

ПРОЕКТИРОВАНИЕ НА ПЛИС УСТРОЙСТВ ЦИФРОВОЙ ОБРАБОТКИ СИГНАЛОВ С ПОМОЩЬЮ ПАКЕТА ZUBR

Соловьев В.В., Климович А.

Белостокский технический университет (Польша)

1. Введение

При разработке на ПЛИС сложных систем цифровой обработки сигналов проектировщику часто приходится сталкиваться с проблемой необходимости проектирования оригинальных цифровых устройств: комбинационных схем, конечных автоматов, устройств логического управления и др. Использование для этого средств, имеющихся в промышленных пакетах автоматизированного проектирования, часто бывает не эффективно.

На практике для эффективной реализации оригинальных цифровых устройств проектировщики пользуются научно-исследовательскими пакетами логического проектирования. Главное отличие научно-исследовательских пакетов от промышленных систем автоматизированного проектирования заключается в том, что в научно-исследовательских пакетах реализуются оригинальные методы проектирования, а также используются новые подходы (модели и методы) к проектированию отдельных функциональных блоков цифровых систем.

В настоящей статье описывается научно-исследовательский пакет ZUBR логического проектирования цифровых систем на основе ПЛИС. Пакет ZUBR разработан усилиями сотрудников и студентов двух университетов: Белорусского государственного университета информатики и радиоэлектроники (БГУИР, бывший МРТИ, г. Минск, Республика Беларусь) и Белостокского технического университета (г. Белосток, Республика Польша).

2. Назначение

Пакет ZUBR предназначен для автоматизации решения задач логического проектирования цифровых систем на основе ПЛИС. Пакет ZUBR позволяет значительно снизить стоимость и повысить быстродействие ряда цифровых устройств (комбинационных схем, конечных автоматов, устройств логического управления и др.) и цифровой системы в целом за счет эффективного использования архитектурных возможностей ПЛИС, а также учета системных требований на самых ранних этапах проектирования. Кроме того, применение пакета ZUBR позволяет уменьшить время проектирования цифровых устройств, так как после применения пакета ZUBR значительно сокращается время логического синтеза в промышленном пакете. Методы и алгоритмы, реализованные в пакете ZUBR, ориентированы на решение нетрадиционных задач или известных задач нетрадиционными подходами.

С точки зрения практического использования пакет ZUBR может рассматриваться как надстройка над промышленными пакетами автоматизированного проектирования на основе ПЛИС, например, фирм Altera, Xilinx, Mentor Graphics и др. Использование пакета ZUBR предусматривает тесное взаимодействие с известными промышленными пакетами. Проектирование стандартных функциональных узлов и выполнение основных этапов проектирования (ввод данных, компиляция, временной анализ, моделирование и др.) осуществляется с помощью промышленного пакета, а проектирование сложных функциональных узлов - с помощью пакета ZUBR.

Для связи пакета ZUBR с промышленными пакетами используются *конвертеры*, преобразующие соответствующее представление исходных данных и результатов синтеза к необходимому формату.

С помощью пакета ZUBR можно решать следующие проектные задачи:

- выполнять логический синтез оригинальных комбинационных схем, конечных автоматов, устройств логического управления, а также других нестандартных функциональных узлов цифровых систем;
- выполнять верификацию синтезированных комбинационных схем, конечных автоматов и устройств логического управления;
- представлять полученные результаты на языках проектирования, используемых в промышленных пакетах.

Кроме автоматизированного решения перечисленных основных проектных задач, пакет ZUBR также позволяет:

- на основании системных требований и архитектурных возможностей ПЛИС автоматически выбирать наиболее подходящий метод синтеза;
- на основании системных требований и архитектурных возможностей ПЛИС автоматически выбирать наиболее подходящие модели конечных автоматов;
- конвертировать описания комбинационных схем и конечных автоматов из форматов системы SIS на языки проектирования и обратно;
- конвертировать описания комбинационных схем и конечных автоматов из одних языков проектирования в другие;
- автоматически строить эталонные тестовые последовательности для верификации цифровых устройств на последующих этапах проектирования.

Более подробную информацию о пакете ZUBR можно найти на странице <http://aragorn.pb.bialystok.pl/~zubr/>.

3. Структура

Все программы, составляющие пакет ZUBR, можно разделить на пять групп:

- системные;
- реализующие методы синтеза;
- реализующие методы верификации результатов синтеза;
- конвертеры;
- другие программы проектирования.

Группу системных программ составляют текстовый редактор, менеджер файлов и менеджер помощи. В качестве текстового редактора используются стандартные редакторы операционной системы Windows Notepad или WordPad (для просмотра больших файлов). Менеджер файлов обеспечивает стандартные операции при работе с файлами (открытие, создание, сохранение и др.). Менеджер помощи упрощает использование пакета ZUBR.

Методы синтеза, реализованные в пакете ZUBR, можно разделить на следующие группы:

- методы синтеза комбинационных схем;
- методы синтеза конечных автоматов;
- методы синтеза микропрограммных автоматов;
- методы синтеза нестандартных функциональных узлов;
- другие методы синтеза цифровых систем.

Отметим, что методы синтеза пакета ZUBR постоянно обновляются, включаются новые методы или даже целые группы методов, исключаются отдельные малоэффективные методы или методы, не удовлетворяющие требованиям пользовательского интерфейса.

Методы верификации, реализованные в пакете ZUBR, можно разделить на следующие группы:

- методы верификации комбинационных схем;
- методы верификации конечных автоматов;
- методы верификации микропрограммных автоматов;
- методы построения тестовых последовательностей.

Данные методы предназначены для проверки результатов синтеза соответствующих устройств. Отдельные методы верификации постоянно соединены с методами синтеза и при выполнении синтеза верификация результатов осуществляется автоматически. По желанию пользователя программы верификации могут также формировать файлы, содержащие тестовые последовательности для проверки устройства на последующих этапах проектирования.

Конвертеры обеспечивают информационную связь пакета ZUBR с промышленными системами автоматизированного проектирования. В качестве входных форматов в пакете ZUBR были выбраны форматы системы SIS [2]: ESPRESSO для описания комбинационных схем [2] и KISS2 для описания конечных автоматов [2]. Связь пакета ZUBR с промышленными пакетами осуществляется с помощью конвертеров, преобразующих описание цифровых устройств в формате системы SIS на языки проектирования VHDL, Verilog, Abel и AHDL, а также обратно, с языков проектирования VHDL, Verilog, Abel и AHDL на форматы системы SIS.

Группа дополнительных программ проектирования в значительной степени облегчает использование методов синтеза, реализованных в пакете ZUBR. Эти программы позволяют эффективно применять многочисленные методы синтеза пакета ZUBR без глубокого их изучения. Данная группа включает следующие программы:

- выбора наиболее эффективного метода синтеза для построения комбинационных схем, конечных автоматов и устройств логического управления (микропрограммных автоматов);
- выбора наиболее эффективных моделей конечных автоматов;
- автоматического определения параметров ПЛИС.

Программа определения параметров ПЛИС позволяет пользователю задавать параметры ПЛИС путем выбора конкретной микросхемы из определенного семейства. Кроме того, пользователь может самостоятельно задать параметры оригинальной ПЛИС, например, при исследовании нового метода синтеза.

4. Методы синтеза комбинационных схем

Традиционно методы синтеза комбинационных схем на ПЛИС включают следующие последовательно выполняемые этапы:

- строится достаточная общая модель ПЛИС, допускающая применение классических методов синтеза;
- для данной модели решаются классические задачи синтеза (минимизации, декомпозиции, факторизации и др.), причем таких задач может быть несколько;
- выполняется отображение полученной логической схемы на структуру реальной ПЛИС.

Недостатки традиционного подхода заключаются в следующем:

- слишком общая модель ПЛИС, которая не позволяет учитывать все архитектурные особенности реальной ПЛИС;

- классические задачи синтеза не позволяют учитывать большинство архитектурных особенностей ПЛИС, поэтому даже получение точных решений для отдельных задач приводит к результатам синтеза, далеким от оптимальных;

- достаточно трудно решаются задачи отображения логической схемы на структуру ПЛИС, поэтому часто на этом этапе необходимо заново решать задачи минимизации, декомпозиции и др.

Основные отличия предлагаемого подхода к синтезу комбинационных схем на ПЛИС от традиционного заключаются в следующем:

- определяется не одна, а три модели ПЛИС: универсальные PAL (Programmable Logic Array), “классические” PAL и CPLD (Complex Programmable Logic Device), причем каждая модель позволяет наиболее полно учитывать архитектурные особенности соответствующего класса ПЛИС;

- осуществляется предварительное преобразование исходной системы булевых функций для наиболее эффективного применения соответствующего метода синтеза;

- задача отображения синтезируемой комбинационной схемы на структуру ПЛИС частично или полностью решается в процессе синтеза.

Снижение стоимости реализации комбинационных схем в предлагаемых методах синтеза достигается за счет эффективного использования:

- возможности программирования логического уровня выходных сигналов ПЛИС, что позволяет реализовать одно из наиболее простых представлений каждой функции: прямое или инверсное;

- внутренних цепей обратных связей ПЛИС, при этом значения уже реализованных функций, а также их инверсий могут использоваться в качестве факторов для реализации других функций;

- макроячеек ПЛИС с двумя обратными связями - для одновременного приема значений входных переменных и реализации промежуточных функций;

- возможности подсоединения различного числа промежуточных шин к различным макроячейкам ПЛИС и др.

В данной работе рассматриваются следующие методы синтеза:

- одноуровневых комбинационных схем (метод М1);
- одноуровневых комбинационных схем с использованием монтажного объединения выходов ПЛИС по ИЛИ (метод М2);

- двухуровневых комбинационных схем (методы М3 и М4);

- многоуровневых комбинационных схем с использованием внутренних цепей обратных связей ПЛИС (метод М5);

- сложных многоуровневых комбинационных схем (метод М6).

Отметим, что все рассматриваемые методы включают этап предварительного преобразования исходной системы булевых функций (СБФ) Y и формирования множества Y^* реализуемых функций для наиболее эффективного применения соответствующего метода синтеза.

Суть метода М1 синтеза одноуровневой комбинационной схемы заключается в разбиении множества Y^* на минимальное число T подмножеств Y_1, \dots, Y_T таким образом, чтобы выполнялись необходимые и достаточные условия реализации на одной PAL (функциональном блоке CPLD) всех функций подмножества $Y_t, t = \overline{1, T}$. В последующем функции каждого подмножества $Y_t, t = \overline{1, T}$, реализуются на отдельной PAL. При назначении функций для реализации внешним выводам PAL сложные функции назначаются выходным макроячейкам с большим числом промежуточных шин, а простые функции - макроячейкам с меньшим числом промежуточных шин. В случае реализации инверсного представления функции необходимое значение образуется путем программирования логического уровня выходного сигнала.

Положительные качества метода М1 заключаются в следующем:

- строятся комбинационные схемы максимального быстродействия, ограничиваемого только временем прохождения сигналов со входов на выходы ПЛИС;

- все выходные сигналы имеют одинаковое время формирования;

- метод позволяет применять раздельную минимизацию булевых функций, качество которой выше, чем совместная минимизация;

- метод применим в цифровых системах как с прямой, так и с инверсной логикой;

- для реализации СБФ требуется минимальное число макроячеек ПЛИС, равное числу реализуемых функций.

Однако метод М1 применим только к относительно простым СБФ.

Суть метода М2 синтеза одноуровневых комбинационных схем с объединением выходов ПЛИС по ИЛИ заключается в формировании минимального числа T подмножеств V_1, \dots, V_T частей функций множества Y^* таким образом, чтобы выполнялись необходимые и достаточные условия реализации на одной PAL (функциональном блоке CPLD) всех частей каждого подмножества $V_t, t = \overline{1, T}$. Последняя задача сводится к задаче покрытия булевой матрицы минорами ограниченной размерности. В последующем части функций

части функций каждого подмножества $V_i, i = \overline{1, T}$, реализуются на отдельной PAL. На заключительном этапе синтеза для каждой функции \bar{y}_i выходы PAL, на которых реализовано множество $Z(\bar{y}_i)$ частей функции $\bar{y}_i, i = \overline{1, N}$, объединяются с помощью монтажного ИЛИ.

Положительными качествами метода М2 являются следующие:

- строятся комбинационные схемы высокого быстродействия;
- время формирования различных выходных сигналов отличается незначительно: на время задержки на монтажном соединении выходов;
- метод позволяет применять раздельную минимизацию СБФ;
- значительно более широкая область использования, по сравнению с методом М1.

В то же время к ограничениям на использование метода М2 относится следующее:

- ПЛИС должны быть изготовлены по технологии с открытым коллектором или поддерживать программируемую опцию “открытый сток” (open-drain);
- метод применим в системах с инверсной логикой;
- инверсное представление функций может быть значительно сложнее их прямого представления;
- метод предусматривает последующее монтажное соединение выходов и установку для каждого монтажного соединения “подтягивающего” (pull-up) резистора, что может значительно усложнить этап конструкторского проектирования;

Многие недостатки метода М2 можно устранить, если монтажное соединение выходов выполнять с помощью второго уровня ПЛИС. Метод синтеза двухуровневых комбинационных схем М3 состоит из трех этапов. На первом этапе определяется множество Y^* реализуемых функций; на втором этапе выполняется второй этап метода М2; на третьем этапе части функций объединяются по ИЛИ с помощью второго уровня ПЛИС. Поскольку в двухуровневой схеме каждая функция $y_j, y_j \in Y$, реализуется на отдельной макроячейке ПЛИС, то для каждой функции можно выбирать прямое или инверсное представление функции, а требуемый вид на выходе определять путем программирования логического уровня выходного сигнала выходной макроячейки ПЛИС.

Существенным недостатком метода М3 является различное время формирования значений выходных функций, реализуемых на первом и втором уровнях схемы. В методе М4 для устранения указанного недостатка все выходные функции реализуются на втором уровне схемы. Метод М4 совпадает с методом М3 за исключением того, что на третьем этапе синтеза в множество частей функций Z включаются все функции, в том числе и реализованные на PAL первого уровня.

К положительным качествам методов М3 и М4 синтеза двухуровневых комбинационных схем относятся:

- методы применимы, в отличие от метода М2, в цифровых системах как с отрицательной, так и с положительной логикой, а также для любых ПЛИС (не требуется наличия программируемых опций open-drain и pull-up);

- метод М4 обеспечивает одинаковое время формирования выходных сигналов.

К недостаткам методов М3 и М4 относятся:

- снижение быстродействия в два раза, по сравнению с методами М1 и М2;
- увеличение стоимости реализации, по сравнению с методом М2, особенно для метода М4;
- различное время формирования значений выходных функций при использовании метода М3.

Особенностью метода М5 является то, что в нем широко используются уже реализованные функции или части функций, а также их инверсии, в качестве фактор-функций для реализации других функций. Метод М5 во многом подобен методу М3 и также включает три этапа. Отличия первого этапа заключаются в том, что выполняется не раздельная, а совместная минимизация множества функций $Y \cup \bar{Y}$. Из двух функций y_i и \bar{y}_i при выполнении необходимых и достаточных условий реализации в множество Y^* выбирается функция, которая наиболее эффективна в качестве фактор-функции при реализации других функций. На втором этапе согласно метода синтеза М2 формируется подмножество частей функций V_1, \dots, V_T . Отличие заключается в том, что если некоторая реализованная функция (часть функции) или ее инверсия является фактор-функцией для других функций, то вводится промежуточная функция и выполняется соответствующее преобразование представления СБФ Y^* . В методе М5 также предусмотрена возможность ввода промежуточных функций с целью упрощения реализации СБФ Y^* . На третьем этапе синтеза части функций объединяются по ИЛИ с помощью PAL последнего уровня в полном соответствии с методом М3.

Метод М5 позволяет наиболее полно использовать архитектурные возможности ПЛИС:

- внутренние обратные связи ПЛИС для передачи прямых и инверсных значений реализованных функций и их частей на вход ПЛИС для использования в качестве фактор-функций;
- скрытые макроячейки ПЛИС для реализации промежуточных функций;
- макроячейки с двумя обратными связями для одновременной реализации промежуточных функций и приема значений входных переменных.

К недостаткам метода М5 относится различное время формирования значений выходных функций.

Метод М6 включает три этапа. На первом этапе определяется множество Y^* реализуемых функций в полном соответствии с одним из методов М2-М5, используемых на третьем этапе. На втором этапе реализуются конъюнкции СБФ Y^* многоуровневой схемой на ПЛИС. При этом отдельные конъюнкции или их части могут использоваться в качестве факторов для реализации других конъюнкций. На третьем этапе синтеза применяется один из методов М2-М5.

Главным положительным качеством метода М6 является отсутствие ограничений на применение.

К недостаткам метода М6 относятся:

- высокая стоимость реализации;
- низкое быстродействие синтезируемых комбинационных схем;
- различное время формирования значений выходных функций.

5. Методы синтеза конечных автоматов

Традиционный подход к синтезу конечных автоматов на ПЛИС обычно включает следующие последовательно выполняемые этапы:

- минимизацию внутренних состояний конечного автомата;
- кодирование внутренних состояний, как правило, бинарное или унарное;
- построение СБФ, соответствующей комбинационной части конечного автомата;
- минимизацию полученной СБФ;
- отображение синтезированной логической схемы на структуру выбранной ПЛИС.

Недостатки традиционного подхода заключаются в следующем:

- архитектурные особенности и параметры ПЛИС учитываются на самом последнем этапе синтеза;
- часто цели минимизации и кодирования внутренних состояний конечного автомата противоречат друг другу;
- на ранних этапах синтеза весьма слабо учитываются требования по стоимости и быстродействию синтезируемой схемы;
- совершенно не учитывается место конечного автомата в цифровой системе и др.

Пусть сложная цифровая система включает отдельные подсистемы, представляющие собой последовательностные схемы, реализуемые в виде конечных автоматов. Предлагаемый подход к синтезу последовательностных схем включает следующие действия:

- в зависимости от свойств последовательностной схемы, архитектурных особенностей ПЛИС, места подсистемы в цифровой системе, требований по стоимости и быстродействию для каждой подсистемы выбираются наиболее подходящие модели конечных автоматов;
- на основании внутренних свойств конечного автомата и выбранных моделей определяется наиболее подходящий метод синтеза;
- выполняется синтез конечных автоматов соответствующих моделей;
- строятся СБФ комбинационных частей конечных автоматов, при необходимости выполняется их минимизация;
- выполняется синтез СБФ одним из рассмотренных методов синтеза комбинационных схем на ПЛИС;
- при необходимости осуществляется отображение логической схемы в структуру целевого ПЛИС.

В данной работе рассматриваются следующие семоды синтеза конечных автоматов:

- быстрых конечных автоматов классов А (традиционный автомат типа Мили) и В (традиционный автомат типа Мура) (метод А1);
- сложных конечных автоматов классов А и В (метод А2);
- конечных автоматов Мура класса С [1] (метод А3);
- конечных автоматов Мили класса D [1] (метод А4);
- конечных автоматов Мили класса E [1] (метод А5);
- конечных автоматов Мура класса F [1] (метод А6);
- конечных автоматов Мили совмещенной модели класса ADE [1] (метод А7);
- конечных автоматов Мили совмещенной модели класса AD [1] (метод А8);
- конечных автоматов Мили совмещенной модели класса AE [1] (метод А9);
- конечных автоматов Мура совмещенной модели класса BF [1] (метод А10).

Метод А1 синтеза быстрых конечных автоматов класса А и класса В позволяет строить автоматы наивысшего быстродействия, частота переключения элементов памяти для которых равна максимальной частоте функционирования ПЛИС. Это достигается благодаря тому, что каждая функция возбуждения элементов памяти (функция переходов) реализуется на одной макроячейке ПЛИС. Для ограничения сложности функций переходов используется операция расщепления внутренних состояний таким образом, чтобы число переходов к каждому состоянию не превышало число промежуточных шин, подсоединяемых к одной макроячейке ПЛИС. В процессе кодирования внутренних состояний контролируется сложность функций переходов и в случае невозможности реализации некоторой функции переходов на одной макроячейке ПЛИС увеличивается число R разрядов кода внутренних состояний.

Метод А1 также позволяет эффективно использовать макроячейки ПЛИС с двумя обратными связями: одновременно для реализации функций переходов и приема значений входных переменных.

Главным недостатком метода А1 является узкая область использования из-за расхождения (выполнения до бесконечности) алгоритма расщепления внутренних состояний.

Метод А2 синтеза сложных конечных автоматов классов А и В позволяет строить на ПЛИС конечные автоматы практически неограниченной сложности, однако не гарантирует наивысшее быстродействие конечного автомата. Для повышения быстродействия конечного автомата в методе А2 расщепление состояний используется для ограничения числа переходов в только в отдельные “плохие” состояния, при этом алгоритм расщепления никогда не расходится.

В обоих методах А1 и А2 синтеза конечных автоматов классов А и В предусматривается программирование логического уровня выходных сигналов ПЛИС для снижения стоимости реализации.

Главной особенностью методов А3 и А4 синтеза конечных автоматов классов С и D является использование триггеров выходных макроячеек ПЛИС в качестве элементов памяти конечного автомата, когда выходные наборы совпадают с частью кода внутренних состояний. Это позволяет значительно снизить стоимость реализации и одновременно повысить быстродействие конечных автоматов, по сравнению с автоматами классов А и В.

Снижение стоимости реализации обеспечивается уменьшением числа используемых выходных ПЛИС, так как выходные функции и часть элементов памяти автомата реализуются на одних и тех же макроячейках ПЛИС. Кроме того, упрощается комбинационная часть конечного автомата, поскольку отпадает необходимость в реализации части функций возбуждения элементов памяти, которые совпадают с выходными функциями.

Увеличение быстродействия конечных автоматов классов С и D, по сравнению с традиционным подходом, объясняется тем, что в автоматах классов С и D реализуются преимущественно выходные функции, которые, как правило, проще функций переходов, что в итоге приводит к уменьшению числа логических уровней при синтезе комбинационной части конечного автомата.

Недостатком метода является необходимость расщепления внутренних состояний при переходе от автомата типа Мили к автомату типа Мура, а метода А4 - необходимость расщепления внутренних состояний при переходе от автомата класса А к автомату класса D. Кроме того, для построения автоматов класса D ПЛИС должны допускать конфигурацию выходных макроячеек с триггерами в цепях обратных связей.

Главной особенностью методов А5 и А6 синтеза конечных автоматов классов Е и F является использование триггеров входных буферов ПЛИС в качестве элементов памяти конечного автомата, когда входные наборы совпадают с частью кода внутренних состояний. Это позволяет значительно снизить стоимость реализации и одновременно повысить быстродействие конечных автоматов, по сравнению с автоматами классов А и В. В общем случае для возможности построения конечных автоматов классов Е и F каждый входной буфер ПЛИС должен иметь две связи с внутренней логикой ПЛИС: регистровую и комбинационную.

К недостаткам методов А5 и А6 следует отнести необходимость расщепления внутренних состояний для приведения исходных конечных автоматов классов А и В к автоматам классов Е и F. Кроме того, при синтезе автомата класса F также используется расщепление внутренних состояний для приведения автомата типа Мили к автомату типа Мура.

Метод А7 синтеза совмещенной модели конечных автоматов класса ADE позволяет наиболее эффективно использовать архитектурные возможности ПЛИС: триггеры входных буферов и выходных макроячеек в качестве элементов памяти конечного автомата. Особенностью метода А7 является максимальное использование моделей автоматов классов D и E, что позволяет минимизировать число R отдельных элементов памяти. В методе А7 используется расщепление внутренних состояний по входным и выходным наборам.

Недостатком метода А7 является повышенное требование к архитектурным возможностям ПЛИС: наличие входных буферов с двумя видами связей и возможность конфигурации выходных макроячеек с триггерами в цепях обратных связей.

Методы А8 и А9 синтеза совмещенных моделей автоматов классов AD и AE соответственно совпадают с методом ADE, когда нельзя использовать входные буферы ПЛИС (метод А8 синтеза автоматов класса AD) или выходные макроячейки (метод А9 синтеза автоматов класса AE) в качестве элементов памяти.

Метод А10 синтеза совмещенной модели автоматов класса BF совпадает с методом А9 для автоматов класса AE в случае его применения к автоматам типа Мура. Для приведения исходного конечного автомата к автомату типа Мура в методе А10 используется расщепление внутренних состояний.

6. Результаты экспериментальных исследований

В качестве исходных данных использовались эталонные примеры комбинационных схем и конечных автоматов, разработанные в центре MCNC (Microelectronics Center of North Carolina) [2]. Данные примеры представлены в виде текстовых файлов на входном языке системы SIS [2]. Методы пакета ZUBR сравнивались с методами, реализованными в следующих промышленных пакетах:

- MAX+PLUSII 10.1 фирмы Altera;
- WebPack 5.2 фирмы Xilinx;
- FPGA Advantage 5.2 фирмы Mentor Graphics.

При синтезе каждого эталонного примера с помощью промышленных пакетов осуществлялась следующая последовательность действий:

- исходное описание эталонного примера со входного языка системы SIS с помощью конвертера пакета ZUBR представлялось на языке VHDL (для пакета MAX+PLUSII - на языке AHDL);
- выполнялся синтез комбинационной схемы или конечного автомата с помощью промышленного пакета на ПЛИС соответствующего семейства, при этом параметры синтеза устанавливались на максимальную минимизацию стоимости (площади);
- результаты синтеза оценивались по стоимости (числу используемых макроячеек/логических элементов ПЛИС) и быстродействию (максимальной задержке в наносекундах прохождения сигналов со входов на выходы синтезированной схемы) средствами промышленного пакета.

При синтезе каждого эталонного примера с использованием пакета ZUBR осуществлялась следующая последовательность действий:

- выполнялся синтез комбинационной схемы или конечного автомата с помощью соответствующего метода синтеза пакета ZUBR на ПЛИС соответствующего семейства;
- формировалась система логических уравнений, соответствующая синтезированной комбинационной схеме или комбинационной части конечного автомата;
- система логических уравнений с помощью конвертера пакета ZUBR представлялась на языке VHDL (для пакета MAX+PLUSII - на языке AHDL);
- выполнялся синтез комбинационной схемы, представленной системой логических уравнений, с помощью промышленного пакета на ПЛИС соответствующего семейства, при этом параметры синтеза устанавливались на максимальную минимизацию стоимости (площади);
- результаты синтеза оценивались по стоимости (числу используемых макроячеек/логических элементов ПЛИС) и быстродействию (максимальной задержке в наносекундах прохождения сигналов со входов на выходы комбинационной схемы) средствами промышленного пакета.

Результаты экспериментальных исследований приведены в табл.1 и табл.2, где Packet – промышленный пакет, с которым сравниваются предлагаемые методы синтеза; Method – используемый метод пакета ZUBR; PLD – семейство ПЛИС, для которого выполняется синтез; C_{mid} – среднее (число раз) снижение стоимости реализации при использовании пакета ZUBR; C_{max} – максимальное (число раз) снижение стоимости реализации при использовании пакета ZUBR; D_{mid} – среднее (число раз) повышение быстродействия при использовании пакета ZUBR; D_{max} – максимальное (число раз) повышение быстродействия при использовании пакета ZUBR.

Таблица 1. Сравнение эффективности методов синтеза комбинационных схем пакета ZUBR

Packet	Method	PLD	C_{mid}	C_{max}	D_{mid}	D_{max}
MAX+PLUSII	M2	CLASSIC	6,11	25,56	7,10	15,20
MAX+PLUSII	M2	MAX7000	1,40	1,88	5,75	13,64
MAX+PLUSII	M2	FLEX10K	1,18	3,26	1,47	1,90
MAX+PLUSII	M5	CLASSIC	6,43	20,91	1,46	4,53
MAX+PLUSII	M5	MAX7000	1,42	1,75	2,19	5,17
MAX+PLUSII	M5	FLEX10K	1,93	3,25	1,00	1,40
WebPack	M5	XC9500	1,16	1,29	0,92	1,12
WebPack	M5	VIRTEX II	1,85	2,78	1,01	1,09
FPGA Adv.	M2	MAX7000	4,59	12,7	1,23	1,53
FPGA Adv.	M2	FLEX10K	1,48	3,00	1,23	1,44
FPGA Adv.	M5	MAX7000	1,35	1,91	0,93	1,00
FPGA Adv.	M5	FLEX10K	1,40	1,84	1,00	1,40

Таблица 2. Сравнение эффективности методов синтеза конечных автоматов пакета ZUBR

Packet	Method	PLD	C_{mid}	C_{max}	D_{mid}	D_{max}
MAX+PLUSII	A2	MAX7000	1,78	2,75	1,27	1,98
MAX+PLUSII	A3	MAX5000	3,37	17,00	-	-
MAX+PLUSII	A4	FLEX10K	7,50	23,00	-	-
MAX+PLUSII	A5	FLEX10K	3,63	8,00	1,43	1,74
MAX+PLUSII	A7	FLEX10K	3,74	5,00	-	-
MAX+PLUSII	A8	FLEX10K	4,22	7,67	1,48	1,69

MAX+PLUSII	A9	FLEX10K	4,16	8,00	1,50	1,74
WebPack	A2	XC9500	2,04	3,00	-	-
WebPack	A5	VIRTEXII	3,46	9,00	1,29	1,42
WebPack	A7	XC9500	2,24	4,50	-	-
WebPack	A7	VIRTEXII	3,14	5,67	-	-
WebPack	A8	XC9500	2,06	3,00	2,09	5,98
WebPack	A8	VIRTEXII	3,65	8,67	1,29	1,43
WebPack	A9	XC9500	2,13	3,00	2,63	5,98
WebPack	A9	VIRTEXII	4,07	9,00	1,20	1,42
FPGA Adv.	A2	MAX7000	1,80	5,33	0,95	1,53
FPGA Adv.	A2	XC9500	1,96	3,63	-	-
FPGA Adv.	A7	MAX9000	2,29	6,75	0,94	1,00
FPGA Adv.	A7	FLEX10K	2,56	6,67	1,41	3,00
FPGA Adv.	A7	VIRTEXII	3,95	13,67	0,97	1,20

Проведенные экспериментальные исследования показали высокую эффективность реализованных в пакете ZUBR методов синтеза. Так, использование пакета ZUBR при синтезе комбинационных схем (табл.1) позволяет, в среднем, снизить стоимость реализации в от 1,16 до 6,43 раза, а для отдельных примеров – в 25,56 раза. При этом среднее значение быстродействия может увеличиваться в 7,1 раза, а для отдельных примеров – в 15,2 раза. Использование пакета ZUBR при синтезе конечных автоматов (табл.2) позволяет, в среднем, снизить стоимость реализации в от 1,78 до 7,5 раза, а для отдельных примеров – в 23 раза. При этом среднее значение быстродействия может увеличиваться в 2,63 раза, а для отдельных примеров – в 5,98 раза.

7. Заключение

Метод M1 наиболее эффективен при синтезе относительно простых комбинационных схем для любых классов ПЛИС в цифровых системах как с прямой, так и с инверсной логикой.

Метод M2 эффективен при синтезе достаточно сложных комбинационных схем в цифровых системах с инверсной логикой на ПЛИС, поддерживающих программируемые опции open-drain и pull-up.

Метод M3 предназначен для использования в ситуациях, когда метод M2 не применим, например, цифровая система функционирует в положительной логике, ПЛИС не допускает монтажного соединения выходов, усложняется этап конструкторского проектирования и др.

Метод M4 предназначен для использования в ситуациях, когда метод M3 не применим из-за различного времени формирования значений выходных сигналов.

Метод M5 может быть рекомендован для реализации сложных СБФ, когда быстродействие не критично, а следует минимизировать стоимость реализации.

Метод M6 предназначены для синтеза очень сложных СБФ, реализация которых другими методами невозможна.

Метод A1 позволяет строить на ПЛИС относительно простые конечные автоматы наивысшего быстродействия.

Метод A2 позволяет строить на ПЛИС конечные автоматы практически неограниченной сложности номинальные по стоимости реализации и быстродействию.

Методы A3, A4, A5 и A6 в отдельных случаях, зависящих от внутренних свойств конечного автомата, позволяют строить конечные автоматы очень низкой стоимости и высокого быстродействия.

Методы A7, A8, A9 и A10 синтеза совмещенных моделей конечных автоматов позволяют максимально использовать архитектурные возможности ПЛИС, что приводит к построению конечных автоматов низкой стоимости и высокого быстродействия.

Литература

1. Соловьев В.В. Проектирование цифровых систем на основе программируемых логических интегральных схем. - Москва: Горячая линия - Телеком, 2001. - 636 с.

2. Yang S. Logic synthesis and optimization benchmarks user guide. Version 3.0. - Technical Report, Microelectronics Centre of North Carolina, 1991. - 43p.

