инструментов) позволяет разработчику более смело экспериментировать в области архитектурных и схемных решений.

1. Bastian J. D. et al. Symbolic parasitic extractor for circuit simulation (SPECS) // Proc. 20th Design Automation Conf.—Las Vegas, 1983.—P. 346.
2. Tarolli G., Herman W. J. Hierarchical circuit extraction with detailed parasitic capacitance // Ibid.—P. 337.
3. McCormick S. P. EXCL: a circuit extractor for IC designs // Proc. 21st Design Automation Conf.—Las Vegas, 1984.—P. 624.
4. Wagner T. G. Hierarchical layout verification // IEEE J. Design and Test of Computers.—1985.—2, N 1.—P. 31.
5. Scott W. S., Ousterhout J. K. The Magic circuit extractor // Ibid.—1986.—3, N 1.—P. 24.
6. Ousterhout J. K. Corner-stitching: a data-structuring technique for VLSI layout tools // IEEE Trans. on CAD.—1984.—CAD-3, N 1.—P. 87.
7. Лившиц З. А., Титов Д. Г. Алгоритмы работы с тайловыми представлениями топологии СБИС // Автометрия.—1991.—№ 3.
8. Лившиц З. А., Пичуев А. В. SimSim: программа логического моделирования МОП СБИС на переключательном уровне // Там же.
9. Ульман Дж. Вычислительные аспекты СБИС.—М.: Радио и связь, 1990.
10. Титов Д. Г. Система проектирования топологии интегральных схем ICE.—Новосибирск, 1991.—(Препр. АН СССР, Сиб. отд-ние. ИАиЭ; 464).
11. Pucknell D. A., Eshraghian K. Basic VLSI Design.—Prentice-Hall, 1988.
12. Mead C., Conway L. Introduction to VLSI Systems.—Addison-Wesley, 1980.
13. Electronic Industries Association, Electronic Design Interchange Format Version 200 // EIA Interim Standard.—1987.—N 44.

Поступила в редакцию 7 мая 1991 г.

УДК 681.32.06 : 519.87

Д. В. Наговицын, К. К. Смирнов

(Новосибирск)

ОПТИМИЗАЦИЯ ВЫХОДНОЙ ФАЗЫ ПРОГРАММИРУЕМЫХ ЛОГИЧЕСКИХ МАТРИЦ

Рассматриваются методы сокращения числа термов системы логических функций, основанные на отыскании оптимального фазового вектора системы, определяющего, какие функции следует инвертировать, а какие оставить неизменными. Проведен сравнительный анализ реализованных известных алгоритмов с некоторыми разработанными модификациями.

Введение. Высокая регулярность структуры ПЛМ обусловливает достаточную простоту их применения при синтезе топологии заказных и полузаказных СБИС и значительно сокращает время проектирования. Эти факторы являются во многих случаях решающими при разработках сложной управляющей логики. В связи с этим возникает основная проблема, связанная с использованием ПЛМ, — сокращение занимаемой на кристалле площади. Традиционные методы ее решения состоят в непосредственной минимизации реализуемой с помощью ПЛМ системы логических функций в смысле нахождения дизъюнктивной нормальной

Таблица 1

1100	100
0-00	010
0110	100
1-10	010
-0-0	001
0001	010
1101	010
1011	010
0111	010
-1-1	001
100-	001
-0-0	100
001-	100

формы системы, имеющей минимальное число различных конъюнктивных членов.

В данной работе рассмотрен еще один подход к улучшению характеристик ПЛМ, основанный на том, что при замене некоторых функций в исходной системе на инверсные новая система, возможно, реализуема меньшим количеством термов. Это позволяет сформулировать задачу назначения оптимальной выходной фазы: найти те функции, при инвертировании которых вновь полученная система реализуется минимальным числом термов. В настоящее время неизвестен алгоритм точного решения этой задачи, отличный от полного перебора, сложность которого имеет экспоненциальную зависимость от числа выходных функций в системе. Это стимулирует поиск алгоритмов, дающих хорошее приближение к точному решению.

В статье описаны реализации некоторых известных ([1, 2]) эвристических алгоритмов. Предложены также новые варианты решения задачи оптимизации выходной фазы и проведен их сравнительный анализ с ранее существующими.

1. **Базовые алгоритмы.** Для представления системы логических функций будем использовать формат таблицы истинности, в которой каждая строка, называемая термом, состоит из входной и выходной частей. Входная часть является последовательностью символов в алфавите (0, 1, -) длиной, равной числу входов ПЛМ. Выходная часть состоит из последовательности символов (0, 1) длиной, равной количеству функций в системе.

Таблица 2

	Выходная часть двухфазной матрицы		
	Шаг 1	Шаг 2	Шаг 3
	+ *	* +	*+
1110	101010*	0100	10
101-	100010*	0001	01*
-0-1	001000	1100	10
-0-0	000001	1100	10
0110	011100	0110*	01*
1100	011100	1000	
0100	001110	1000	
001-	010100	0010*	
100-	010100	1001	
000-	000110		
0101	010101		
1101	100011*		
0111	100011*		
1111	110001*		

Таблица 3

1110	110
101-	110
-0-0	001
0100	010
000-	010
0101	001
1101	111
0111	111
1111	101

В табл. 1 в качестве примера приведена таблица истинности двухразрядного сумматора. При этом каждый из символов отображает определенное соединение или его отсутствие (прочерк) в ПЛМ. Нуль во входной части определяет подключение инверсного, а единица — прямого входных сигналов к линии терма в плоскости И. Единицы в выходной части указывают на наличие соединения линий элементарных конъюнкций с соответствующей выходной линией в плоскости ИЛИ.

Пусть дана система логических функций, заданная некоторой таблицей истинности. Введем несколько понятий и соответствующих обозначений, которые используются в описании алгоритмов.

Будем называть выходным вектором i-го терма вектор O_i, представляющий выходную часть терма. Обозначим через $D(O_i)$ множество различных термов, выходной вектор которых равен O_i, а через $R(O_i)$ количество элементов в этом множестве. Задачу оптимального назначения фазы будем рассматривать как нахождение некоторого вектора F (фазового вектора), координаты которого определяют, какие функции системы должны остаться неизменными ($F_i = 0$), а какие инвертироваться ($F_i = 1$).

Идея всех алгоритмов назначения фазового вектора состоит в том, что при любом выборе фазового вектора соответствующая ему система функций реализуется некоторым подмножеством термов «двухфазной» системы, полученной объединением исходной и инверсной систем функций.

1.1. *Метод последовательного назначения.* Предложенная в [1] процедура назначения фазового вектора работает с двухфазной матрицей минимизированной системы $2n$ функций. Алгоритм предусматривает последовательное выполнение следующих шагов.

1. Нахождение выходной функции, реализующейся максимальным числом термов. Если столбец двухфазной матрицы, имеющий наибольшее количество единиц, соответствует функции f_i ($\bar{f}_i$), то в результирующий набор системы войдут термы, реализующие функцию f_i ($\bar{f}_i$), при этом i-я координата фазового вектора полагается равной 1(0).

2. Редуцирование матрицы. Столбец, соответствующий функции, найденной на шаге 1, и все термы, выходные векторы которых содержат в позиции выбранного столбца единицы, и далее сам выбранный столбец удаляются из матрицы. Термы с нулевыми выходными векторами также удаляются. Если редуцированная матрица не пуста, шаги 1 и 2 повторяются.

По существу, этот алгоритм заключается в последовательном удалении из матрицы термов, реализующих отдельные функции; критерием, по которому происходит удаление, является максимальность количества термов в прямой либо в инвертированной функции. В качестве иллюстрации алгоритма приведен пример нахождения оптимального фазового вектора для системы логических функций, реализующих двухразрядный сумматор. В табл. 2 представлена система $2n$ функций, и далее изображены шаги последовательного назначения фазового вектора. Мы будем использовать только выходную часть, поэтому в промежуточных шагах фигурирует только она. Расположение функций в выходной части соответствует $f1$, $f2$, $f3$, $\overline{f1}, \overline{f2}, \overline{f3}$; термы и столбцы, которые будут удалены на следующем шаге, помечены звездочкой, плюс соответствует выбранной функции.

В табл. 3 представлен результирующий набор термов, соответствующий фазовому вектору (0 1 1) (т. е. функции $f2$ и $f3$ проинвертированы). Исходная таблица истинности содержала 11 термов, в то время как полученная после оптимизации выходной фазы содержит 9.

1.2. *Итеративный алгоритм последовательного улучшения.* В работе [2] предложен алгоритм, использующий иную стратегию в назначении

Параметры теста		Итеративный вариант алгоритма последовательного назначения			Алгоритм PLAYGROUND		
Схема	Начальное кол-во термов	Кол-во термов после оптимизации	Кол-во итераций	Процессорное время	Кол-во термов после оптимизации	Кол-во итераций	Процессорное время
*add*4	75	61	3	00:48.46	61	3	00:42.08
*ml*4	124	111	2	01:46.89	112	2	01:41.41
*ac*4	135	105	3	01:34.29	105	3	01:23.44
*add*2	11	9	3	00:02.97	9	3	00:02.88
*ad*5	167	137	3	04:12.07	137	3	04:26.76
*c*4	39	38	2	00:07.59	38	3	00:12.76
*c*8	69	65	3	00:57.78	65	4	01:26.47
*c*16	305	304	3	17:17.81	304	2	10:52.80
dis	120	105	5	03:20.24	114	2	00:49.43
*sq*6	50	42	4	00:49.37	40	2	00:16.68
*ad*6	355	293	3	14:24.65	293	3	13:00.04
*r*18	179	177	4	10:21.55	176	3	07:19.49
*r*3	220	218	4	16:55.68	220	2	04:55.06
*r*6	170	167	3	05:47.79	167	4	09:04.07
*r*21	231	231	3	11:01.89	232	3	09:53.38
*r*14	171	170	1	01:02:66	171	2	03:40.85

выходной фазы системы логических функций, получивший название PLAYGROUND. Он включает следующие основные этапы:

1. В качестве текущей выбирается исходная система функций, в качестве текущего фазового вектора — нулевой.

2. Для текущей системы функций вычисляется минимизированная двухфазная матрица.

3. Выбирается «фаза» (при этом фазовый вектор не изменяется), т. е. прямые или инверсные значения текущей системы функций (предпочтение отдается фазе, которой соответствует минимальное число ненулевых термов). При выборе инверсной системы текущий фазовый вектор инвертируется.

4. Для выбранной фазы определяется ненулевой выходной вектор $\widetilde{O}$, для которого значение $R(\widetilde{O})$ максимально (т. е. максимизируется число различных термов, соответствующих одному выходному вектору выбранной фазы текущей системы). Определяем «новое» текущее значение фазового вектора; оно полагается равным сумме по модулю 2 «старого» текущего значения и вектора O.

5. Определяем новую текущую систему функций, в качестве которой берется система, соответствующая новому фазовому вектору. Эта система минимизируется. Если получено большее количество термов, чем на предыдущей итерации, то процесс вычислений заканчивается и решением

Таблица 4

Модификация алгоритма PLAYGROUND		
Кол-во термов после оптимизации	Кол-во итераций	Процессорное время
61	3	00:44.89
110	3	02:52.33
105	3	01:19.27
9	3	00:02.74
137	3	04:05.96
38	3	00:12.12
65	3	00:54.67
304	3	19:04.06
105	5	03:15.62
39	3	00:32.77
293	3	13:34.42
178	3	06:17.41
217	6	24:17.91
167	3	05:47.85
231	5	21:13.53
169	3	06:71.88

будет фазовый вектор, полученный на предыдущей итерации, в противном случае осуществляется переход на шаг 2.

Сравнивая методы, предложенные в [1, 2], можно отметить, что основным преимуществом первого из указанных алгоритмов является то, что процесс назначения фазового вектора осуществляется за один проход, в то время как в итеративной программе PLAYGROUND приходится неоднократно решать трудоемкую задачу минимизации системы функций. В то же время второй алгоритм в большинстве случаев дает лучшие результаты (в смысле уменьшения количества термов).

2. Модификации базовых алгоритмов. Разработка программных средств оптимального назначения выходной фазы является составной частью проекта интегрированной системы синтеза управляющей логики на основе ПЛМ, осуществляемого в ИАиЭ СО АН СССР [3]. Для решения этой задачи были выполнены программные реализации описанных выше базовых алгоритмов, предложен ряд модификаций этих алгоритмов и произведено экспериментальное сравнение различных подходов на «модельных» и практических примерах.

Так, наиболее очевидная модификация метода последовательных приближений — его итеративный вариант, при котором система логических функций, полученная после назначения фазового вектора, если количество термов в ней после минимизации уменьшилось, используется в качестве исходной для следующей итерации. Такой итеративный алгоритм, естественно, дает «не худшие» результаты по уменьшению количества термов, однако требует больших затрат времени. Другим возможным объектом для модификации является шаг редуцирования двухфазной матрицы: полученная после удаления столбцов и термов система минимизируется с тем, чтобы более адекватно представить начальные условия для очередного шага выбора компоненты фазового вектора.

В данной работе также предложена и апробирована версия итеративного алгоритма, отличающегося от процедуры [2] стратегией выбора текущего фазового вектора (см. шаги 3 и 4 в п. 1.2). Суть ее заключается в следующем. Рассматриваются все различные выходные векторы текущей двухфазной системы логических функций, каждому из которых соответствуют два возможных варианта фазового вектора (прямая или инвертированная система). В качестве нового текущего выбирается тот фазовый вектор, которому соответствует максимальное сокращение числа термов в двухфазной матрице. Такой подход позволит отыскивать минимум на шаге выбора фазового вектора, исходя из более обширного множества возможных фазовых векторов по сравнению с [2].

3. Реализация. Экспериментальные результаты. Программная реализация всех указанных выше алгоритмов была осуществлена на языке Си в операционной среде VAX/VMS. В качестве встроенного средства минимизации использовалась программа LOGIC [3].

Экспериментальная проверка показала, что наиболее эффективными процедурами (имея в виду как количество термов для результирующего набора, так и время отработки) являются следующие методы: итеративный вариант алгоритма последовательного назначения, алгоритм последовательного улучшения фазового вектора и его модификация, рассмотренная в п. 2. Некоторые результаты по характеристикам указанных алгоритмов сведены в табл. 4.

Из таблицы видно, в частности, что за счет оптимального выбора фазового вектора удается в среднем на 10—15 % сократить число термов (т. е. площадь ПЛМ) по сравнению с минимальным покрытием исходной системы функций.

Из анализа данных табл. 4 следует, что все приведенные в ней алгоритмы имеют сравнимые характеристики, при этом модифицированный итеративный алгоритм обладает определенными преимуществами в части минимизации числа термов, а итеративный вариант алгоритма последовательного назначения — в скорости сходимости.

СПИСОК ЛИТЕРАТУРЫ

1. Sasao T. Input variable assignment and output phase optimization of PLA's // IEEE Trans. Comput.—1984.—C-33, N 10.—P. 879.
2. Wey C. L., Chang T. S. An efficient output phase assignment for PLA's minimization // IEEE Trans. Comput.—1990.—9, N 1.
3. Лившиц З. А., Смирнов К. К. LOGIC — программный комплекс для генерации топологии ПЛМ // Автометрия.—1991.—№ 1.

Поступила в редакцию 27 марта 1991 г.