

# ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ ОПТИМИЗАЦИИ ПРЕДСТАВЛЕНИЙ СИСТЕМ БУЛЕВЫХ ФУНКЦИЙ НА ОСНОВЕ БЛОЧНОГО РАЗБИЕНИЯ И РАЗЛОЖЕНИЯ ШЕННОНА

Бибило П. Н., Кириенко Н. А., Ланкевич Ю. Ю.

Объединённый институт проблем информатики Национальной академии наук Беларуси

Минск, Республика Беларусь

E-mail: kir@newman.bas-net.by

*Рассматриваются результаты экспериментального исследования эффективности оптимизации представлений систем булевых функций при синтезе комбинационных схем из библиотечных КМОП-элементов. Процедуры оптимизации включают алгоритмы разбиения систем функций на подсистемы и алгоритмы оптимизации многоуровневых представлений функций на основе разложения Шеннона.*

## ВВЕДЕНИЕ

Процесс синтеза логических устройств содержит два этапа: технологически независимую оптимизацию и технологическое отображение. На этапе технологического отображения оптимизированные представления описаний логических устройств покрываются описаниями базисных логических элементов заданной библиотеки проектирования.

Эффективными методами технологически независимой оптимизации являются методы, базирующиеся на разложении Шеннона и позволяющие получать оптимизированные BDD (Binary Decision Diagram) [1], а также на разбиении исходного функционального описания систем полностью определенных булевых функций на блоки [2].

В настоящей работе приводятся результаты экспериментального исследования эффективности применения при синтезе логических схем из КМОП-элементов упомянутых выше методов.

## I. ОПТИМИЗАЦИЯ МНОГУРОВНЕВЫХ ПРЕДСТАВЛЕНИЙ, ОСНОВАННАЯ НА РАЗЛОЖЕНИИ ШЕННОНА

Разложением Шеннона полностью определенной булевой функции  $f(x)$  по переменной  $x_i$  называется представление

$$f(x) = \bar{x}_i f_0 \vee x_i f_1.$$

Подфункции  $f_0 = f(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n)$  и  $f_1 = f(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n)$  могут быть разложены по одной из переменных  $x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n$ . Процесс разложения подфункций заканчивается, когда все  $n$  переменных будут использованы для разложения, либо когда все подфункции вырождаются до констант 0, 1.

Под BDDI (Binary Decision Diagram with Inverse Cofactors) в данной работе понимается ориентированный ациклический граф, задающий последовательные разложения Шеннона булевой функции  $f(x_1, \dots, x_n)$  по всем ее переменным при заданном порядке (перестановке) переменных, по которым проводятся разложения, с

учетом нахождения инверсных коэффициентов [3]. Возможно построение BDDI двух видов - совместных и отдельных. Совместными BDDI будем называть такие BDDI, которые построены по общей для всех компонентных функций перестановке переменных. Если BDDI для каждой из компонентных функций системы строятся независимо, т.е. так, что каждая из функций разлагается по своей перестановке переменных, то такие BDDI будем называть отдельными.

## II. РАЗБИЕНИЕ МНОГУРОВНЕВЫХ ПРЕДСТАВЛЕНИЙ СИСТЕМ БУЛЕВЫХ ФУНКЦИЙ НА ПОДСИСТЕМЫ

Процедуру разбиения рассмотрим на графовой модели представления логической сети. Ограничениями на сложность каждого из блоков являются число  $r$  входных и число  $q$  выходных переменных блока. Текущим представлением логической сети является взвешенный ориентированный граф  $G = (V, E)$ . Вершине  $v_i \in V$  соответствует логическое уравнение. Две вершины  $v_i, v_j$  графа  $G$  связывает дуга  $e_{ij}$ , принадлежащая множеству дуг  $E$  (дуга исходит из вершины  $v_i$  и заходит в вершину  $v_j$ ), в том и только в том случае, когда выходная переменная уравнения, соответствующего  $v_i$ , является входной уравнения, соответствующего  $v_j$ .

Задачу разбиения исходной системы логических уравнений на подсистемы можно представить как задачу разбиения множества  $V$  вершин графа  $G$  на множество непересекающихся подмножеств вершин  $\{V_1, \dots, V_p\}$ , таких, что  $V_k \subset V, V_k \neq \emptyset, V_k \cap V_l = \emptyset, \cup V_k = V$ , для  $k, l \in \{1, \dots, p\}, k \neq l$ . Вершина графа может иметь числовые характеристики, которыми являются числа входных, выходных переменных подсистемы логических уравнений, соответствующих этой вершине.

В настоящем эксперименте исследуется процедура разбиения графа, представляющего систему логических уравнений, на минимальное число  $p$  блоков  $V_k$  при выполнении заданных ограничений на числа входных ( $r$ ) и выходных

( $q$ ) переменных блоков  $V_k$ . В основе процедуры лежит последовательный алгоритм формирования блоков, содержащий этапы выбора начальной вершины для блока, построения множества смежных вершин для блока, последовательного подключения вершин в блок из множества смежных вершин, проверки ограничений на числа входных и выходных переменных. Особенностью алгоритма является выполнение процедуры элиминации внутренних переменных блока с целью их сокращения.

### III. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТАЛЬНОГО ИССЛЕДОВАНИЯ

Оптимизационные процедуры выполнялись на внутреннем языке SF в экспериментальной системе проектирования логических устройств FLC [4]. С помощью вышеописанных процедур выполнялась оптимизация исходных описаний логических схем в виде системы логических уравнений (формат LOG языка SF). Полученные оптимизированные описания конвертировались в VHDL-описания и подавались на вход синтезатора LeonardoSpectrum, который выполнял синтез схем. Процедура построения совместных BDDI выполнялась программой BDD\_Builder [3]. Для блочного разбиения многоуровневых представлений использовалась процедура PartCLTT системы FLC. В качестве примеров использовались многоуровневые описания комбинационной логики из набора Berkeley PLA test set [5]. Результаты эксперимента представлены в таблице 1.

Для каждого примера выполнялся синтез с помощью системы LeonardoSpectrum трех вариантов схем.

Вариант 1. Синтез по исходному описанию комбинационной схемы.

Вариант 2. Оптимизация исходного описания с помощью программы BDD\_Builder, синтез по оптимизированному описанию.

Вариант 3. Разбиение исходного описания схемы на блоки с заданными ограничениями на

число входов и выходов, оптимизация каждого блока с помощью программы BDD\_Builder, синтез по оптимизированному описанию.

Для каждого тестового примера приведены следующие параметры: числа  $n$  и  $m$  аргументов и функций системы; величины  $S_1, S_2, S_3$  – площади синтезированных схем по соответствующему варианту, число блоков разбиения  $p$ , ограничения на параметры  $r, q$  блоков. Площадь схемы оценивается в числе транзисторов.

### ЗАКЛЮЧЕНИЕ

Результаты эксперимента позволили установить, что достаточно эффективной процедурой логической оптимизации является оптимизация на основе совместных BDDI. Предварительное разбиение системы функций на подсистемы и последующее построение совместных BDDI позволяет уменьшить площадь схемы (в 8 случаях из 11). При этом уменьшение параметров блоков, на которые разбиваются логические уравнения, позволяет добиваться лучших результатов по площади.

1. Bryant, R. E. Ordered Binary Decision Diagrams / R. E. Bryant, C. Meinel // Logic synthesis and verification /ed. by S. Hassoun, T. Sasao, R. K. Brayton. – [S.1]: Kluwer Academic Publishers. – 2002. – P. 285 – 307.
2. Бибило, П. Н. Оптимизационные преобразования логической схемы на основе блочного разбиения / П. Н. Бибило, Н. А. Кириенко // Информатика. – 2009. – № 3. – С. 5 – 15.
3. Бибило, П. Н. Использование полиномов Жегалкина при минимизации многоуровневых представлений систем булевых функций на основе разложения Шеннона / П. Н. Бибило, Ю. Ю. Ланкевич // Программная инженерия. – 2017. – № 3. – С. 369 – 384.
4. Бибило, П. Н. Логическое проектирование дискретных устройств с использованием продукционно-фреймовой модели представления знаний / П. Н. Бибило, В. И. Романов // Минск : Белорусская наука. – 2011. – 279 с.
5. Berkeley PLA test set [Electronic resource] / Mode of access: <http://www1.cs.columbia.edu/cs6861/sis/espresso-examples/>. – Date of access: 29.06.2017.

Таблица 1 – Результаты экспериментального исследования эффективности оптимизации

Пример	$n, m$	$S_1$	$S_2$	$S_3$	Число блоков $p$	Параметры $r, q$
Apex6	135, 94	<b>1830</b>	2110	1858	25	21, 100
C8	28, 18	<b>312</b>	324	318	9	10, 100
Cht	47, 36	680	670	<b>652</b>	7	10, 100
Count	35, 168	256	256	256	8	8, 100
Dalu	75, 16	1834	1396	<b>1286</b>	23	30, 100
x3	135, 99	3462	3788	<b>3154</b>	87	8, 100
term1	34, 10	1044	3086	<b>876</b>	40	8, 100
x4	94, 71	1544	2892	<b>1482</b>	44	10, 100
cordic	23, 2	140	138	<b>138</b>	13	4, 100
ttt2	24, 21	698	766	<b>648</b>	30	10, 100
example2	85, 63	850	996	<b>778</b>	11	15, 100