

ОПТИМИЗАЦИЯ АВТОМАТА МИЛИ С КОДИРОВАНИЕМ НАБОРОВ МИКРООПЕРАЦИЙ

Баркалов А.А., Ахмад Фуад Хамада Бадер, Бабаков Р.М.
Донецкий национальный технический университет, г.Донецк
факультет вычислительной техники и информатики
E-mail: babakov@cs.dgtu.donetsk.ua

Abstract

Barkalov A.A., Achmad Fuad Kchamada Bader, Babakov R.M. Optimization of the Miles automaton with the microoperation sets encoding. The method of optimization of hardware expenses in the logic circuit of the Mealy automaton with coding of sets of microoperations is offered and researched at usage of heterogeneous element base of programmable integrated circuits. The method is based on encoding of sets into fragment of microoperations. Analytical dependences of efficiency of a method are obtained depending on parameters of the realizable flow-chart of control algorithm.

Введение

Широкое применение средств цифровой автоматике и вычислительной техники ставит актуальную задачу усовершенствования характеристик современных вычислительных систем. Одним из возможных путей решения этой задачи является усовершенствования характеристик устройства управления, которые во многом определяют характеристики системы в целом [1, 2]. Важнейшей характеристикой устройства управления являются аппаратные затраты в схеме устройства, определяющие в конечном счете стоимость системы [3].

В настоящей работе предлагается метод оптимизации аппаратных затрат в логической схеме управляющего автомата Мили с кодированием наборов микроопераций, являющегося одним из типов устройств управления, при реализации схемы устройства в гетерогенном базисе программируемых логических устройств.

1. Организация РУ-автомата Мили

Метод кодирования наборов микроопераций является одним из методов оптимизации аппаратных затрат в логической схеме управляющего автомата Мили [4, 5]. Следствием его использования является гетерогенная структура автомата, включающая программируемые логические устройства (ПЛУ) как элементный базис схемы формирования адреса и ПЗУ как базис реализации схемы формирования микроопераций.

При использовании данного метода каждому набору микроопераций $Y_t \subseteq Y$ ставится в соответствие двоичный код разрядности $Q = \lceil \log_2 T \rceil$, где T — число различных наборов. Для кодирования используются переменные из множества $Z = \{z_1, \dots, z_Q\}$, причем Y -подсхема реализует преобразование $Z \rightarrow Y$. Автомат с кодированием наборов микроопераций (рис. 1) называется РУ-автоматом.

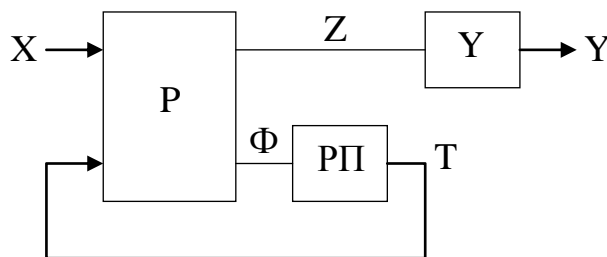


Рисунок 1 — Структурная схема автомата Мили с кодированием наборов микроопераций

Здесь Y-подсхема формирует функции

$$Y = Y(Z), \tag{1}$$

а P-подсхема строится по преобразованной ПСТ и реализует систему функций

$$\begin{aligned} Z &= Z(X, T), \\ \Phi &= \Phi(X, T). \end{aligned} \tag{2}$$

Основные этапы и пример синтеза PУ-автомата рассмотрены в [5].

2. Структурная организация PУ_F-автомата

Оптимизируем схему формирования микроопераций PУ-автомата с целью уменьшения аппаратных затрат. Для этого выполним следующее. Выделим из множества микроопераций Y часть, образующую так называемый *фрагмент микроопераций* (ФМО), которая будет формироваться Y-подсхемой аналогично PУ-автомату, часть — непосредственно P-подсхемой устройства (рис. 2). Назовем структуру, изображенную на рис. 2, автоматом с кодированием фрагмента микроопераций, или PУ_F-автоматом, а метод синтеза структуры — методом кодированием наборов фрагмента микроопераций.

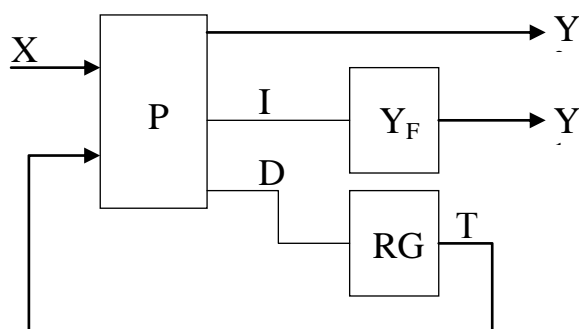


Рисунок 2 — Структурная схема PУ_F-автомата

Здесь Y_F-подсхема формирует подмножество микроопераций Y¹ ∈ Y, входящих во фрагмент микроопераций. P-подсхема формирует код I набора MO, входящих в ФМО, а также подмножество микроопераций Y², формируемых отдельно. В целом PУ_F-автомат реализует следующие системы функций:

$$D = D(T, X), \tag{3}$$

$$I = I(T, X), \tag{4}$$

$$Y^1 = Y^1(I), \tag{5}$$

$$Y^2 = Y^2(T, X), \tag{6}$$

причем Y¹ ∪ Y² = Y, Y¹ ∩ Y² = ∅. Отметим, что количество и кодирование наборов микроопераций в PУ- и PУ_F-автоматах в общем случае различны. В остальном PУ_F-автомат функционирует аналогично PУ-автомату.

3. Аналитическое определение эффективности PУ_F-автомата

Определим аналитические характеристики логической схемы PУ_F-автомата.

Пусть в структуре на рис. 1 наборы микроопераций закодированы двоичным кодом разрядности R₀, и подсхема Y формирует N микроопераций, подразделяющихся на W наборов.

Пусть также в структуре на рис. 2 мощность множества Y¹ составляет N₁ микроопераций, множества Y² — R₂ микроопераций. Разрядность кода набора ФМО I обозначим через R₁. Таким образом, общее число выходов P-подсхемы PУ_F-автомата равняется R₁+R₂. Также обозначим число наборов микроопераций во фрагменте через W₁.

Дополнительно примем, что в обеих структурах P-подсхема анализирует L логических условий и R разрядов кода состояния автомата.

Определим выраженные в битах информационные емкости логических схем PУ- и PУ_F-автоматов. Условно считая, что P-подсхема реализуется на ПЛМ, используем матричное представление P-подсхемы для определения ее информационной емкости [4, 5]:

$$V_{ПЛМ} = V_{„И”} + V_{„ИЛИ”}. \quad (7)$$

С учетом (7), емкость Р-подсхемы РY-автомата определяется как

$$V_P^{PY} = 2 \cdot (L + R) \cdot Q_{PY} + Q_{PY} \cdot (R + R_0), \quad (8)$$

где Q_{PY} — число внутренних термов ПЛУ Р-подсхемы.

Аналогично для РY_F-автомата

$$V_P^{PY_F} = 2 \cdot (L + R) \cdot Q_{PY_F} + Q_{PY_F} \cdot (R + R_1 + R_2), \quad (9)$$

где Q_{PY_F} — число внутренних термов Р-подсхемы РY_F-автомата. Отметим, что в общем случае $Q \neq Q_1$, поскольку системы функций, формируемые Р-подсхемами, различны.

Информационная емкость ПЗУ Y-подсхемы РY-автомата составляет

$$V_Y = N \cdot 2^{R_0} = N \cdot 2^{\lceil \log_2 W \rceil} \quad (10)$$

бит, а емкость ПЗУ Y_F-подсхемы РY_F-автомата равна

$$V_{Y_F} = N_1 \cdot 2^{R_1} = N_1 \cdot 2^{\lceil \log_2 W_1 \rceil}. \quad (11)$$

Поскольку в общем случае базис ПЗУ дешевле базиса программируемой логики, приведем единицу емкости ПЛУ в стоимостное соответствие с единицей емкости ПЗУ с помощью специального коэффициента К. Тогда полная информационная емкость логической схемы РY-автомата определяется как

$$\begin{aligned} V^{PY} &= K \cdot V_P^{PY} + V_Y = \\ &= K \cdot (2 \cdot (L + R) \cdot Q_{PY} + Q_{PY} \cdot (R + R_0)) + N \cdot 2^{R_0}, \end{aligned} \quad (12)$$

а в случае РY_F-автомата этот параметр определяется как

$$\begin{aligned} V^{PY_F} &= K \cdot V_P^{PY_F} + V_{Y_F} = \\ &= K \cdot (2 \cdot (L + R) \cdot Q_{PY_F} + Q_{PY_F} \cdot (R + R_1 + R_2)) + N_1 \cdot 2^{R_1}. \end{aligned} \quad (13)$$

Определим эффективность $E(PY_F)$ использования РY_F-автомата как отношение функций (12) и (13):

$$E(PY_F) = \frac{K \cdot (2 \cdot (L + R) \cdot Q_{PY} + Q_{PY} \cdot (R + R_0)) + N \cdot 2^{R_0}}{K \cdot (2 \cdot (L + R) \cdot Q_{PY_F} + Q_{PY_F} \cdot (R + R_1 + R_2)) + N_1 \cdot 2^{R_1}}. \quad (14)$$

Таким образом, критерием целесообразности применения РY_F-автомата по сравнению с РY-автоматом будем считать значение эффективности $E(PY_F) > 1$. Отметим, что значения аргументов формулы (14) могут быть получены лишь в результате полного синтеза обеих структур автоматов, а сама формула (14) справедлива лишь при реализации Р-подсхем в базисе ПЛМ.

Примем следующее допущение. Если число выходов ПЛУ Р-подсхемы не изменилось при переходе от РY- к РY_F-автомату, то не изменилась и информационная емкость ПЛУ (число входов ПЛУ в обеих структурах является одинаковым). Тогда в качестве эффективности РY_F-автомата по отношению к РY-автомату будем рассматривать эффективность по ПЗУ $E_{ROM}(PY_F)$, равную отношению информационной емкости Y-подсхемы к емкости Y_F-подсхемы при неизменном числе выходов Р-подсхемы. С учетом (10) и (11) имеем:

$$E_{ROM}(PY_F) = \frac{N \cdot 2^{R_0}}{N_1 \cdot 2^{R_1}} = \frac{N \cdot 2^{\lceil \log_2 W \rceil}}{N_1 \cdot 2^{\lceil \log_2 W_1 \rceil}}. \quad (15)$$

Полагая, что выигрыш имеет место при $E_{ROM}(PY_F) > 1$, получаем следующую систему, определяющую целесообразность применения РY_F-автомата:

$$\begin{cases} R_0 = R_1 + R_2 & (16) \end{cases}$$

$$\begin{cases} N \cdot 2^{\lceil \log_2 W \rceil} > N_1 \cdot 2^{\lceil \log_2 W_1 \rceil} & (17) \end{cases}$$

Здесь (16) указывает на равенство числа выходов Р-подсхемы РY_F-автомата по сравнению с РY-автоматом; неравенство (17) вытекает из формулы (15) для значения E_{ROM}(PY_F)>1.

Данную систему можно записать следующим образом:

$$\begin{cases} R_0 = R_1 + N - N_1 & (18) \\ N \cdot 2^{R_0} > N_1 \cdot 2^{R_1} & (19) \end{cases}$$

Очевидно, что при W₁=2 для кодирования наборов требуется R₁=⌊log₂W₁⌋=1 переменных. Следовательно, минимальное число наборов во фрагменте W_{1(min)}=2 и R_{1(min)}=1. При W₁=1 микрооперации, формируемые автоматом, являются константами. Подставив в (18) значение R₁=R_{1(min)}=1, получим:

$$N_{1(min)} = N + 1 - R_0. \tag{20}$$

Таким образом, минимальное число микроопераций, которые должны быть включены во фрагмент, определяется формулой (20) и является ограниченной величиной. Если во фрагмент включить количество МО, меньшее N_{1(min)}, оставшиеся МО будут вынуждены будут формироваться непосредственно Р-подсхемой. Следовательно, величина R₂ в формуле (16) будет расти при сохранении значения R₁, и число выходов Р-подсхемы РY_F-автомата будет больше, чем в случае РY-автомата, что противоречит сделанному нами допущению.

4. Пример формирования фрагмента микроопераций

Пусть РY-автомат формирует N=8 микроопераций, образующих W=9 наборов Y₁–Y₉ (таблица 1), что приводит к емкости ПЗУ, равной 8*2⁴=128 бит.

Таблица 1 — Содержимое ПЗУ Y-подсхемы

Y _i	R ₀	y ₁	y ₂	y ₃	y ₄	y ₅	y ₆	y ₇	y ₈
Y ₁	0 0 0 0	0	0	1	0	0	1	0	0
Y ₂	0 0 0 1	1	0	0	0	1	0	0	1
Y ₃	0 0 1 0	0	0	1	0	0	1	1	0
Y ₄	0 0 1 1	1	0	0	1	1	0	0	1
Y ₅	0 1 0 0	0	0	1	1	0	1	0	0
Y ₆	0 1 0 1	1	0	0	0	1	0	1	1
Y ₇	0 1 1 0	1	1	0	0	1	0	0	1
Y ₈	0 1 1 1	0	1	1	1	0	1	1	0
Y ₉	1 0 0 0	1	0	0	1	1	0	1	1

Согласно (20), минимальное количество микроопераций во фрагменте должно быть равно N_{1(min)} = (8+1-⌊log₂9⌋) = 5, причем для микроопераций подмножества Y¹ должно быть определено два набора (W₁=2, R₁=1). Если величина W₁=2 достижима, то Р-подсхема будет формировать R₁=1 разрядов кода набора фрагмента и R₂=N-N₁=8-5=3 унитарных микроопераций из подмножества Y².

Не останавливаясь на способе формирования, включим во фрагмент микрооперации y₁, y₃, y₅, y₆, y₈. Рассматривая в табл. 1 лишь эти операции, можно выделить W₁=2 набора: Y₁={y₃, y₆} и Y₂={y₁, y₅, y₈}, для кодирования которых достаточно R₁=1 разряд. Таблица Y_F-подсхемы для нашего примера будет иметь следующий вид:

Таблица 2 — Содержимое ПЗУ Y_F-подсхемы

R ₁	y ₁	y ₃	y ₅	y ₆	y ₈
0	0	1	0	1	0
1	1	0	1	0	1

Микрооперации y₂, y₄, y₇ образуют множество Y² и формируются непосредственно Р-подсхемой.

Поскольку из N₁=5 микроопераций удалось сформировать W₁=2 набора, число выходов Р-подсхемы Y_F-автомата не изменилось по сравнению с РY-автоматом, а емкость ПЗУ Y_F-подсхемы стала равной 5*2¹=10 бит. Очевидно, что для нашего примера емкость

ПЗУ Y_F -подсхемы в 16 раз меньше по сравнению с емкостью Y -подсхемы эквивалентного РУ-автомата, и эффективность по ПЗУ $E_{ROM}=16$.

Предположим, что при формировании фрагмента микроопераций мы не включили туда МО y_8 (таблица 2 без последнего столбца). В результате во фрагменте по прежнему будет два набора ($Y_1=\{y_3, y_6\}$ и $Y_2=\{y_1, y_5\}$), но МО y_8 будет формироваться Р-подсхемой. В результате число выходов Р-подсхемы будет равно 5, и ограничение (18) нарушится.

Нарушение ограничения (18) приводит к тому, что определение $E_{ROM}(PY_F)$ оказывается невозможным, поскольку число выходов Р-подсхемы изменяется при переходе к PY_F -структуре. Однако это не означает, что PY_F -автомат в данном случае будет менее эффективен, чем эквивалентный РУ-автомат. Если в результате применения метода кодирования наборов фрагмента число выходов Р-подсхемы изменилось (увеличилось или уменьшилось), точное значение эффективности следует определять по формуле (14).

5. Исследование предложенного метода

Выполним исследование эффективности PY_F -автомата по сравнению с РУ-структурой, для чего определим значения эффективности по ПЗУ с использованием формулы (15) для различных характеристик логических схем автоматов.

Определим зависимость E_{ROM} как функцию от N_1 и W_1 при заданных N и W . Примем количество микроопераций $N=100$ и число наборов $W=0.8*N=80$ [5]. При этом минимальное число микроопераций во фрагменте, согласно (20), составит $N_{1(min)} = 100+1-7 = 94$.

Условимся изменять значение W_1 в интервале $[1, W]$ с шагом $0.1W$ и значение N_1 в интервале от $N_{1(min)}$ до $N-1$ с шагом 1. В результате получим следующую таблицу (таблица 3).

Таблица 3 — Зависимость $E_{ROM}(N, W, N_1, W_1)$ при $N=100, W=80$

$N_1 \backslash W_1$	1	10	20	30	40	50	60	70	80
94	136.1	8.51	4.26	4.26	2.13	2.13	2.13	1.06	1.06
95	134.7	8.42	4.21	4.21	2.11	2.11	2.11	1.05	1.05
96	133.3	8.33	4.17	4.17	2.08	2.08	2.08	1.04	1.04
97	131.9	8.25	4.12	4.12	2.06	2.06	2.06	1.03	1.03
98	130.6	8.16	4.08	4.08	2.04	2.04	2.04	1.02	1.02
99	129.3	8.08	4.04	4.04	2.02	2.02	2.02	1.01	1.01

Отметим, что значения эффективности изменяются ступенчатым образом, что обусловлено двукратным уменьшением емкости ПЗУ при уменьшении разрядности кода набора на единицу.

Заключение

Предложенный в работе метод оптимизации автомата Мили с кодированием наборов фрагмента переменных позволяет уменьшить аппаратные затраты в схеме формирования микроопераций автомата Мили с кодированием наборов микроопераций. Данный метод не затрагивает остальные узлы логической схемы автомата и может использоваться совместно с другими методами оптимизации.

Литература

1. Глушков В.М. Синтез цифровых автоматов. — М.: Физматгиз, 1962. — 476 с.
2. Шалыто А.А. Методы аппаратно и программной реализации алгоритмов. — СПб.: Наука, 2000. — 780 с.
3. Соловьев В.В. Проектирование функциональных узлов цифровых систем на программируемых логических устройствах. — Минск: Бестпринт, 1996. — 252 с.
4. Баркалов А.А. Синтез устройств управления на программируемых логических устройствах. — Донецк: ДонНТУ, 2002. — 262 с.
5. Баркалов А.А., Палагин А.В. Синтез микропрограммных устройств управления. — Киев: Институт кибернетики НАН Украины, 1997. — 135 с.