FCS-структурой КМУУ («F» от англ. «Format of microinstruction») (рис. 2).

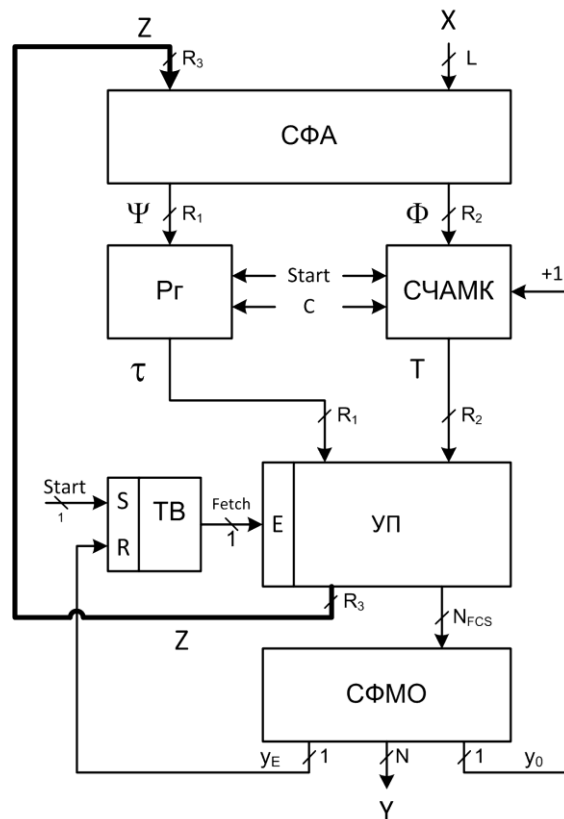


Рисунок 2. Структурная схема КМУУ с расширением формата микрокоманд

На рис. 2 R_1, R_2 – разрядности кода ОЛЦ и ее компонент, где

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

$$R_2 = \lceil \log_2 (F_{\max}) \rceil. \quad (2)$$

В формулах (1), (2) G – количество ОЛЦ, F_{\max} – максимальное количество компонент ОЛЦ.

FCS-структура КМУУ состоит из автомата S_1 с жесткой логикой и автомата S_2 с естественной адресацией микрокоманд. В состав автомата S_1 входят СФА, СЧАМК и УП.

Блок СФА формирует адреса переходов, реализует функции

$$\Psi = \Psi(Z, X); \quad (3)$$

$$\Phi = \Phi(Z, X). \quad (4)$$

Управляющая память наряду с хранением операционной части микропрограммы выполняет функцию преобразователя кодов, который адресу $A(O_g)$ вершины-выхода ОЛЦ α_g ставит в соответствие код $K(B_i)$ класса ПОЛЦ, которому данная ОЛЦ $\alpha_g \in B_i$ принадлежит. Для кодирования I различных ПОЛЦ используются переменные $Z = \{z_1, \dots, z_{R_3}\}$, где R_3 –разрядность кода ПОЛЦ:

$$R_3 = \lceil \log_2 I \rceil. \quad (5)$$

Автомат S_2 состоит из СЧАМК, УП и СФМО. Счетчик обеспечивает последовательную выборку микрокоманд в пределах ОЛЦ.

УП содержит микропрограмму работы автомата, реализует системы

$$Z = Z(\tau, T); \quad (6)$$

$$Y = Y(\tau, T); \quad (7)$$

СФМО декодирует операционную часть микрокоманд, формируя сигналы микроопераций и двух дополнительных внутренних переменных y_0 и y_E . Структура схемы формирования микроопераций зависит от выбранной стратегии кодирования и в случае горизонтального кодирования МО представляет собой набор проводников [1].

Современные микросхемы программируемой логики типа FPGA содержат блоки встроенной памяти, основной размер которых 18k [10-12]. Ряд возможных конфигураций $\{2^{N_A} * N_D\}$ этих блоков выглядит следующим образом:

$$\{16k * 1\}, \{8k * 2\}, \{4k * 4\}, \{2k * 9\}, \{1k * 18\}, \{512 * 36\}, \{256 * 72\}, \quad (8)$$

где N_A , N_D – разрядности адреса и данных соответственно, а конфигурация $256 * 72$ возможна только в режиме однопортовой памяти.

Синтез FCS-структуры КМУУ целесообразен, если количество классов ПОЛЦ меньше количества ОЛЦ, для хранения кода класса ПОЛЦ имеется достаточное количество выходов блоков встроенной памяти, т.е.

$$\begin{cases} I < G; \\ t_{EMB} - N_{CS} \geq R_3. \end{cases} \quad (9)$$

Необходимо отметить, что выполнение условий (9) является необходимым, но не достаточным условием уменьшения аппаратных затрат при применении данной методики. Для экономии ресурсов микросхемы системы функций (3), (4) после декомпозиции должны быть реализованы на меньшем количестве LUT-элементов.

Разработка VHDL-моделей для исследования структур устройств управления

При реализации УУ в базе современных программируемых логических устройств типа FPGA в качестве параметров схем могут выступать количество используемых LUT-элементов, триггерных элементов, блоков ввода/вывода, встроенных блоков памяти, линий глобальной синхронизации, а также временные характеристики максимальной длительности такта работы устройства, длительности предустановки информационных сигналов элементов памяти и формирования внешних сигналов. Перечисленные параметры можно получить в процессе синтеза УУ в одной из САПР цифровых устройств. В качестве модели УУ используется его описание на одном из поддерживаемых САПР языков описания аппаратуры. Наиболее распространенными такими языками являются VHDL [13] и Verilog [14].

В исследованиях использовалась САПР Xilinx ISE WebPack (v12.4). Основными факторами в пользу такого выбора было наличие бесплатной версии САПР, которая поддерживает проектирование небольших цифровых устройств, а также поддержка скриптового языка программирования TCL, что позволяет автоматизировать процесс синтеза. Для описания УУ использовался язык VHDL, поскольку он имеет более привычный с точки зрения классического программирования синтаксис.

Для выполнения исследований для диссертационной работы был создан программный комплекс (САПР) для синтеза УУ, который по описанию алгоритма управления создает VHDL-модель управляющего устройства, интерпретирующего указанный алгоритм.

Результатом синтеза логической схемы КМУУ в базе современных микросхем типа FPGA при помощи САПР Xilinx является файл отчета, который содержит информацию о количестве необходимых ресурсов микросхемы и временных параметрах синтезированной схемы УУ.

Информация о количестве доступных ресурсов микросхемы позволяет сделать вывод о возможности реализации схемы в данном элементном базисе, а сравнение временных параметров с требованиями к минимально необходимому быстродействию устройства еще и о целесообразности такой реализации.

В ходе исследований для предложенной структуры КМУУ были получены результаты, приведенные на рис. 3.

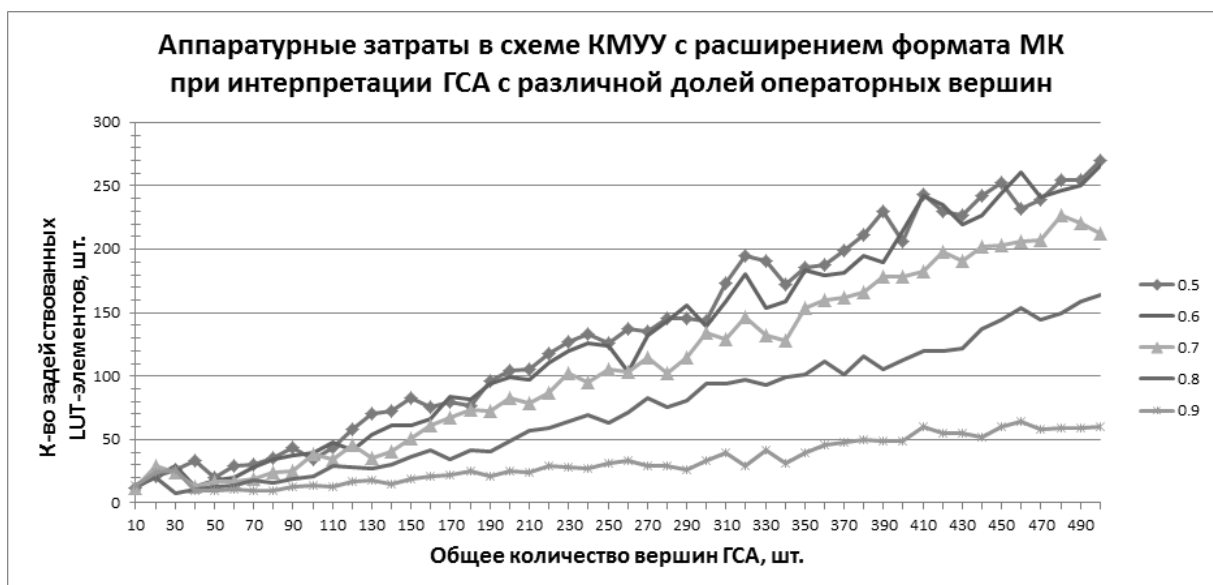


Рисунок 3. Исследование зависимости аппаратных затрат от характеристик интерпретируемого алгоритма.

Анализ результатов (рис. 3) позволяет сделать вывод о том, что использование расширения формата микрокоманд позволяет уменьшить аппаратные затраты в схеме КМУУ в среднем на 26%, при этом быстродействие устройства увеличивается за счет уменьшения количества уровней в схеме.

Заключение

Предложена новая структура КМУУ с модификацией системы адресации микрокоманд, аппаратные затраты в логической схеме которой уменьшаются в среднем на 26% в сравнении со схемой базовой структуры КМУУ.

Список литературы:

- 1.** Баркалов А.А. Синтез микропрограммных устройств управления / А.А. Баркалов, А.В. Палагин. – Киев: Институт кибернетики им. В.М. Глушкова НАН Украины, 1997. – 135 с.
- 2.** Новиков Г.И. Оценка эффективности параллельной выборки микрокоманд/ Г.И. Новиков // Известия вузов СССР. Приборостроение. – 1974. – № 2. – С. 71-73.
- 3.** Баранов С.И., Баркалов А.А. Микропрограммирование: принципы, методы применения // Зарубежная радиоэлектроника. – 1984. – №5. – С. 3-29.
- 4.** Баркалов А.А. Синтез устройств управления на программируемых логических устройствах. – Донецк: ДонНТУ, 2002 – 262 с.
- 5.** Barkalov A., Titarenko L. Logic Synthesis for Compositional Microprogram Control Units. – Berlin: Springer, 2008. – 272 pp.
- 6.** Баркалов А.А., Красичков А.А., Халед Баракат. Оптимизация композиционного микропрограммного устройства управления с элементарными операторными линейными цепями // Радиоэлектроника и информатика. – 2006. – №2. – С. 50-54.
- 7.** Barkalov A. Optimization of Control Memory Size of Control Unit with Codes Sharing / A. Barkalov,

M. Kołopieńczyk, L. Titarenko // Proceedings of the International Conference on Mixed Design of Integrated Circuits and Systems, MIXDES 2006, art. no. 1706598, pp. 354-358. **8.** *Barkalov A.* Design of CMCU with EOLC and encoding of collections of microoperations / A. Barkalov, M. Kołopieńczyk, L. Titarenko // Proceedings of the 14th International Conference "Mixed Design of Integrated Circuits and Systems", MIXDES 2007, art. no. 4286163, pp 262-265. **9.** *Barkalov A.* Synthesis of compositional control unit with modified operational linear chains / A.A. Barkalov, L. Titarenko, J. Bieganski // Experience of Designing and Application of CAD Systems in Microelectronics – Proceedings of the 10th International Conference, CADSM 2009, art. no. 4839800, pp. 182-185. **10.** Spartan-3A FPGA Family: Data Sheet. www.xilinx.com/support/documentation/data_sheets/ds529.pdf **11.** Virtex-6 Family Overview. www.xilinx.com/support/documentation/data_sheets/ds150.pdf **12.** Stratix V FPGA Family Overview. Электронный ресурс. Режим доступа: altera.com/products/devices/stratix-fpgas/stratix-v/overview/stxv-overview.html **13.** VHDL – язык описания аппаратуры. Электронный ресурс. Режим доступа: <http://allhdl.ru/vhdl.php> (Загл. с экрана). **14.** Verilog – язык проектирования аппаратуры. Электронный ресурс. Режим доступа: <http://allhdl.ru/verilog.php> (Загл. с экрана).