

УДК 004.312

ОПТИМИЗАЦИЯ ПЛОЩАДИ КРИСТАЛЛА ПРИ РЕАЛИЗАЦИИ УСТРОЙСТВА УПРАВЛЕНИЯ НА CPLD

*А.В. Хомаха¹, А.С. Лаврик¹, А.А. Баркалов²,
¹ДонНТУ, г.Донецк, ²Зеленогурский университет, Польша.*

Введение

Известно, что одной из важнейших частей любой цифровой системы является устройство управления (УУ) [1]. В случае представления алгоритма управления в виде линейной граф-схемы алгоритма (ГСА), можно использовать модель композиционного микропрограммного устройства управления (КМУУ) с разделением кодов [2]. В настоящее время для реализации логических схем устройств управления широко используются программируемые логические интегральные схемы (ПЛИС) типа CPLD (Complex Programmable Logic Devices) [4]. Высокая стоимость этого базиса продуцирует актуальную задачу сокращения аппаратурных затрат в схеме. При её решении следует учитывать не только специфику интерпретируемого алгоритма управления, но и особенности элементного базиса. В таком 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 аппаратурных затрат в схеме КМУУ, основанный на использовании двух источников кодов.

Основные положения

Пусть ГСА Г представлена КМУУ с разделением кодов U_1 (Рис. 1).

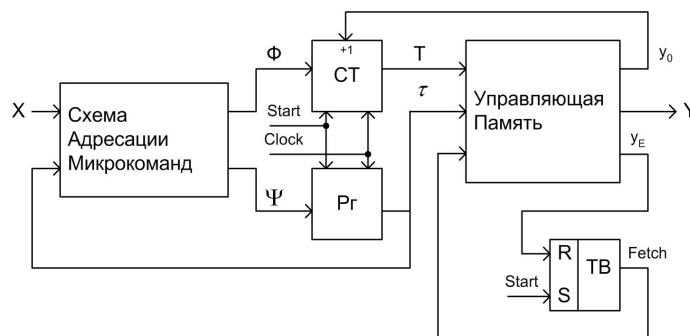


Рисунок 1 – Структурная схема КМУУ U_1

В КМУУ U_1 , схема адресации микрокоманд (САМ) реализует систему функций возбуждения счётчика СТ и регистра Рг. КМУУ U_1 работает следующим образом. По сигналу $Start = 1$ начальный адрес (все нули) заносится в Рг и СТ. В тоже время, триггер выборки ТВ устанавливается в единичное состояние. При этом $Fetch = 1$, что разрешает выборку микрокоманд из УП. Каждая ячейка УП содержит микрооперации $y_n \in Y$ и специальные переменные y_0 и y_E . Если $y_0 = 1$, то к текущему содержимому СТ прибавляется единица и адресуется следующий компонент текущей ОЛЦ. Иначе –

значения СТ и Rг загружаются из САМ. Первый случай соответствует переходу из любого компонента ОЛЦ кроме выхода. Второй случай соответствует переходу из выхода ОЛЦ. Если $y_E = 1$, триггер выборки обнуляется, сигнал $Fetch = 0$ и работа КМУУ прекращается. Это соответствует переходу из конечной вершины. Отметим, что ОЛЦ $\alpha_i, \alpha_j \in C$ являются псевдоэквивалентными ОЛЦ [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 потребляет некоторые ресурсы микросхемы.

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

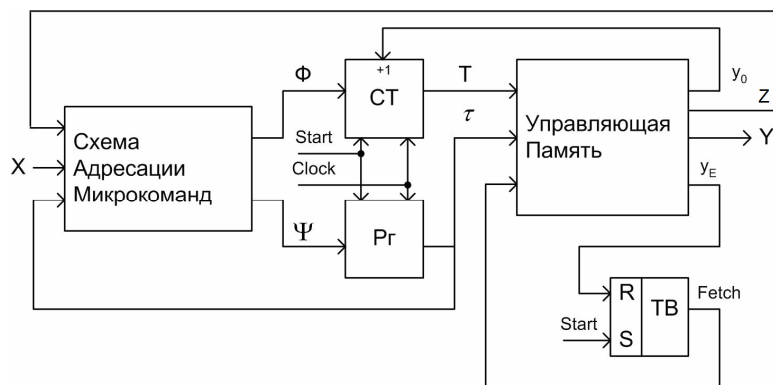
Основная идея предлагаемого метода

Пусть $C_1 \subseteq C$ будет множеством ОЛЦ, где $\alpha_g \in C_1$, если её выход не соединён с конечной вершиной b_E . Пусть $\Pi_C = \{B_1, \dots, B_I\}$ - разбиение множества C_1 на классы ПОЛЦ. Закодируем ОЛЦ $\alpha_g \in C_1$ таким образом, чтобы большинство классов $B_i \in \Pi_C$ было представлено одним интервалом R_C -мерного булева пространства. Для этого может быть использован широко известный алгоритм ESPRESSO [3]. Пусть $\Pi_C = \Pi_A \cup \Pi_B$, где $B_i \in \Pi_A$, если класс представлен одним интервалом, иначе $B_i \in \Pi_B$.

Если выполняется условие $\Pi_B = \emptyset$, то регистр Rг 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, что является минимально возможным числом переходов [4].

Отметим, что схемы блоков САМ, СТ, Rг 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 [4], каждое слово в УП состоит из $N+2$ бит. Число 2 добавляется к N для учёта дополнительных переменных y_0 и y_E (Рис. 1).

Если каждое ППЗУ имеет t выходов и не менее M слов, то достаточно K_0 микросхем для реализации УП, где $K_0 = \left\lceil \frac{N+2}{t} \right\rceil$. Очевидно, что остаётся R_0 свободных выходов микросхем ППЗУ, где $R_0 = K_0 t - N - 2$. Эти выходы могут быть использованы для кодирования классов $B_i \in \Pi_B$, где $R_B = \lceil \log_2(|\Pi_B| + 1) \rceil$. Единица добавляется к $|\Pi_B|$ для обозначения ситуации $B_i \notin \Pi_B$. Если выполняется условие $R_0 \geq R_B$, то все классы $B_i \in \Pi_B$ могут быть представлены с помощью УП. Иначе может быть представлено лишь I_{CM} классов, где $I_{CM} = 2^{R_0}$. Оставшиеся классы $B_i \in \Pi_B$ должны быть помещены в Π_A и представлены Rг. В обоих случаях, для интерпретации ЛГСА Г предлагается КМУУ U_2 (Рис. 2).

Рисунок 2 – Структурная схема КМУУ U_2

Принципы действия КМУУ U_1 и U_2 практически идентичны.

В работе предлагается следующий алгоритм построения КМУУ U_2 :

1. Построение множеств C, C_1 и P_C для ЛГСА Γ .
2. Кодирование ОЛЦ $\alpha_g \in C$ и их компонентов.
3. Построение разбиений Π_A и Π_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.

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, а также 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.

Дальнейшие направления наших исследований связаны с применением предложенного подхода для реализации КМУУ на микросхемах FPGA [3].

Литература

- [1] Baranov S. Logic Synthesis for Control Automata. – Kluwer Academic Publishers, 1994. – 312 pp.
- [2] Barkalov A., Titarenko L. Logic Synthesis for Compositional Microprogram Control Units. – Berlin: Springer, 2008. – 272 pp.
- [3] Maxfield C. The Design Warrior's Guide to FPGAs. – Amsterdam: Elsevier, 2004. – 541 pp.
- [4] Соловьев В.В. Проектирование цифровых схем на основе программируемых логических интегральных схем. – М.: Горячая линия-ТЕЛЕКОМ, 2001. – 636 с.