

Лекция 3.

Двухуровневый логический синтез

Математические модели и методы синтеза СБИС
Осень 2014



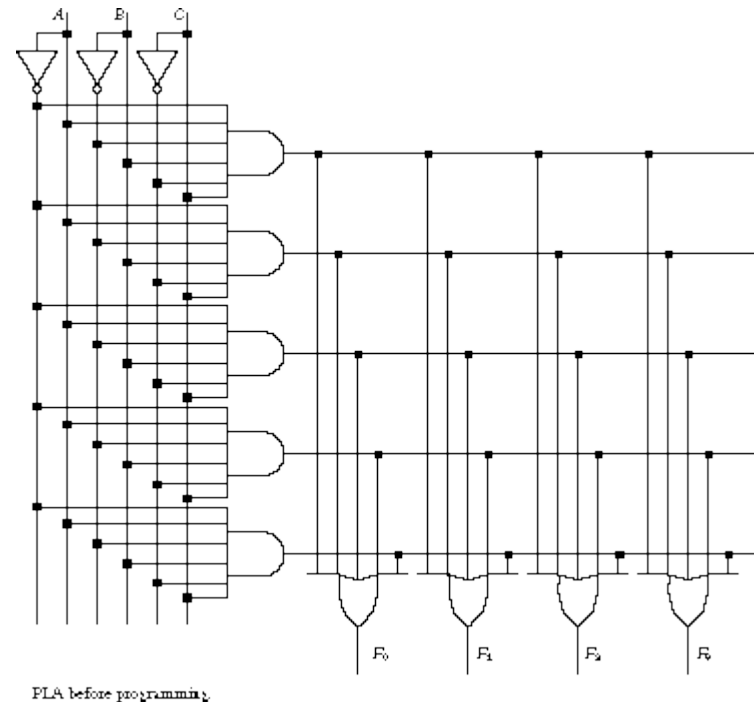
Двухуровневый логический синтез

- Реализация булевых функций в виде ДНФ и программируемые логические матрицы (ПЛМ)
- Эвристические подходы к минимизации ДНФ
- Обобщенно-монотонное разложение булевых функций
- Основные этапы алгоритма ESPRESSO

Реализация булевых функций в виде ДНФ

Программируемые логически матрицы (ПЛМ)

- Реализация системы булевых функций на основе ДНФ
- Первая интегральная схема TMS2000 (Texas Instruments, 1970)
- Оптимизация числа литералов приводит к уменьшению площади схемы.



Минимизация ДНФ – основные подходы

- Известные методы минимизации ДНФ
 - Эквивалентные преобразования. Сложно применим при большом числе переменных и неясно как оценить качество полученного результата.
 - Метод карт Карно. Применим только при малом числе переменных.
 - Метод Квайна. Экспоненциальная сложность в худшем случае.
- Эвристические подходы - свойства.
 - Поиск ДНФ близких к минимальным.
 - Последовательное (итеративное) улучшение текущего решения.
- Эвристические подходы
 - Градиентный метод
 - Алгоритм ESPRESSO

Алгоритм ESPRESSO – общая идея

- Пример – функция 4-х переменных, заданная таблицей
- Указаны только единичные наборы
- Покрытие построено из некоторого набора импликант функции (не обязательно простых)

		x_1			
		0	0	1	1
x_3	x_4	x_2			
	0	0	1	1	0
0	0	1	1		
0	1	1	1	1	1
1	1			1	1
1	0	1	1	1	1

Грани

K_1 0*0*

K_2 0*10

K_3 1001

K_4 1101

K_5 1*1*

Алгоритм ESPRESSO – общая идея

- «Расширение» граней (EXPAND) – последовательное преобразование граней в максимальные грани
- Для каждой грани может быть несколько различных способов сделать её максимальной
- В нашем примере «расширяются» грани K_2, K_3, K_4
- Полученное покрытие состоит только из максимальных граней
- Полученное покрытие может не быть тупиковым или минимальным

	x_1	0	0	1	1
	x_2	0	1	1	0
x_3	x_4				
0	0	1	1		
0	1	1	1	1	1
1	1			1	1
1	0	1	1	1	1

Грани

K_1	0*0*
K_2	**10
K_3	1001
K_4	1101
K_5	1*1*

Алгоритм ESPRESSO – общая идея

- «Расширение» граней (EXPAND) – последовательное преобразование граней в максимальные грани
- Для каждой грани может быть несколько различных способов сделать её максимальной
- В нашем примере «расширяются» грани K_2, K_3, K_4
- Полученное покрытие состоит только из максимальных граней
- Полученное покрытие может не быть тупиковым или минимальным

	x_1	0	0	1	1
	x_2	0	1	1	0
x_3	x_4				
0	0	1	1		
0	1	1	1	1	1
1	1			1	1
1	0	1	1	1	1

Грани

K_1	0*0*
K_2	**10
K_3	1**1
K_4	1101
K_5	1*1*

Алгоритм ESPRESSO – общая идея

- «Расширение» граней (EXPAND) – последовательное преобразование граней в максимальные грани
- Для каждой грани может быть несколько различных способов сделать её максимальной
- В нашем примере «расширяются» грани K_2, K_3, K_4
- Полученное покрытие состоит только из максимальных граней
- Полученное покрытие может не быть тупиковым или минимальным

	x_1	0	0	1	1
	x_2	0	1	1	0
x_3	x_4				
0	0	1	1		
0	1	1	1	1	1
1	1			1	1
1	0	1	1	1	1

Грани

K_1	0*0*
K_2	**10
K_3	1**1
K_4	**01
K_5	1*1*

Алгоритм ESPRESSO – общая идея

- «Расширение» граней (EXPAND) – последовательное преобразование граней в максимальные грани
- Для каждой грани может быть несколько различных способов сделать её максимальной
- В нашем примере «расширяются» грани K_2, K_3, K_4
- Полученное покрытие состоит только из максимальных граней
- Полученное покрытие может не быть тупиковым или минимальным

	x_1	0	0	1	1
	x_2	0	1	1	0
x_3	x_4				
0	0	1	1		
0	1	1	1	1	1
1	1			1	1
1	0	1	1	1	1

Грани

K_1	0*0*
K_2	**10
K_3	1**1
K_4	**01
K_5	1*1*

Алгоритм ESPRESSO – общая идея

- «Тупиковое покрытие» – удаление граней, которые покрываются другими максимальными гранями (IRREDUNDANT)
- В нашем примере можно удалить K_3
- Полученное покрытие является тупиковым
- Полученное покрытие может не быть минимальным

	x_1	0	0	1	1
	x_2	0	1	1	0
x_3	x_4				
0	0	1	1		
0	1	1	1	1	1
1	1			1	1
1	0	1	1	1	1

Грани

K_1 0*0*

K_2 **10

K_3 1**1

K_4 **01

K_5 1*1*

Алгоритм ESPRESSO – общая идея

- «Тупиковое покрытие» – удаление граней, которые покрываются другими максимальными гранями (IRREDUNDANT)
- В нашем примере можно удалить K_3
- Полученное покрытие является тупиковым
- Полученное покрытие может не быть минимальным

	x_1	0	0	1	1
	x_2	0	1	1	0
x_3	x_4				
0	0	1	1		
0	1	1	1	1	1
1	1			1	1
1	0	1	1	1	1

Грани

K_1 0*0*

K_2 **10

K_4 **01

K_5 1*1*

Алгоритм ESPRESSO – общая идея

- «Сужение» граней (REDUCE) – последовательное уменьшение каждой из граней при сохранении покрытия
- Результирующее покрытие состоит не только из максимальных граней
- Позволяет изменить форму покрытия
- Порядок «сужения» граней имеет важное значение
- Является ключевым шагом, так как позволяет в дальнейшем произвести такое «расширение» граней, которое позволит покрыть другие грани
- В нашем примере можно «сузить», например, K_2 и K_4

	x_1	0	0	1	1
	x_2	0	1	1	0
x_3	x_4				
0	0	1	1		
0	1	1	1	1	1
1	1			1	1
1	0	1	1	1	1

Грани

K_1	0*0*
K_2	0*10
K_4	**01
K_5	1*1*

Алгоритм ESPRESSO – общая идея

- «Сужение» граней (REDUCE) – последовательное уменьшение каждой из граней при сохранении покрытия
- Результирующее покрытие состоит не только из максимальных граней
- Позволяет изменить форму покрытия
- Порядок «сужения» граней имеет важное значение
- Является ключевым шагом, так как позволяет в дальнейшем произвести такое «расширение» граней, которое позволит покрыть другие грани
- В нашем примере можно «сузить», например, K_2 и K_4

	x_1	0	0	1	1
	x_2	0	1	1	0
x_3	x_4				
0	0	1	1		
0	1	1	1	1	1
1	1			1	1
1	0	1	1	1	1

Грани

K_1 0*0*

K_2 0*10

K_4 1*01

K_5 1*1*

Алгоритм ESPRESSO – общая идея

- «Сужение» граней (REDUCE) – последовательное уменьшение каждой из граней при сохранении покрытия
- Результирующее покрытие состоит не только из максимальных граней
- Позволяет изменить форму покрытия
- Порядок «сужения» граней имеет важное значение
- Является ключевым шагом, так как позволяет в дальнейшем произвести такое «расширение» граней, которое позволит покрыть другие грани
- В нашем примере можно «сузить», например, K_2 и K_4

	x_1	0	0	1	1
	x_2	0	1	1	0
x_3	x_4				
0	0	1	1		
0	1	1	1	1	1
1	1			1	1
1	0	1	1	1	1

Грани

K_1 0*0*

K_2 0*10

K_4 1*01

K_5 1*1*

Алгоритм ESPRESSO – общая идея

	x_1	0	0	1	1
	x_2	0	1	1	0
x_3	x_4				
0	0	1	1		
0	1	1	1	1	1
1	1			1	1
1	0	1	1	1	1

Грани

K_1 0*0*

K_2 **10

K_4 1**1

K_5 1*1*

«Расширение» граней

	x_1	0	0	1	1
	x_2	0	1	1	0
x_3	x_4				
0	0	1	1		
0	1	1	1	1	1
1	1			1	1
1	0	1	1	1	1

Грани

K_1 0*0*

K_2 **10

K_4 1**1

K_5 1*1*

«Тупиковое покрытие»

Алгоритм ESPRESSO

- Этапы алгоритма
 - «Сужение» граней (REDUCE)
 - «Расширение» граней (EXPAND)
 - «Тупиковое покрытие» (IRREDUNDANT)
- Разработка алгоритма началась в компании IBM и была завершена в университет Berkeley
- Литература
 - Brayton, Hachtel, McMullen, Sangiovanni-Vincentelli, Logic Minimization Algorithms for VLSI Synthesis, Kluwer Academic Press, 1984
 - Richard L. Rudell, (1986-06-05), “Multiple-Valued Logic Minimization for PLA Synthesis” Memo No. UCB/ERL M86-65 (U. California Berkeley M.S. Thesis)
 - Giovanni DeMicheli, Synthesis and Optimization of Digital Circuits, McGraw Hill, 1994

Задача проверки тождественности

- Дано:
 - ДНФ для ФАЛ f (в РСН формате).
- Задача:
 - Установить, является ли ФАЛ f тождественно равной 1.

Рекурсивная проверка тождественности

- Разложение Шеннона:
- Основные вопросы:
 - Выбор переменной: по какой переменной лучше всего проводить разложение?
 - Критерий останова: когда можно остановить рекурсию?
 - Особенности реализации: как эффективно хранить подфункции в памяти?

Обобщенно-монотонные ФАЛ и поляризованные ДНФ

Рекурсивное разложения ФАЛ по обобщенно-монотонным ФАЛ

Построение тупиковых ДНФ – процедура IRREDUNDANT

- Метод Квайна
 - Построение сокращенной ДНФ
 - Размер сокращенной ДНФ
 - Нахождение всех тупиковых ДНФ
 - Градиентный метод
- Процедура IRREDUNDANT
 - Построение тупиковой ДНФ на основе подмножества простых импликант, образующих покрытие ФАЛ
 - Использование рекурсивного разложения по обобщенно-монотонным ФАЛ

Построение тупиковых ДНФ – процедура IRREDUNDANT