

## КЛАССИФИКАЦИЯ МЕТОДОВ ОПТИМИЗАЦИИ УСТРОЙСТВ УПРАВЛЕНИЯ НА ПРОГРАММИРУЕМЫХ БИС

Баркалов А.А. Аль-Бахри А.М. Зеленева И.Я.  
кафедра ЭВМ, ДГТУ  
barkalov@cs.dgtu.donetsk.ua

### Abstract

*Barkalov A.A., Zelenjova I.J, Al-Bahri A.M. Classification of the optimization methods of control units on programmable VLSI. This article is devoted to the analysis of different methods of automata optimization. The variety of approaches has been discussed with analysis of their advantages and disadvantages. Tasks for a formal choice of the methods optimization set are discussed and ways to solve that problems are proposed.*

### Введение

Устройство управления (УУ) является одним из центральных блоков цифровых систем [1]. В настоящее время для синтеза логических схем устройств управления широко применяются программируемые БИС, наиболее характерными представителями которых являются программируемые логические матрицы (ПЛМ), программируемые матрицы логики (ПМЛ), программируемые постоянные запоминающие устройства (ППЗУ), программируемые матрицы вентилей (ПМВ) [2]. Одной из важных задач, возникающих при синтезе схемы УУ, является оптимизация аппаратурных затрат в ней. В настоящей работе предлагается классификация методов оптимизации аппаратурных затрат при реализации УУ в виде автомата с жесткой логикой на стандартных программируемых БИС. Применяемый метод оптимизации схемы автомата зависит от сочетаний характеристик реализуемого автомата и элементного базиса. Обычно автомат характеризуется числом логических условий  $L$ , числом выходных сигналов  $N$ , числом внутренних состояний  $M$  и требуемых для их кодировки внутренних переменных  $R$ , максимальным числом букв в термах дизъюнктивных нормальных форм логических функций, соответствующих строкам прямой структурной таблицы автомата,  $L_{\max}$  и числом строк в ней  $H$ . Элементный базис при всем его многообразии может характеризоваться числом входов  $s$ , выходов  $t$  и промежуточных шин  $q$ .

### 1. Классификация методов оптимизации и их характеристика

#### Тривиальная реализация

При выполнении условий

$$\begin{aligned} R+L &\leq s, \\ H &\leq q, \end{aligned} \quad (1)$$

$$N+R \leq t,$$

автомат реализуется на одной микросхеме.

**Одноуровневая реализация**

Если система условий (1) не выполняется, но  $L_{max} + R \leq S$  (2), то автомат может быть представлен в виде одноуровневой схемы (рис.1)

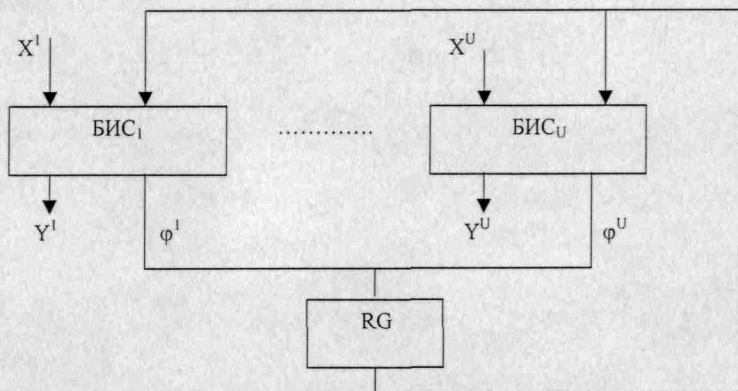


Рис.1 Одноуровневая реализация автомата

Для синтеза схемы необходимо решить задачу покрытия исходных множеств логических условий  $X = \{x_1, \dots, x_L\}$ , микроопераций  $Y = \{y_1, \dots, y_n\}$  и термов  $F = \{F_1, \dots, F_h\}$  минимально возможным числом блоков  $X^U, Y^U, F^U$  ( $U=1..U$ ) соответственно [3]. Блоки  $X^U, Y^U, F^U$  ( $U=1..U$ ) реализуются на одной БИС. Эта задача близка по своей природе к задачам компоновки [4] и решается аналогичными способами. Одноуровневая реализация автомата обладает максимальным быстродействием, но это сопровождается значительным дублированием входных и выходных переменных на полюсах БИС и, как следствие, избыточными аппаратными затратами.

**Структурная редукция**

Этот подход ориентирован на минимизацию аппаратных затрат в схеме автомата за счет введения многоуровневости [3]. Основу многоуровневых схем представляют двухуровневые схемы, в которых могут использоваться следующие методы:

- а) Замена входных переменных, при которой логические условия  $X$  заменяются элементами множества  $P = \{p_1, \dots, p_G\}$ , где  $G \ll L$  (MP-автоматы).
- б) Замена наборов микроопераций, когда каждому набору микроопераций  $Y_t \subseteq Y$  ( $t=1, \dots, T$ ) ставится в соответствие двоичный код  $K(Y_t)$  разрядности  $Q = \lceil \log_2 T \rceil$ . Это равносильно замене множества  $Y$  множеством  $Z = \{z_1, \dots, z_q\}$  (PY-автоматы). Использование метода кодирования полей совместимых микроопераций [5] порождает PD-автоматы.
- в) Кодирование строк прямой структурной таблицы, когда каждой строке  $F_h$  ставится в соответствие код  $K(F_h)$  разрядности  $Q = \lceil \log_2 H \rceil$ . Это равносильно замене множества  $F$  множеством  $Z$ , что порождает PF-автоматы.

Объединение различных методов построения двухуровневых схем порождает схемы с большим числом уровней. Например, на рис.2 приведена двухуровневая схема, представляющая собой объединение MP и PY – автоматов, то есть MPY – автомат.

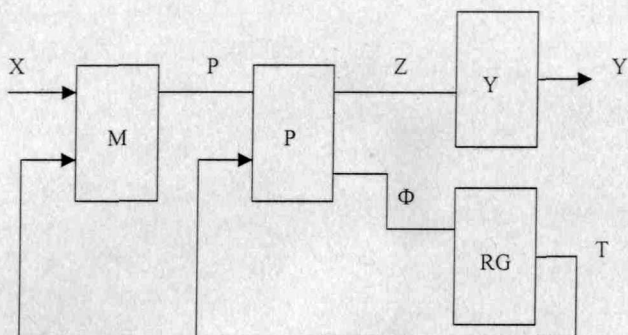


Рис.2 Структурная схема МРУ – автомата

Здесь М-подсхема реализует систему функций  $P=P(X,T)$ , заменяющих логические условия X, Р-подсхема формирует кодирующие наборы микроопераций функции  $Z=Z(P,T)$  и функции возбуждения памяти  $\Phi=\Phi(P,T)$ , Y-подсхема формирует микрооперации  $Y=Y(Z)$ .

### Нестандартное представление термов

В классических схемах автоматов [1] термы, соответствующие строкам прямой структурной таблицы (ПСТ), определяются парами <исходное состояние, входной сигнал>. При переходе к РУ- автомату это требует введения дополнительных переменных Z, что приводит к увеличению требований к числу выходов БИС Р-подсхемы. В работах [6,7] предлагается определять термы ПСТ парами <исходное состояние, состояние перехода>, что порождает PR-автоматы, имеющие два регистра памяти (рис.3).

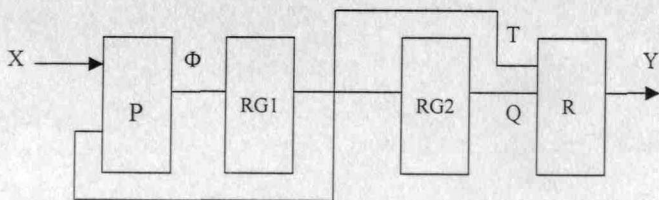


Рис.3 Структурная схема PR-автомата

Здесь R-подсхема формирует систему функций  $Y=Y(Q,T)$ , причем переменные Q кодируют исходные состояния, а переменные T – состояние перехода автомата.

### Гетерогенная реализация

Идея метода заключается в использовании различного элементного базиса для реализации различных структурных элементов схемы [3]. Например, в автомате Мура для реализации функций  $Y=Y(T)$  целесообразно использовать ПЗУ; М-подсхемы целесообразно реализовывать на мультиплексорах, Р-подсхемы на ПЛМ, У-подсхемы – на ПЗУ, D-подсхемы – на дешифраторах. Такой подход позволяет уменьшить стоимость схемы автомата по сравнению с реализацией всей схемы на ПЛМ[3].

### Алгоритмические методы

Эти методы направлены на оптимальное кодирование состояний автомата и/или промежуточных переменных в многоуровневых схемах. Преимущество этих методов состоит в исключении необходимости введения дополнительных структурных элементов в схему автомата, что приводит к увеличению времени цикла. Например, учет наличия псевдоэквивалентных состояний в автомате Мура [8] позволяет выполнить кодирование состояний так, что длина ПСТ автомата Мура уменьшается до длины ПСТ эквивалентного состояния Мили. Под псевдоэквивалентными понимаются состояния, соответствующие операторным вершинам с запараллелеными выходами [8]. Пусть множество состояний  $A=\{a_1, \dots, a_7\}$  разбито на блоки  $B_1=\{a_1\}$ ,  $B_2=\{a_2, a_3, a_4, a_5\}$ ,  $B_3=\{a_6, a_7\}$ . Если закодировать состояния следующим образом  $k(a_1)=000$ ,  $k(a_2)=100$ ,  $k(a_3)=101$ ,  $k(a_4)=110$ ,  $k(a_5)=111$ ,  $k(a_6)=001$ ,  $k(a_7)=011$ , то блоку  $B_1$  соответствует код 000, блоку  $B_2 - 1xx$ , блоку  $B_3 - 0x1$ . Теперь в ПСТ вместо переходов из состояния  $a_n \in B_i$  можно рассмотреть только переходы из блока  $B_i$

### Преобразование алгоритма управления

Обычно алгоритм управления задается граф-схемой алгоритма, по которой синтезируется автомат. Преобразование ГСА может выполняться в следующих случаях [5]:

- Условие (2) не выполняется, при этом для реализации автомата необходимо использовать сложные факторизационные методы [2], что резко увеличивает аппаратные затраты в схеме и время цикла за счет введения дополнительных уровней. Преобразование заключается в введении в ГСА дополнительных операторных вершин для разрыва длинных цепочек условных вершин. Введение операторных вершин приводит к росту числа состояний автомата и, возможно, параметра  $R$ . Таким образом процедура носит итерационный характер. Задача может не иметь решения, если после преобразований выполняется условие:
 
$$1+R > S \quad (3)$$
- Для минимизации сложности М-подсхемы ГСА преобразовывается так, чтобы переход из каждого состояния автомата зависел не более, чем от одного логического условия. Такие автоматы называются  $M_0P$ -автоматами [9]. В  $M_0P$ -автоматах М-подсхема реализуется на одном мультиплексоре ( $G=1$ ), а БИС Р-подсхемы требуют  $R+1$  входов, что позволяет использовать для реализации ПЗУ.
- Для минимизации сложности D-подсхемы ГСА преобразовывается так, чтобы в каждой вершине была записана только одна микрооперация [5]. При этом все микрооперации автомата являются совместимыми и D-подсхема реализуется на одном дешифраторе, имеющем  $\lceil \log_2 N \rceil$  входов и  $N$  выходов.

Преобразование алгоритма управления всегда связано с введением дополнительных состояний и увеличением времени выполнения алгоритма. Такое преобразование

целесообразно, если приводит к уменьшению стоимости автомата по сравнению с исходной реализацией (по исходной ГСА).

### Учет линейных участков ГСА

Если исходная ГСА имеет более 75% операторных вершин, то она называется линейной ГСА [5]. В линейных ГСА имеется значительное число линейных участков – последовательностей операторных вершин, между которыми нет условных вершин. Если разбить множество состояний  $A = \{a_1, \dots, a_M\}$  на блоки  $\alpha_1, \dots, \alpha_G$ , соответствующие операторным линейным цепям [5], и выполнить в пределах каждого блока кодирование состояний, при котором:

$$k(a_s) = k(a_m) + 1, \quad (4)$$

где  $a_m$  – исходное состояние автомата,  $a_s$  – состояние перехода, то регистр памяти автомата может быть заменен счетчиком. На рис.4 приведена структурная схема автомата Мура на счетчике.

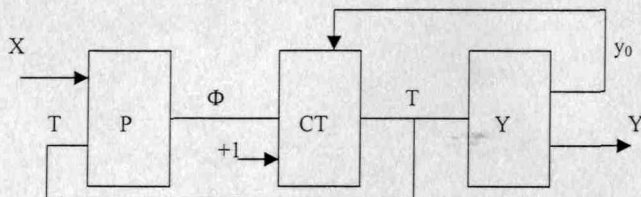


Рис.4 Структурная схема автомата Мура на счетчике

Если для состояний  $a_m, a_s$  выполняется (4), то формируется сигнал  $y_0$  и к содержимому счетчика СТ прибавляется единица, за счет чего и происходит очередной переход автомата. Если условие (4) не выполняется, то Р-подсхема формирует сигнал  $\Phi$  для занесения в СТ нужного кода.

Применение этого метода позволяет уменьшить сложность Р-подсхемы за счет исключения из ПСТ переходов, удовлетворяющих условию (4). Недостаток подобной организации – коды состояний зависят от места состояния в блоке, то есть задаются жестко. Это затрудняет применение алгоритмических методов для оптимизации схемы автомата.

### Комплексные методы

Их суть заключается в одновременном применении различных методов оптимизации. Например, при синтезе МРУ-автомата (структурная редукция) может использоваться оптимальное кодирование состояний (алгоритмический метод) и реализация М-подсхемы на мультиплексорах (гетерогенная реализация).

## 2. Выбор метода оптимизации

Выбор метода оптимизации зависит от характеристик алгоритма управления, критерия оптимизации и имеющегося в наличии элементного базиса.

Если критерием оптимальности схемы является максимальное быстродействие, то используется одноуровневая реализация в сочетании с алгоритмическими методами.

Если критерием оптимальности является минимум аппаратных затрат, то может быть использовано любое сочетание методов оптимизации. Основная проблема при этом заключается в отсутствии достоверных оценок числа ПЛМ в схеме автомата на основании анализа его характеристик и параметров элементного базиса. Это не позволяет априорно оценить сложность реализации без синтеза схемы автомата.

### **Заключение**

Методы оптимизации устройств управления на программируемых БИС отличаются большим разнообразием. Это вызывает необходимость разработки формальных методов выбора совокупности методов, позволяющих оптимизировать схему автомата при конкретном сочетании характеристик автомата и элементного базиса. В настоящее время априорные оценки числа ПЛМ или ПМЛ в схеме автомата отсутствуют и оценить сложность схемы можно только после ее синтеза. Один из путей решения этой проблемы – использование методов теории эксперимента для получения статистических зависимостей сложности реализации от параметров автомата и базиса для конкретного метода синтеза.

Подобные оценки могут быть использованы в экспертных системах, включаемых в состав САПР устройств управления для априорного выбора совокупности методов оптимизации логической схемы автомата.

### **Литература**

1. Баранов С.И. Синтез микропрограммных автоматов. – Л.: Энергия, 1979. – 232 с.
2. Соловьев В.В. Проектирование функциональных узлов цифровых систем на программируемых логических устройствах. – Минск: Бестпринт, 1996.-252с.
3. Скларов В.А. Синтез автоматов на матрицах БИС. – Минск: Наука и техника, 1984. – 287 с.
4. Оранов А.М. О комбинаторном сходстве задач синтеза и компоновки схем// Автоматизация проектирования дискретных систем. Материалы II международной конференции (12-14 ноября 1997 года, Минск – том 2). – Минск: ИТК МАН Беларуси, 1997. – с.113-118.
5. Баркалов А.А., Палагин А.В., Синтез микропрограммных устройств управления. – Киев: ЧКМАН Украины, 1997.- 135 с.
6. Баркалов А.А., Юсифов С.И., Дас Д.К. PR-автоматы: функционирование, оптимизация, выбор метода синтеза. – Киев: ЧК АН Украины, Препринт № 91-55, 1991.-25 с.
7. Баркалов А.А., Дас Д.К. Оптимизация логической схемы автомата Мили на ПЛМ // Кибернетика и системный анализ, 1991.-№5. – с.180-184.
8. Баркалов А.А. Принципы оптимизации логической схемы микропрограммного автомата Мура // Кибернетика и системный анализ. – 1998. - №1. – с.65-72.
9. Баркалов А.А., Зеленева И.Я, Бабаков Р.М. Структуры логических схем управляющих автоматов на программируемых БИС. // Научные работы ДГТУ. Серия “Информатика, кибернетика и вычислительная техника”. Выпуск 6. – Донецк: ДГТУ, 1999. – с.207-211.