

АВТОМАТИЗАЦИЯ ЛОГИЧЕСКОГО ПРОЕКТИРОВАНИЯ

УДК 519.714.5

Л.Д. Черемисинова

МИНИМИЗАЦИЯ ПЛОЩАДИ РЕГУЛЯРНЫХ МАТРИЧНЫХ СТРУКТУР ЗАКАЗНЫХ СБИС МЕТОДОМ СВЕРТКИ

Рассматривается проблема оптимизации площади двумерных матричных структур управляющей логики заказных СБИС. Описываются пути решения двух взаимосвязанных задач: минимизации функционального описания блока управляющей логики, реализуемого матричной структурой, и укладки этого описания на кристалл. Первая задача решается на этапе логического проектирования матричной структуры, вторая – на этапе ее топологического проектирования. Площадь матричных структур существенно зависит от того, насколько хорошо эти задачи будут решены на обоих этапах.

Введение

При проектировании управляющей логики заказных СБИС в качестве базовых широко используются программируемые матричные структуры. Для них возможна автоматическая генерация топологии по их структурному (или функциональному) описанию, облегчается решение задач диагностики и логического проектирования. Базовые структуры ориентируются, как правило, на некоторую технологию изготовления СБИС. Такими структурами, в частности, являются двумерные регулярные структуры типа программируемых логических матриц (ПЛМ), матриц Вайнбергера (МВ), транзисторных матриц, регулярных схем на базе последовательно соединенных МОП-транзисторов (РМОП-схема) [1, 2]. За основной критерий оптимальности при проектировании СБИС на основе регулярных структур принимается площадь кристалла.

Топологическая реализация регулярной структуры обычно выполняется в два этапа: получение и оптимизация структурного описания, задающего символьное представление топологии; генерация и оптимизация топологического рисунка. Комбинаторная задача оптимизации символьной топологии является узким местом топологического проектирования, и от решения этой задачи зависит конкурентоспособность автоматически сгенерированной топологии по сравнению с полученной вручную. Структурное описание, задающее символьное представление топологии, генерируется по функциональному описанию блока управляющей логики – представлению реализуемой системы логических выражений в виде суперпозиции элементарных функций, реализуемых регулярной структурой выбранного типа.

Сложность функционального описания оказывает существенное влияние на площадь, занимаемую регулярной матричной структурой на кристалле. Последствия неоптимальности функционального описания не могут быть в полной мере устранены без изменения логики проектируемого устройства при укладке структуры на плоскость кристалла на этапе топологического проектирования. Отсюда следует, что оптимизация символьной топологии начинается с этапа логического проектирования блока управляющей логики, когда получается такое функциональное описание, которое не только укладывается на матричную структуру выбранного типа, но и обеспечивает минимум занимаемой площади.

Существенным недостатком матричных структур на этапе топологического проектирования является то, что они, имея регулярную организацию, проигрывают многоуровневым реализациям на основе произвольной логики по площади, занимаемой на кристалле, за счет неэффективного ее использования. Последнее выражается в сильной разреженности матричных структур. Один из ранее широко использовавшихся на практике способов топологической оптимизации заключался в разбиении символьного представления матричной структуры на «густые» аппликации и их сшивании. Этот метод позволял локально оптимизировать матричную

структуру. Более глубокую оптимизацию обеспечивает широко применяемый в последние годы метод топологической оптимизации матричных структур, известный как свертка [3 – 5]. Свертка основана на разрыве шин матричной структуры и реализации на одной вертикальной (и/или горизонтальной) шине двух или более ее столбцов (и/или строк). Алгоритмы свертки, не изменяя функциональности матричной структуры, основаны на поиске оптимального переупорядочения и совмещения ее столбцов и строк.

1. Программируемые матричные структуры заказных СБИС

ПЛМ предназначена для реализации системы дизъюнктивных нормальных форм (ДНФ) булевых функций и состоит из двух транзисторных матриц И и ИЛИ. Столбцам матриц соответствуют входные переменные (и их инверсии) и соответственно выходные. В строках матрицы И реализуются элементарные конъюнкции, подаваемые на строки матрицы ИЛИ, в столбцах которой реализуются дизъюнкции (или их инверсии в зависимости от программирования выходов) этих конъюнкций.

РМОП-схема так же, как и *ПЛМ*, реализует систему ДНФ или их инверсий. Ее ядром является матрица И, реализующая элементарные конъюнкции входных переменных. Но в отличие от *ПЛМ*, в которой используется параллельное соединение транзисторов, в *РМОП-структурах* транзисторы матрицы И соединены параллельно. Отсюда и второе различие: выходы матрицы И *РМОП-схемы* не могут использоваться для формирования нескольких ДНФ. Строки матрицы И делятся на секции, распределяемые между взаимно непересекающимися столбцами матрицы ИЛИ. Топологически матрица ИЛИ выполняется в виде одного столбца с разрывами. Конструктивной особенностью *РМОП-структур* является также ограничение на число транзисторов в каждой из строк матрицы И.

Матрица Вайнбергера предназначена для реализации многоуровневых комбинационных логических схем в базисе «отрицаний ДНФ». Конструктивно матрица Вайнбергера состоит из столбцов, некоторые из которых в верхней своей части имеют нагрузочные транзисторы, и строк, служащих для съема сигналов со входов и частей схемы – столбцов, снабженных нагрузочными транзисторами. Несколько связанных между собой столбцов, снабженных одним нагрузочным транзистором, реализуют функцию отрицания дизъюнкции конъюнкций, получаемых на связанных столбцах. Еще одной конструктивной особенностью матрицы Вайнбергера является ограничение на «длину» столбца – число переключателей на пересечении с каждой из строк (оно обуславливается размером нагрузочного транзистора) и на число связываемых между собой столбцов.

Описанные типы регулярных матричных структур заказных СБИС подробно описаны в монографиях [1, 2].

Можно считать, что матричная структура любого из упомянутых типов состоит из взаимно пересекающихся строк и столбцов, на пересечении которых находятся транзисторы (или схемы из транзисторов). Столбцам такой структурной матрицы соответствуют сигналы (входные, выходные, внутренние), строкам – элементы, задаваемые множеством соответствующих транзисторов. Программирование матричной структуры состоит в удалении из нее некоторых транзисторов путем разрушения соответствующих связей строк и столбцов. Площадь матричной структуры пропорциональна произведению чисел ее столбцов и строк.

2. Минимизация функционального описания блока управляющей логики на этапе логического проектирования

Структурное описание матричной структуры определяет число столбцов, строк и общее число транзисторов ее топологической реализации. Оно задает символическое представление топологии и генерируется по функциональному описанию блока управляющей логики – представлению реализуемой системы логических выражений в виде суперпозиции элементарных функций, реализуемых регулярной структурой выбранного типа.

Структурное описание *ПЛМ* и *РМОП-схемы* полностью определяется реализуемой ими системой ДНФ. Соответственно минимизация структурного описания этих матричных структур

на этапе логического проектирования сводится к минимизации описывающей их системы булевых функций в классе ДНФ. При этом важно минимизировать не только общее число элементарных конъюнкций ДНФ функций, но и число литералов (переменных и их инверсий) в конъюнкциях. Число конъюнкций определяет число столбцов матрицы И матричной структуры, а число литералов в конъюнкциях – число ее транзисторов, т.е. заполненность матрицы И. Кроме того, числа конъюнкций в отдельных ДНФ системы задают число транзисторов матрицы ИЛИ ПЛМ-структуры. Эти параметры очень важны, так как разреженные матричные структуры лучше свертываются, как это будет показано ниже, порождая более компактное по площади представление.

При реализации блока управляющей логики ПЛМ-структурой производится совместная минимизация функций системы в классе ДНФ, т.е. минимизируется сложность системы ДНФ в целом. При реализации управляющего блока РМОП-структурой производится отдельная минимизация функций системы в классе ДНФ, т.е. минимизируется сложность каждой ДНФ по отдельности. Конструктивное ограничение РМОП-структур на число транзисторов в строках матрицы И накладывает дополнительное условие на конъюнкции реализуемой системы ДНФ – они должны иметь ограниченный ранг. Эти ограничения учитываются также на этапе логического проектирования. Для этого используются методы декомпозиции и факторизации ДНФ.

Дополнительной оптимизации структурного описания ПЛМ и РМОП-схем можно достичь, если минимизировать булевы функции (совместно или отдельно) с учетом возможности выбора их полярности (в безынверсной или инверсной форме). Эта возможность появляется благодаря тому, что ПЛМ- и РМОП-структуры, как правило, снабжаются выходным буфером программируемых инверторов.

Структурное описание матрицы Вайнбергера – многоуровневая логическая сеть в базисе «инверсий ДНФ», таким образом на этапе логического проектирования должно быть получено представление реализуемых булевых функций в виде суперпозиции функций «инверсия ДНФ». Конструктивные особенности матрицы Вайнбергера накладывают ограничения на эти ДНФ (а точнее, их инверсии): они могут содержать более заданного числа конъюнкций ограниченного ранга. Сложность многоуровневой сети в базисе ограниченных «инверсий ДНФ» с точки зрения реализации матрицей Вайнбергера определяется суммарным числом конъюнкций всех инверсий ДНФ, а это значит, что реализуемые булевы функции должны быть представлены в виде суперпозиции функций одного и того же набора элементарных функций «инверсия ДНФ».

Наиболее естественный и практичный метод получения и оптимизации представления реализуемой системы булевых функций в виде суперпозиции ограниченных «инверсий ДНФ» состоит в решении следующих задач: совместная минимизация функций системы в классе ДНФ, совместная факторизация «длинных» конъюнкций всех ДНФ и дизъюнктивное разложение полученной системы ДНФ с «короткими» конъюнкциями. Последняя задача сводится к поиску минимального дизъюнктивного базиса – системы ограниченных ДНФ, через дизъюнкции которых представляются все функции исходной ДНФ.

В [6] приводятся разные способы реализации факторов, конъюнкций или их дизъюнкций, стоимостные оценки предпочтительности разных вариантов их реализации, ориентированные на особенности используемого базиса («инверсий ДНФ»), а также метод синтеза укладываемой на матрицу Вайнбергера многовыходной многоуровневой логической сети в базисе ограниченных «инверсий ДНФ».

3. Минимизация структурного описания блока управляющей логики на этапе топологического проектирования

Формальной топологической моделью матричной структуры для решения задачи ее свертки является булева матрица \mathbf{B} , ее единичный элемент $b_{ij} \in \mathbf{B}$ говорит о наличии транзистора на пересечении i -й строки и j -го столбца. Обозначим через $C(\mathbf{B}) = \{c_1, c_2, \dots, c_n\}$ и $R(\mathbf{B}) = \{r_1, r_2, \dots, r_m\}$ множества соответственно столбцов и строк матрицы \mathbf{B} , а через $R(c_j)$ и $C(r_i)$ – множества строк и столбцов, порождаемых каждым столбцом $c_j \in C(\mathbf{B})$ и строкой $r_i \in R(\mathbf{B})$ матрицы.

$R(c_j)$ ($C(r_i)$) представляет собой множество строк (столбцов), имеющих единицы на пересечении со столбцом c_j (строкой r_i):

$$r_i \in R(c_j) \leftrightarrow b_{ij} = 1 \quad (c_j \in C(r_i) \leftrightarrow b_{ij} = 1).$$

Столбцы c_k и c_l (строки r_k и r_l) не пересекаются, если

$$R(c_k) \cap R(c_l) = \emptyset \quad (C(r_k) \cap C(r_l) = \emptyset).$$

Непересекающиеся столбцы не имеют транзисторов на пересечении с одними и теми же строками. Матрица отношения непересечения столбцов или строк симметрична относительно главной диагонали (в силу симметричности этого отношения).

Набор $I^c_k = (c_{k1}, c_{k2}, \dots, c_{km})$ ($I^r_k = (r_{k1}, r_{k2}, \dots, r_{km})$) попарно непересекающихся столбцов c_{ki} (строк r_{ki}) (если не наложены никакие дополнительные ограничения на тип свертки) называется свертываемым. Далее всюду будем говорить «свертываемое множество столбцов», подразумевая и «или строк». Упорядоченный свертываемый набор столбцов будем обозначать через $I^{c_o}_k = \langle c_{k1}, c_{k2}, \dots, c_{km} \rangle$ в отличие от неупорядоченного $I^c_k = (c_{k1}, c_{k2}, \dots, c_{km})$. Неупорядоченный свертываемый набор порождает $m!$ различных упорядоченных. Под упорядочением набора I^c_k понимается выбор одного из этих упорядоченных наборов $I^{c_o}_k$. Свертываемый (упорядоченный) набор $I^c_k(I^{c_o}_k)$ из двух столбцов называется свертываемой парой c_{k1}, c_{k2} или $\langle c_{k1}, c_{k2} \rangle$ (неупорядоченной или упорядоченной).

Пример. В качестве примера матричной структуры возьмем ПЛМ, приведенную на рис. 1. Она описывается следующей булевой матрицей, состоящей из двух секций:

$$B = \begin{matrix} & c_1 & c_2 & c_3 & c_4 & c_5 & c_6 & c_7 & c_8 & & \\ r_1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & & \\ r_2 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & & \\ r_3 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & & \\ r_4 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & & \\ r_5 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & & \\ r_6 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & & \end{matrix}$$

В случае ПЛМ свертываться могут только столбцы из одной и той же секции (описывающей одну из матриц И или ИЛИ). Для ПЛМ, описываемой матрицей B , существует семь неупорядоченных свертываемых пар столбцов (или 14 упорядоченных):

$$P_u = \{c_1, c_2; c_7, c_8; c_1, c_4; c_1, c_5; c_2, c_3; c_2, c_5; c_5, c_6\}.$$

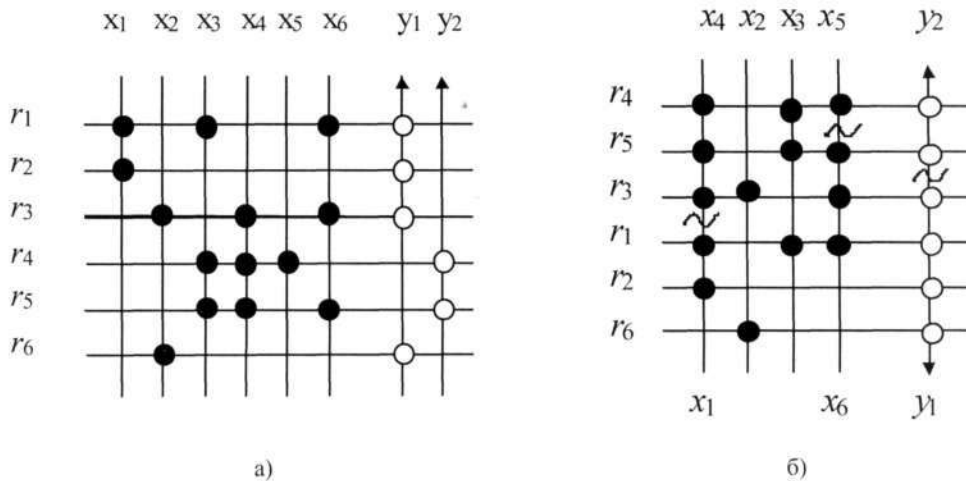


Рис. 1. Пример ПЛМ: а) исходная матрица; б) ее столбцовая свертка

Свертка матричных структур. Минимизация площади матричной структуры на этапе топологического проектирования достигается за счет свертки столбцов (и/или строк) матричной структуры – разрыва шин и реализации на одной вертикальной (и/или горизонтальной) шине двух или более исходных столбцов (и/или строк). В зависимости от числа разрывов выделяется простая (simple folding [3, 4]) и многократная (multiple folding [5]) свертка. При простой свертке столбцы и/или строки могут быть разделены на два сегмента. Соответственно внешние сигналы могут быть сняты с противоположных сторон матричной структуры (сверху и снизу) или на одной строке могут быть реализованы два элемента. При многократной свертке число свертываемых столбцов и/или строк (число сегментов) не ограничивается.

Кроме того, свертка может быть одномерной (столбцовой или строчной) или двумерной (столбцово-строчной). Очевидно, что простая одномерная свертка может обеспечить максимальное сокращение площади до 50%, двумерная – до 75%. Кроме того, выделяют частный вид свертки – двудольную свертку (bipartite folding) [7] – как особый тип простой свертки, в которой все разрывы находятся на одном уровне.

Для ПЛМ-структур возможны все виды сверток для обеих матриц И и ИЛИ [3, 4], для матриц Вайнбергера – многократная строчная свертка [8], для РМОП-структур используется только строчная свертка матрицы И ограниченного типа [9].

Каждая упорядоченная свертываемая пара $\langle c_{k1}, c_{k2} \rangle$ порождает отношение R_k^r : $R(c_{k1}) < R(c_{k2})$ частичного порядка на множестве $R(\mathbf{B})$: строки из $R(c_{k1})$ должны быть выше строк из $R(c_{k2})$ в свернутой матрице \mathbf{B} , отношение задается декартовым произведением на множестве $R(\mathbf{B})$: $R_k^r = R(c_{k1}) \times R(c_{k2})$. Аналогично каждый упорядоченный свертываемый набор I_k^{co} (I_k^{co}) порождает иррефлексивное и асимметричное отношение частичного порядка:

$$R^r(I_k^{co}) = \{r_p \times r_q / r_p \in R(c_{ki}), r_q \in R(c_{kj}), i < j\}.$$

Два свертываемых набора I_k^r и I_l^r не пересекаются, если все c_{ki}, c_{lj} различны. Множество $L_k^c = \{I_{k1}^r, I_{k2}^r, \dots, I_{km}^r\}$ попарно не пересекающихся свертываемых наборов столбцов называется неупорядоченным множеством столбцовой свертки. Аналогично определяется упорядоченное множество свертки (УМС) $L_k^{co} = \{I_{k1}^{co}, I_{k2}^{co}, \dots, I_{km}^{co}\}$.

УМС L_k^{co} порождает отношение $R^r(L_k^{co})$ на множестве $R(\mathbf{B})$, представляющее объединение отношений $R^r(I_{ki}^{co})$, порождаемых свертываемыми наборами $I_{ki}^{co} \in L_k^{co}$: $R^r(L_k^{co}) = \cup_i (R^r(I_{ki}^{co}))$.

Отношение $R^r(L_k^{co})$ в общем случае не есть отношение частичного порядка: оно иррефлексивно, асимметрично, не обязательно транзитивно. Его транзитивное замыкание $R^r(L_k^{co})$ иррефлексивно, транзитивно, но может быть асимметричным. В [4, 5] показано, что упорядоченное множество L_k^{co} свертки реализуемо, если порождаемое им транзитивное замыкание $R^r(L_k^{co})$ есть отношение частичного порядка на множестве $R(\mathbf{B})$, т.е. оно асимметрично.

Пример. Упорядоченная свертываемая пара $\langle c_1, c_4 \rangle$ порождает следующее отношение R_{14} на $R(\mathbf{B})$: $R_{14} = R(c_1) \times R(c_4) = \{(r_1, r_3), (r_1, r_4), (r_1, r_5), (r_2, r_3), (r_2, r_4), (r_2, r_5)\}$, а УМС $L^{co} = \{\langle c_1, c_4 \rangle, \langle c_5, c_6 \rangle\}$ порождает отношение $R^r(L^{co}) = R_{14} \cup R(c_5) \times R(c_6) = R_{14} \cup \{(r_4, r_7), (r_4, r_8), (r_4, r_9)\}$. Его транзитивное замыкание $R^r(L^{co})$ не есть отношение частичного порядка, так как оно не асимметрично: $(r_4, r_7) \in R^r(L^{co})$ и $(r_7, r_4) \in R^r(L^{co})$. Следовательно, УМС $L^{co} = \{\langle c_1, c_4 \rangle, \langle c_5, c_6 \rangle\}$ не реализуемо, но УМС $L^{co} = \{\langle c_4, c_1 \rangle, \langle c_5, c_6 \rangle\}$ реализуемо. Аналогично можно доказать, что реализуемо и УМС $L^{co} = \{\langle c_4, c_1 \rangle, \langle c_5, c_6 \rangle, \langle c_8, c_7 \rangle\}$ (ему соответствует столбцовая свертка ПЛМ на рис. 1,б).

Реализуемость УМС L_k^{co} обеспечивает существование такого линейного упорядочения множества строк, что все столбцы одного и того же свертываемого набора $I_{km}^{co} \in L_k^{co}$ могут быть расположены на одной вертикальной линии таким образом, что все столбцы $c_{kmi} \in I_{km}^{co}$ расположены над столбцами $c_{kmj} \in I_{km}^{co}$, если $i < j$. Множество $R(\mathbf{B})$ при этом может быть линейно упорядочено, например, путем топологической сортировки.

Таким образом, реализуемое УМС содержит всю необходимую для свертки матричной структуры информацию, и задача свертки состоит в нахождении реализуемого УМС максимальной мощности. Показано [4], что эта задача NP-трудна. Точное решение задачи одномерной (столбцовой или строчной) свертки разбивается на следующие подзадачи:

- 1) построение множества упорядоченных свертываемых наборов;
- 2) построение всевозможных УМС и проверка их на реализуемость;
- 3) выбор наибольшего реализуемого УМС и линейное упорядочение строк матрицы.

Наиболее сложным, ответственным за экспоненциальную сложность задачи свертки является второй этап. В литературе предложены разные графовые критерии проверки множества свертки на реализуемость [1, 4, 5, 10, 11] (для случая ПЛМ), основанные на поиске циклов в графах специального вида.

Графовые условия реализуемости множества свертки. В [4] вводится граф $G_c = (C, E_c)$ отношения пересечения на множестве C столбцов, в котором ребрами связаны вершины, соответствующие пересекающимся столбцам $E_c = \{e = (c_i, c_j) / R(c_i) \cap R(c_j) \neq \emptyset\}$.

Для проверки на реализуемость множества L^{Co} упорядоченных свертываемых пар строится смешанный граф $G_{L1} = (C, E_c, A)$, где C и E_c определяют граф G_c , а A – множество дуг, задающих свертываемые пары из L^{Co} : $A = (a=(c_i, c_j)) / \langle c_i, c_j \rangle \in L^{Co}$.

Доказано, что УМС L^{Co} реализуемо, если и только если граф G_{L1} не имеет циклов (чередующихся), состоящих из чередующихся ребер и дуг. В [5] смешанный граф G_{L1} определяется для случая многократной свертки – для проверки на реализуемость множества свертываемых наборов.

Пример. На рис. 2,а и 2,б приведены смешанные графы G_{L1} для нереализуемого УМС $L^{Co}_1 = \{\langle c_1, c_4 \rangle, \langle c_5, c_6 \rangle, \langle c_7, c_8 \rangle\}$ и для реализуемого УМС $L^{Co}_2 = \{\langle c_1, c_4 \rangle, \langle c_6, c_5 \rangle, \langle c_7, c_8 \rangle\}$. Во втором случае чередующегося цикла не существует, в первом – существует: $c_1 \rightarrow c_4 - c_7 \rightarrow c_8 - c_5 \rightarrow c_6 - c_1$.

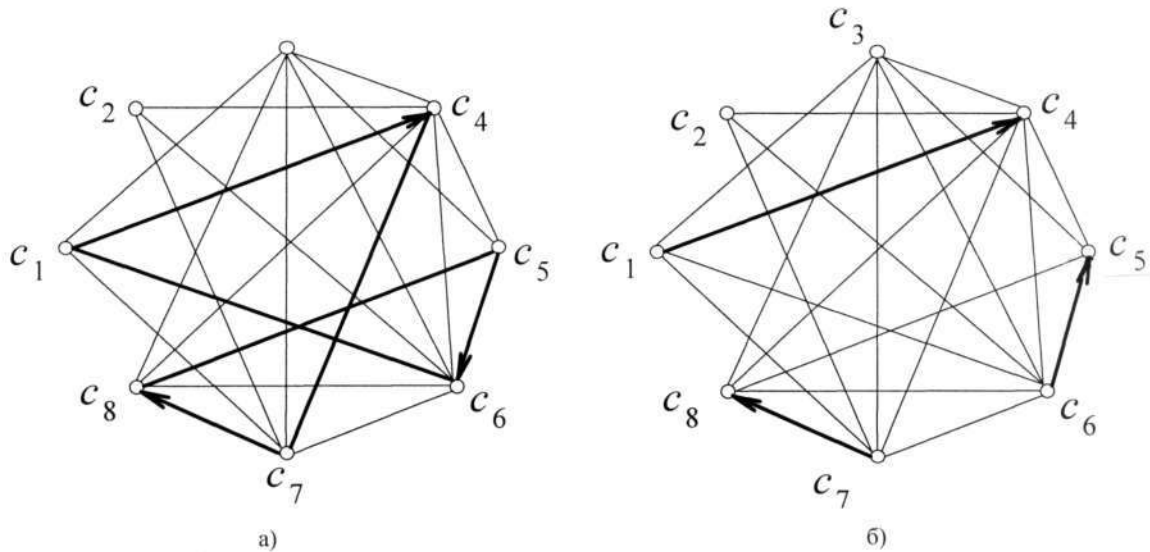


Рис. 2. Графы G_{L1} : а) для нереализуемого УМС $L^{Co}_1 = \{\langle c_1, c_4 \rangle, \langle c_5, c_6 \rangle, \langle c_7, c_8 \rangle\}$; б) для реализуемого УМС $L^{Co}_2 = \{\langle c_1, c_4 \rangle, \langle c_6, c_5 \rangle, \langle c_7, c_8 \rangle\}$

В работе [1] дан несколько другой критерий реализуемости множества свертки L^{Co} : строится двудольный граф $G_{L2} = (R \cup L, A)$, вершины из R и L соответствуют строкам матрицы и свертываемым парам столбцов из УМС L^{Co} . Множество A дуг связывает вершины из R и L , принадлежащие разным долям графа: вершина $r_i \in R$ и пара $l_j = \langle c_{j1}, c_{j2} \rangle \in L^{Co}$ связываются дугой, направленной из r_i в l_j , если $r_i \in R(c_{j1})$, и дугой, направленной из l_j в r_i , если $r_i \in R(c_{j2})$. Множество свертки L^{Co} реализуемо, если и только если граф G_{L2} ациклический.

Пример. На рис. 3,а и 3,б приведены смешанные графы G_{L2} для упомянутых выше нереализуемого L^{Co}_1 и реализуемого L^{Co}_2 УМС. Во втором случае граф G_{L2} является ациклическим, в первом существует цикл $r_1 \rightarrow \langle c_1, c_4 \rangle \rightarrow r_3 \rightarrow \langle c_7, c_8 \rangle \rightarrow r_4 \rightarrow \langle c_5, c_6 \rangle \rightarrow r_1$.

В работе [10] вводится дополнение графа G_c – граф $G_u = (C, E_c^*)$, в котором ребрами связаны вершины, соответствующие непересекающимся столбцам, $E_c^* = \{e=(c_i, c_j) / R(c_i) \cap R(c_j) = \emptyset\}$.

Для проверки на реализуемость УМС L^{Co} упорядоченных свертываемых пар строится смешанный граф $G_{L3} = (C, E_c^*, A)$, в котором дуги из A соединяют свертываемые пары из L^{Co} :

$$A = (a=(c_i, c_j)) / \langle c_i, c_j \rangle \in L^{Co}, \text{ а } E_c^* = E_c^* \setminus A.$$

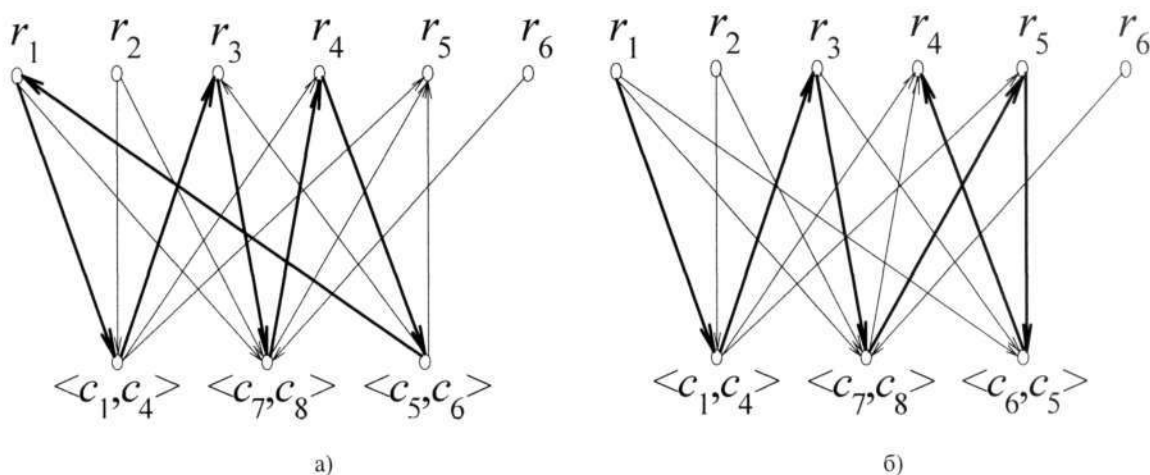


Рис. 3. Графы G_{L2} : а) для нереализуемого УМС $L^{co}_1 = \{<c_1,c_4>, <c_5,c_6>, <c_7,c_8>\}$; б) для реализуемого УМС $L^{co}_2 = \{<c_1,c_4>, <c_6,c_5>, <c_7,c_8>\}$

Показано, что если УМС L^{Co} реализуемо, то оно порождает чередующийся путь в графе G_{L3} , содержащий дуги, соответствующие парам из L^{Co} . Обратное не всегда верно, т.е. чередующийся путь в G_{L3} не гарантирует, что соответствующее множество свертки реализуемо.

Пример. На рис. 4,а и 4,б приведены смешанные графы G_{L3} для нереализуемого $L^{co}_1 = \{<c_1,c_4>, <c_5,c_6>, <c_7,c_8>\}$ УМС. В первом случае в графе G_{L2} чередующегося пути не существует, во втором – такой путь, содержащий дуги, соответствующие всем парам из L^{Co} , существует: $c_6 \rightarrow c_5 - c_7 \rightarrow c_8 - c_1 \rightarrow c_4$.

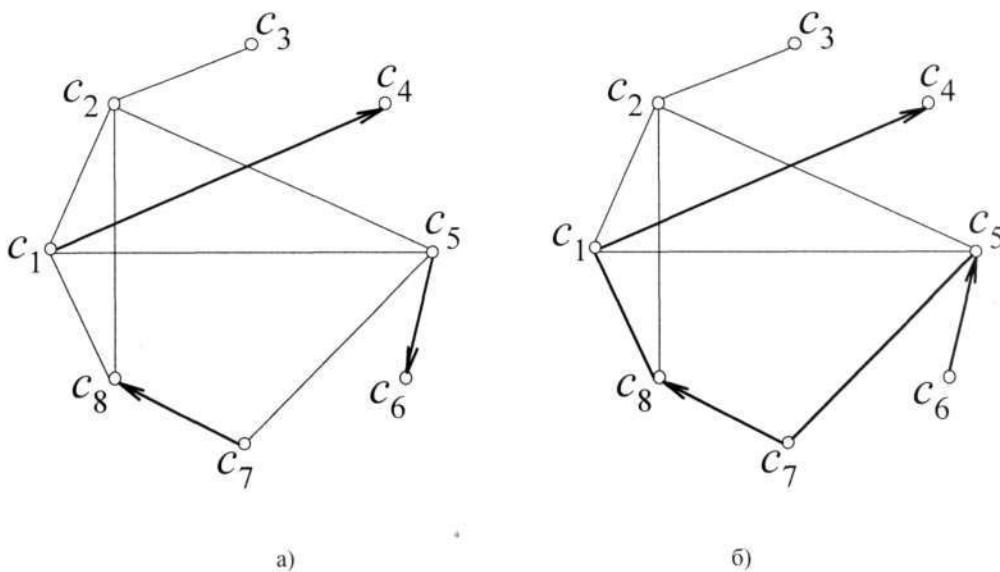


Рис. 4. Графы G_{L3} : а) для нереализуемого УМС $L^{co}_1 = \{<c_1,c_4>, <c_5,c_6>, <c_7,c_8>\}$; б) для реализуемого УМС $L^{co}_2 = \{<c_1,c_4>, <c_6,c_5>, <c_7,c_8>\}$

Четвертый критерий реализуемости множества свертки основан на свойстве выпуклости отношения «образовывать реализуемое множество свертки» на множестве свертываемых пар: любое подмножество реализуемого множества свертки также реализуемо. Это значит, что и любые его две пары также образуют реализуемое множество. Введем отношение совместимости на множестве P свертываемых пар: пары $<c_{i1},c_{i2}>$ и $<c_{j1},c_{j2}>$ совместимы, если множество $\{<c_{i1},c_{i2}>, <c_{j1},c_{j2}>\}$ реализуемо. Отношение совместимости на множестве P иррефлексивно, симметрично, но не обязательно транзитивно. В [11] вводится граф $G_p = (P, E_p)$ отношения со-

вместимости свертываемых пар, в котором ребрами связаны вершины, соответствующие совместимым свертываемым парам $\langle c_{j_1}, c_{j_2} \rangle$.

Проверка реализуемости УМС основана на утверждении: если УМС реализуемо, то оно состоит из попарно совместимых упорядоченных свертываемых пар. Очевидно, если множество L^{Co} упорядоченных свертываемых пар реализуемо, то подграф G_p^L графа $G_p = (P, E_p)$, образованный вершинами, соответствующими парам $\langle c_{j_1}, c_{j_2} \rangle \in L^{Co}$, является полным. Обратное не всегда верно (в силу возможной нетранзитивности отношения совместимости), т.е. полнота подграфа G_p^L не гарантирует, что соответствующее множество свертки является реализуемым.

Пример. На рис. 5 приведен подграф $G_p^L = (P^L, E_p^L)$ графа $G_p = (P, E_p)$ с множеством вершин $P^L = \{\langle c_1, c_4 \rangle, \langle c_4, c_1 \rangle, \langle c_5, c_6 \rangle, \langle c_6, c_5 \rangle, \langle c_7, c_8 \rangle, \langle c_8, c_7 \rangle\}$. В этом графе существует клика, включающая вершины, соответствующие парам из реализуемого УМС $L_1^{Co} = \{\langle c_4, c_1 \rangle, \langle c_5, c_6 \rangle, \langle c_8, c_7 \rangle\}$, а вершины, соответствующие парам из нереализуемого УМС $L_2^{Co} = \{\langle c_1, c_4 \rangle, \langle c_5, c_6 \rangle, \langle c_7, c_8 \rangle\}$, не образуют клику графа.

Граф $G_p = (P, E_p)$, будучи построенным на множестве всех упорядоченных свертываемых пар, может использоваться для поиска кандидатов в максимальные реализуемые множества свертки. Задача сводится к поиску максимальных клик этого графа.

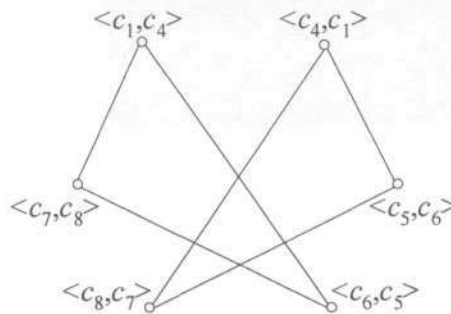


Рис. 5. Подграф графа $G_p = (P, E_p)$ отношения совместимости свертываемых пар

Первые два условия реализуемости множества свертки используются в многочисленных методах свертки матричных структур, основанных на направленном построении УМС – кандидатов на максимальное реализуемое множество свертки (в [7] приведен достаточно представительный обзор методов свертки). Вторые два условия могут непосредственно использоваться для их поиска, если отобразить на графах $G_{L3} = (C, E_c^*, A)$ и $G_p = (P, E_p)$ все возможные упорядоченные свертываемые пары, а затем искать максимальные чередующиеся пути [10] или максимальные клики соответствующих графов [11].

4. Минимизация площади матричных структур методом свертки

Остановимся на подходе [11], основанном на построении графа $G_p = (P, E_p)$ совместимости свертываемых пар и поиске реализуемого множества свертки непосредственно на этом графе. Этот подход привлекателен в силу отработанности методов поиска максимальных клик графа. Задача нахождения максимального реализуемого множества свертки разбивается на следующие подзадачи:

- 1) построение множества P свертываемых пар;
- 2) определение отношения S_p совместимости на множестве P свертываемых пар;
- 3) построение графа $G_p = (P, E_p)$, задающего отношение S_p ;
- 4) определение максимальных клик G_i графа G_p ;
- 5) проверка на реализуемость упорядоченных множеств свертки, порождаемых подграфами G_i , и исключение множеств, не удовлетворяющих этому условию;
- 6) выбор подграфа G_i (и соответствующего ему УМС) с максимальным числом вершин.

Определение отношения совместимости на множестве свертываемых пар. Неупорядоченные свертываемые пары c_{i_1}, c_{i_2} и c_{j_1}, c_{j_2} совместимы, если совместимы упорядоченные свертываемые пары, полученные из исходных при каком-то их упорядочении. Рассмотренные выше критерии реализуемости УМС трудоемки и требуют упорядочения свертываемых пар c_{k_1}, c_{k_2}

$\langle c_{k1}, c_{k2} \rangle$ или $\langle c_{k2}, c_{k1} \rangle$). Для проверки на реализуемость множества свертки, состоящего из двух свертываемых пар, возможен более простой критерий.

В [12] показано, что свертываемые пары c_{i1}, c_{i2} и c_{j1}, c_{j2} не совместимы, если и только если выполняются следующие четыре условия:

$$\begin{aligned} R_1 &= R(c_{i1}) \cap R(c_{j2}) \neq \emptyset; & R_2 &= R(c_{i2}) \cap R(c_{j1}) \neq \emptyset; \\ R_3 &= R(c_{i1}) \cap R(c_{j1}) \neq \emptyset; & R_4 &= R(c_{i2}) \cap R(c_{j2}) \neq \emptyset. \end{aligned}$$

Из этого утверждения вытекает метод анализа свертываемых пар c_{i1}, c_{i2} и c_{j1}, c_{j2} на совместимость. Он состоит в нахождении четырех пересечений R_1, R_2, R_3, R_4 , в зависимости от их мощностей делаются следующие заключения:

- если $R_1 = \emptyset$ или $R_2 = \emptyset$, совместимы две пары: $\{\langle c_{i1}, c_{i2} \rangle, \langle c_{j1}, c_{j2} \rangle\}$ и $\{\langle c_{i2}, c_{i1} \rangle, \langle c_{j2}, c_{j1} \rangle\}$;
- если $R_3 = \emptyset$ или $R_4 = \emptyset$, совместимы две пары: $\{\langle c_{i1}, c_{i2} \rangle, \langle c_{j2}, c_{j1} \rangle\}$ и $\{\langle c_{i2}, c_{i1} \rangle, \langle c_{j1}, c_{j2} \rangle\}$.

Пример. Свертываемые пары c_1, c_4 и c_5, c_6 столбцов матрицы \mathbf{B} порождают две совместимые пары $\{\langle c_1, c_4 \rangle, \langle c_6, c_5 \rangle\}$ и $\{\langle c_4, c_1 \rangle, \langle c_5, c_6 \rangle\}$, так как

$$\begin{aligned} R_1 &= R(c_1) \cap R(c_6) = \{r_1\} \neq \emptyset; & R_2 &= R(c_4) \cap R(c_5) = \{r_4\} \neq \emptyset; \\ R_3 &= R(c_1) \cap R(c_5) = \emptyset; & R_4 &= R(c_4) \cap R(c_6) = \{r_3, r_5\} \neq \emptyset. \end{aligned}$$

В [12] предлагается для сокращения размера графа $G_p = (P, E_p)$ ставить в соответствие его вершинам неупорядоченные свертываемые пары. Это сокращает число вершин графа G_p вдвое, а число возможных ребер – до $2n(n-1)$ раз ($n = |P|$).

Возможные упорядочения свертываемых пар учитываются путем пометки ребер графа. Метка, приписанная ребру графа, указывает, какие из возможных упорядочений свертываемых пар, соответствующих связываемым этим ребром вершинам, порождают совместимые упорядоченные пары. Ребро графа не помечается, если пары совместимы при любых упорядочениях их столбцов, и помечается 1 или 0, если пары совместимы при одинаковом или различном упорядочении их столбцов. Таким образом, вводится в рассмотрение неориентированный граф $G=(P, E_0, E_1, E_2)$ с множествами P вершин и $E_0 \cup E_1 \cup E_2 = E_p$ ребер. Ребра $e_i \in E_0$ и $e_j \in E_1$ имеют пометки 0 и 1 соответственно, ребра $e_l \in E_2$ не помечены.

Пример. Как было показано выше, для рассматриваемой матрицы \mathbf{B} существует семь свертываемых пар столбцов. На рис. 6, а приведен граф $G=(P, E_0, E_1, E_2)$ совместимости свертываемых пар. Пометки ребер из E_2 опущены.

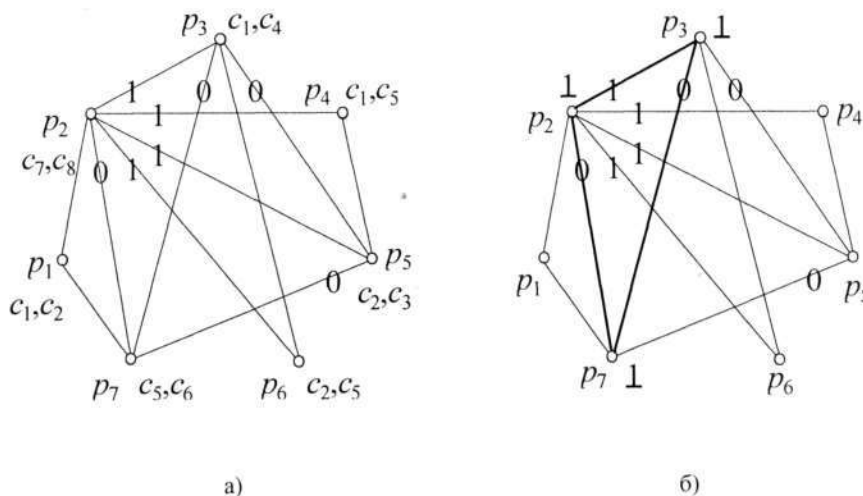


Рис. 6. Граф $G=(P, E_0, E_1, E_2)$: а) с помеченными ребрами; б) с помеченными ребрами и вершинами

Упорядочение столбцов внутри пары свертки в графовой интерпретации задается пометкой соответствующей вершины графа: единицей, если в паре сохраняется прямой порядок, или нулем, если принимается обратный порядок. Соответственно вершины, связанные ребром с пометкой 1, должны быть помечены одинаково, вершины, связанные ребром с пометкой 0, должны иметь разные метки. Функция $\varphi(p_k)$ называется *правильной пометкой вершин* $p_k \in P$ графа, если для всех вершин $(p_k, p_i) \in E_0$ имеет место $\varphi(p_k) \neq \varphi(p_i)$, а для всех $(p_k, p_i) \in E_1$ имеет место $\varphi(p_k) = \varphi(p_i)$. Если $(p_k, p_i) \in E_2$, на пометку вершин p_k и p_i не накладывается ограничений.

Поиск максимального реализуемого множества свертки. Показано [12], что поиск наибольшего реализуемого множества свертки сводится к поиску наибольшей клики графа $G=(P, E_0, E_1, E_2)$, допускающей правильную пометку ее вершин. Наибольшая клика графа $G=(V, E_0, E_1, E_2)$ отыскивается в [13] методом сокращенного обхода дерева поиска. Построение очередной клики в соответствии с используемой «жадной» стратегией связывается с вершиной $p_i \in P$ с наибольшей степенью. Интерес представляют клики с большим числом вершин, чем у ранее найденной. Верхней границей мощности искомого клики является число q , такое, что не менее q вершин графа G имеют степень q . Если клика такой мощности будет найдена, то она и является наибольшей. Анализ клик на существование правильной пометки их вершин упрощается при использовании следующего утверждения: если полный граф $G=(V, E_0, E_1, E_2)$ не допускает правильную пометку вершин, то в процессе любой пометки его ребер встретится цикл длины три, в котором все ребра помечены и одно или три ребра имеют пометку 0.

Пример. Верхняя граница мощности наибольшей клики графа G на рис. 6,а равна четырем, но такой клики, допускающей правильную пометку ее вершин, не находится. Первая из найденных максимальных клик, мощность которой равна трем, $\{p_2, p_7, p_3\}$ допускает правильную пометку $(1, 1, 0)$ вершин и не нарушает условие реализуемости порождаемых ею УМС $L_1^{Co} = \{<c_7, c_8>, <c_1, c_4>, <c_6, c_5>\}$ и $L_2^{Co} = \{<c_8, c_7>, <c_4, c_1>, <c_5, c_6>\}$. На рис. 6, б приведена пометка вершин найденной клики, а на рис. 1, б – порождаемая УМС L_2^{Co} свертка ПЛИМ.

Список литературы

1. Ульман Дж. Вычислительные аспекты СБИС. – М.: Радио и связь, 1990. – 480 с.
2. Бибило П.Н. Кремниевая компиляция заказных СБИС. – Мн.: Ин-т техн. кибернетики АН Беларуси, 1996. – 268 с.
3. Devadas S., Newton A.R. Topological optimization of multiple-level array logic // IEEE Trans. on Comput.-Aided Design. – 1987. – V. CAD-6. – No 6. – P. 915–941.
4. Hatchel G.D., Newton A.R., Sangiovanni-Vincentelli A. L. An algorithm for optimal PLA-folding // IEEE Trans. Comput.-Aided Des. Integr. Circuits and Syst. – 1982. – V. CAD-1. – No 2. – P. 63–77.
5. DeMicheli G., Sangiovanni-Vincentelli A.L. Multiple Constrained Folding of Programmable Logic Arrays: Theory and Applications // IEEE Trans. Computer-Aided Design Integr. Circuits and Syst. – 1983. – V. CAD-2. – No 2. – P. 151–167.
6. Черемисинова Л.Д. К реализации системы ДНФ на матрицах Вайнбергера // Автоматизация проектирования дискретных систем. – Мн.: Ин-т техн. кибернетики АН Беларуси, 1995. – С. 48–56.
7. Черемисинова Л.Д. Минимизация площади матричных структур методом свертки // Идентификация образов. – Мн.: Ин-т техн. кибернетики НАН Беларуси, 2001. – С. 145–155.
8. Черемисинова Л.Д. Минимизация площади матрицы Вайнбергера, реализующей систему ДНФ // Управляющие системы и машины. – 1999. – № 2. – С. 39–46.
9. Бибило П.Н. Минимизация площади регулярных МОП-схем с последовательным соединением транзисторов // Микроэлектроника. – 1995. – № 6. – С. 408–411.
10. Hwang S.Y., Dutton R.W. and Blank T. A best-first search algorithm for optimal PLA folding // IEEE Trans.on Comput.-Aided Design. – 1986. – V. CAD-6. – No 5. – P. 433–442.
11. Lecky J.E., Murphy O.J. and Absher R.G. Graph theoretic algorithms for the PLA folding problem // IEEE Trans.on Comput.-Aided Design. – 1989. – V. 8. – N 9. – P. 1014–1021.
12. Cheremisinova L.D. Some results in optimal PLA folding // Proc. of the Third Int. Conf. on Computer-Aided Design of Discrete Devices (CAD DD'99), Minsk, Nov.10–12, 1999. – Minsk: NAS of Belarus, 1999. – Vol.1. – P. 59–64.

13. Черемисинова Л.Д. Поиск наибольшей клики специального вида графа отношения совместимости // Докл. Первого науч. семинара «Танаевские чтения». – Мн.: ОИПИ НАН Беларуси, 2003. – С. 138–142.

Поступила 30.01.04

*Объединенный институт проблем
информатики НАН Беларуси,
Минск, Сурганова, 6
e-mail: cld@newman.bas-net.by*

L.D. Cheremisinova

AREA MINIMIZATION OF REGULAR ARRAY STRUCTURES OF CUSTOM VLSI

The problem under consideration is area minimization of two-dimensional array-based structures of custom VLSI. Approaches to solving two tied problems are presented. The first problem is logic minimization of functional representation of the control logic realized by array-based structure that is done during logic design stage. The second one is topological compaction aimed at minimizing the chip area occupied by the array structure and it is solved at topological level before layout generation.