

УДК 681.324

УМЕНЬШЕНИЕ АППАРАТУРНЫХ ЗАТРАТ В УСТРОЙСТВЕ УПРАВЛЕНИЯ С РАЗДЕЛЕНИЕМ КОДОВ

Лаврик А.С.

Донецкий национальный технический университет

Введение

Устройства управления (УУ) являются важной частью всех цифровых систем [1]. В настоящее время для реализации схем устройств управления широко используются программируемые логические интегральные схемы (ПЛИС) типа CPLD (Complex Programmable Logic Device), включающие в себя макроячейки программируемой матричной логики (ПМЛ), имеющие большой коэффициент объединения по входу [2, 3]. Высокая стоимость этого базиса требует решения актуальной задачи уменьшения аппаратных затрат в схеме. При её решении должны учитываться как особенности элементного базиса, так и специфика интерпретируемого алгоритма управления. Если алгоритм управления представлен в виде линейной граф-схемы алгоритма (ГСА), целесообразно использовать модель композиционного микропрограммного устройства управления (КМУУ) с разделением кодов [4]. В этом случае, благодаря большому коэффициенту объединения по входу макроячеек ПМЛ, коды классов псевдоэквивалентных операторных линейных цепей (ОЛЦ) могут быть представлены более чем одним источником. В данной работе предлагается метод уменьшения аппаратных затрат в схеме КМУУ основанный на использовании двух источников кодов.

Особенности КМУУ с разделением кодов

Пусть ГСА G представлена множествами вершин B и соединяющих их дуг E . Пусть $B = \{b_0, b_E\} \cup E_1 \cup E_2$, где b_0

– начальная вершина, b_E – конечная вершина, E_1 – множество операторных вершин, где $|E_1| = M$, E_2 – множество условных вершин. Вершина $b_q \in E_1$ содержит набор микроопераций (МО) $Y(b_q) \subseteq Y$, где $Y = \{y_1, \dots, y_N\}$ – множество микроопераций [1]. В вершинах $b_q \in E_2$, записаны элементы множества логических условий $X = \{x_1, \dots, x_L\}$.

ОЛЦ α_g – последовательность операторных вершин; каждой паре её соседних вершин соответствует одна из дуг ГСА. Сформируем множество ОЛЦ $C = \{\alpha_1, \dots, \alpha_G\}$ ГСА Γ . Каждая ОЛЦ имеет лишь один выход O_g и произвольное количество входов. Назовём ГСА Γ линейной ГСА (ЛГСА), если выполняется следующее условие:

$$\frac{M}{G} \geq 2. \quad (1)$$

Каждая вершина $b_q \in E_1$ соответствует микрооперации MI_q , хранимой в управляющей памяти (УП) КМУУ по адресу A_q . Для адресации микрокоманд достаточно

$$R = \lceil \log_2 M \rceil. \quad (2)$$

бит. Пусть ОЛЦ $\alpha_g \in C$ включает F_g компонент и пусть $Q = \max(F_1, \dots, F_G)$. Закодируем ОЛЦ $\alpha_g \in C$ двоичными кодами $K(\alpha_g)$ разрядности

$$R_G = \lceil \log_2 G \rceil. \quad (3)$$

Закодируем каждую компоненту ОЛЦ кодами $K(b_q)$ разрядности

$$R_Q = \lceil \log_2 Q \rceil. \quad (4)$$

Пусть компоненты закодированы таким образом, что

$$K(b_{g_{i+1}}) = K(b_{g_i}) + 1, \quad (5)$$

где $i = 1, \dots, F_g - 1$, $g = 1, \dots, G$. Пусть для кодирования компонент ОЛЦ используются переменные $T_r \in T$, а для кодирования ОЛЦ – $\tau_r \in \tau$, где $|\tau| = R_G$, $|T| = R_Q$. В случае выполнения условия

$$R_G + R_Q = R \quad (6)$$

ЛГСА Γ может быть представлено моделью U_1 КМУУ с разделением кодов (рис. 1).

В КМУУ U_1 схема адресации микрокоманд (САМ) реализует систему функций возбуждения счетчика СТ и регистра Рг

$$\begin{aligned} \Phi &= \Phi(\tau, X), \\ \Psi &= \Psi(\tau, X). \end{aligned} \quad (7)$$

При этом адрес микрокоманды MI_q представляется в виде

$$A(b_q) = K(\alpha_g) * K(b_q), \quad (8)$$

где вершина b_q входит в ОЛЦ $\alpha_g \in C$, * – знак операции конкатенации.

По сигналу Start в Рг и СТ заносится начальный адрес микропрограммы, а триггер выборки ТВ устанавливается в

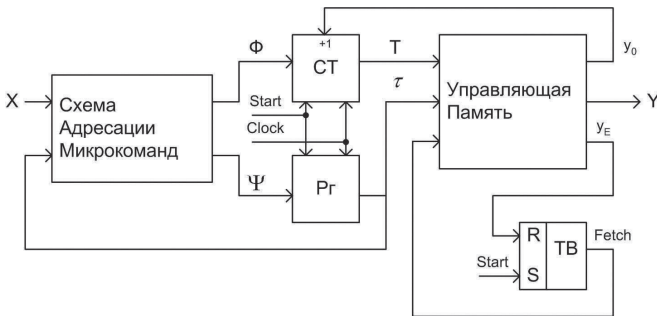


Рисунок 1 – Структурная схема КМУУ U_1

единичное состояние. При этом сигнал $\text{Fetch}=1$, что разрешает выборку микрокоманд из УП. Если считанная микрокоманда не соответствует выходу ОЛЦ, то одновременно с микрооперациями $Y(b_q)$ формируется сигнал y_0 . Если $y_0 = 1$, то к содержимому СТ прибавляется единица и адресуется следующая компонента текущей ОЛЦ. Если выход ОЛЦ достигнут, то $y_0 = 0$. При этом адрес входа следующей ОЛЦ формируется схемой адресации микрокоманд (САМ). При достижении окончания микропрограммы формируется сигнал y_E , триггер ТВ обнуляется и выборка микрокоманд прекращается.

ОЛЦ $\alpha_i, \alpha_j \in C$ являются псевдоэквивалентными ОЛЦ (ПОЛЦ) [4], если их выходы соединены с одной и той же вершиной ГСА Г.

Аппаратурные затраты в логической схеме САМ могут быть уменьшены введением специального блока преобразователя кодов (ПК), преобразующего коды ОЛЦ в коды классов псевдоэквивалентных ОЛЦ [4]. Однако этот блок занимает некоторые ресурсы микросхемы.

В данной статье предлагается использование свободных ячеек УП для представления кодов классов ПОЛЦ (первый источник), а также использование регистра R_g в качестве второго источника кодов ПОЛЦ.

Основная идея предлагаемого метода

Пусть $C_1 \subseteq C$ будет множеством ОЛЦ, где $\alpha_g \in C_1$, если её выход не соединён с конечной вершиной b_E . Найдём разбиение $\Pi_{C_1} = \{B_1, \dots, B_I\}$ множества C_1 на классы ПОЛЦ. Закодируем ОЛЦ $\alpha_g \in C_1$ таким образом, чтобы большинство классов $B_i \in \Pi_{C_1}$ было представлено одним обобщённым интервалом R_G -мерного булева пространства. Используем для этого широко известный алгоритм ESPRESSO [5]. Пусть $\Pi_{C_1} = \Pi_A \cup \Pi_B$, где $B_i \in \Pi_A$, если блок представлен одним интервалом, иначе $B_i \in \Pi_B$.

Если выполняется условие

$$\Pi_B = \emptyset, \quad (9)$$

то регистр R_g является источником всех кодов $K(B_i)$ для классов $B_i \in \Pi_C$. В этом случае САМ реализует H_0 термов, где H_0 – число переходов эквивалентного автомата Мили [4]. Это минимально возможное число переходов.

Отметим, что схемы блоков САМ, СТ, R_g и ТВ реализуются на макроячейках ПМЛ. Для реализации УП, требуются внешние ППЗУ имеющие t выходов, где $t \in \{1, 2, 4, 8, 16\}$ [2, 3]. При использовании унитарного кодирования микроопераций [6], каждое слово в УП состоит из $N + 2$ бит. Число 2 добавляется для учёта дополнительных переменных y_0 и y_E .

Если каждое ППЗУ имеет не менее M слов, то достаточно K_0 микросхем для реализации УП, где

$$K_0 = \left\lceil \frac{N + 2}{t} \right\rceil. \quad (10)$$

Очевидно, что остаётся R_0 свободных выходов микросхем ППЗУ, где

$$R_0 = K_0 t - N - 2. \quad (11)$$

Эти выходы могут быть использованы для кодирования классов $B_i \in \Pi_B$, где

$$R_B = \left\lceil \log_2 (|\Pi_B| + 1) \right\rceil. \quad (12)$$

Единица добавляется к $|\Pi_B|$ для обозначения ситуации $B_i \in \Pi_B$. Если условие

$$R_0 \geq R_B \quad (13)$$

выполняется, то все классы $B_i \in \Pi_B$ могут быть представлены с помощью УП. Иначе может быть представлено лишь I_{CM} классов:

$$I_{CM} = 2^{R_0}. \quad (14)$$

Оставшиеся классы $B_i \in \Pi_B$ должны быть помещены в Π_A и представлены Pг. В обоих случаях, для интерпретации ЛГСА Γ , предлагается КМУУ U_2 (рис. 2).

В КМУУ U_2 , блок САМ реализует следующие функции:

$$\begin{aligned} \Phi &= \Phi(\tau, Z, X), \\ \Psi &= \Psi(\tau, Z, X), \end{aligned} \quad (15)$$

где переменные $z_r \in Z$ используются для кодирования классов $B_i \in \Pi_B$, $|Z| = R_0$. В зависимости от переменных τ и T управляющая память реализует функции Z , Y , y_0 и y_E . Принцип действия КМУУ U_2 практически идентичен U_1 .

В работе предлагается следующий алгоритм построения КМУУ U_2 :

1. Построение множеств C , C_1 и Π_C для ЛГСА Γ .
2. Кодирование ОЛЦ $\alpha_g \in C$ и их компонент.
3. Построение разбиений Π_A и $|\Pi_B|$.
4. Кодирование классов $B_i \in \Pi_B$.
5. Построение таблицы переходов КМУУ.
6. Спецификация управляющей памяти.
7. Реализация схемы КМУУ в данном элементном базисе.

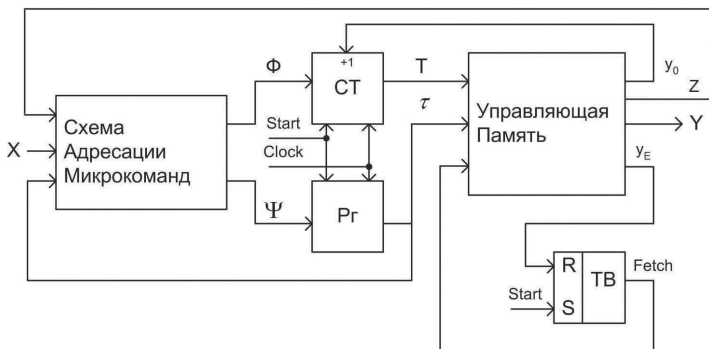


Рисунок 2. Структурная схема КМУУ U_2

Заключение

Предложенный метод направлен на уменьшение числа макроячеек ПМЛ в схеме адресации микрокоманд КМУУ с разделением кодов. Это возможно благодаря таким главным факторам как большой коэффициент объединения по входу современных макроячеек ПМЛ; естественная избыточность микросхем ППЗУ, обусловленная тем, что количество их выходов ограничено определённым множеством чисел; существование классов псевдоэквивалентных ОЛЦ.

Исследования показали, что количество макроячеек уменьшается до 12% для КМУУ $U_2(\Gamma_1)$ в сравнении с эквивалентным КМУУ $U_1(\Gamma_1)$. Стоит отметить, что данный метод применим лишь для интерпретации линейных ГСА при условии выполнения (6).

Дальнейшие исследования в данном направлении связаны с применением предложенного подхода для реализации КМУУ на микросхемах FPGA [5].

Литература

- [1] Baranov S. Logic Synthesis for Control Automata. – Kluwer Academic Publishers, 1994. – 312 pp.
- [2] Электронный ресурс. Altera devices overview. <http://www.altera.com>.
- [3] Электронный ресурс. Xilinx CPLDs <http://www.xilinx.com>.
- [4] Barkalov A., Titarenko L. Logic Synthesis for Compositional Microprogram Control Units. – Berlin: Springer, 2008. – 272 pp.
- [5] Maxfield C. The Design Warrior's Guide to FPGAs. – Amsterdam: Elseveir, 2004. – 541 pp.