

Решение задач трассировки межсоединений с ресинтезом для реконфигурируемых систем на кристалле

С.В. Гаврилов¹, Д.А. Железников^{1,2}, В.М. Хватов^{1,2}

¹Институт проблем проектирования в микроэлектронике Российской академии наук, г. Москва, Россия

²Национальный исследовательский университет «МИЭТ», г. Москва, Россия

Sergey_G@ippm.ru

Существующие средства автоматизации проектирования ориентированы, главным образом, на технологии западных производителей, вследствие чего возникает необходимость в адаптации имеющихся методов и средств проектирования реконфигурируемых систем на кристалле и разработке собственных специализированных средств САПР для решения актуальных задач в этой области.

Предложены методы решения задач трассировки межсоединений совместно с логическим ресинтезом, применяемые к архитектуре реконфигурируемой системы на кристалле на базе отечественных ПЛИС семейства Алмаз-14. В данном кристалле разработчиками АО «НИИМЭ» и ПАО «Микрон» заложены широкие конфигурационные решения, не имеющие зарубежных аналогов. Широкий спектр дополнительных элементов для конфигурирования, а также возможности логического ресинтеза микросхемы ПЛИС Алмаз-14 приводят к необходимости разработки новых методов трассировки межсоединений, которые позволили бы учитывать и использовать эти архитектурные особенности. Разработан эффективный алгоритм автоматической трассировки межсоединений для реконфигурируемой системы на кристалле на базе ПЛИС семейства Алмаз-14 на основе алгоритма A* – модификации классического алгоритма поиска кратчайшего пути на графе, алгоритма Дейкстры, включая модель смешанного коммутационного графа. Для описания разнообразия дополнительных коммутационных элементов разработана специальная обобщенная математическая модель, а также специальный интерфейс на командном языке Tcl, включающий в себя перечень элементов для конфигурирования, их описание и функциональное назначение.

Результат работы – повышение эффективности автоматизированного проектирования с помощью разработанных и реализованных на языке программирования C программных механизмов для оптимального использования конфигурационных и трассировочных элементов ПЛИС, а также механизмы для полной и целостной трассировки межсоединений.

Ключевые слова: трассировка межсоединений; автоматизация проектирования; программируемые логические интегральные схемы (ПЛИС); логический ресинтез.

Для цитирования: Гаврилов С.В., Железников Д.А., Хватов В.М. Решение задач трассировки межсоединений с ресинтезом для реконфигурируемых систем на кристалле // Изв. вузов. Электроника. – 2017. – Т.22. – №3. – С. 266–275.

Solution of Interconnect Trace Tasks with Resynthesis for Reconfigurable Systems-on-Chip

S.V. Gavrilov¹, D.A. Zheleznikov^{1,2}, V.M. Khvatov^{1,2}

¹*Institute for Design Problems in Microelectronics RAS, Moscow, Russia*

²*National Research University of Electronic Technology, Moscow, Russia*

Sergey_G@ippm.ru

The existing means for designing automation are oriented, mainly, at technologies of western manufactures. As a result, a need in adaptation of available methods and means of designing the reconfigurable systems on chip and development of domestic specialized CAD devices for solving the actual tasks in this field appears.

The methods for solving the interconnect routing problems combined with the logical resynthesis, considering the architecture of the reconfigurable system-on-chip (RSoC) based on FPGA of Almaz-14, have been proposed. In the chip the developers from JSC «NIIME» and PJSC «Micron» have created an extensive configuration options having no foreign analogs. The availability of a wide range of additional elements for configuring as well as the capabilities of logical resynthesis of the FPGA Almaz-14 microcircuit leads to the necessity of developing the new methods for routing the interconnects, which could take into account and use these architectural features. The efficient algorithm of automatic routing of interconnects for RSoC based on FPGA of Almaz-14 series based on the algorithm A*, of the modification of a classical algorithm of searching for the shortest path on graph, the Dijkstra's algorithm, including the model of the mixed commutation graph, has been developed. For description of the variety of additional switching elements a special generalized mathematical model as well as a special command interface in Tcl language, which includes a list of configuration elements, their description and functionality, has been developed.

The of result of the work is an improvement of the automated design efficiency using the developed and implemented in C language for optimal use of the configurations and route elements of FPGA, as well as of the mechanisms for the full and correct routing of interconnects.

Keywords: interconnect routing; computer-aided design; Field-Programmable Gate Array (FPGA); logical resynthesis.

For citation: Gavrilov S.V., Zheleznikov D.A., Khvatov V.M. Solution of Interconnect Trace Tasks with Resynthesis for Reconfigurable Systems-on-Chip // Proc. of universities. Electronics. – 2017. – Vol.22. – №3. – P. 266–275.

Введение. В индустрии реконфигурируемых систем на кристалле современные коммерческие ПЛИС занимают доминирующую позицию на рынке. Их архитектура наряду с реконфигурируемой логикой содержит блоки памяти, умножители, блоки цифровой обработки сигналов и др. Также широко известны ПЛИС, в которых реконфигурируемая часть находится рядом с несколькими процессорными ядрами на кристалле. Однокристалльное реконфигурируемое или программируемое решение допускает оперативное изменение своей внутренней аппаратной структуры и конечного

предназначения на этапах производства и проектирования, вследствие чего обеспечивается комбинация гибкости проектирования и скорости выхода конечной продукции на рынок без привлечения значительных инвестиций на начальном этапе [1,2].

Текущее состояние российской микроэлектроники специального назначения соответствует мировому уровню. Так, отечественная компания ПАО «Микрон», крупнейший российский производитель интегральных микросхем и RFID (Radio Frequency IDentification), в рамках программы импортозамещения в 2017 г. планирует начать выпуск реконфигурируемых микросхем на базе отечественных ПЛИС семейства Алмаз-14 емкостью до 250 тыс. логических вентилей со встроенными блоками фазовой автоподстройки частоты, умножителями и внешней конфигурационной памятью. Разработчики АО «НИИМЭ» и ПАО «Микрон» наряду с адаптацией архитектуры ПЛИС от зарубежных производителей заложили в данный кристалл широкие конфигурационные решения, не имеющие зарубежных аналогов [3,4].

Особенности архитектурной модели ПЛИС серии Алмаз-14. Архитектура ПЛИС серии Алмаз-14 состоит из следующих элементов: логические ячейки, логические элементы (ЛЭ), группы ЛЭ, ячейки ввода/вывода, макроблоки и коммутационные элементы. Основополагающим архитектурным компонентом является логическая ячейка (рис.1), способная реализовать либо логическую функцию от трех переменных, либо триггер и имеющая возможность инвертировать входные сигналы данных. Две логические ячейки составляют ЛЭ, из которых, в свою очередь, состоят группы ЛЭ, формирующие матрицу ПЛИС.

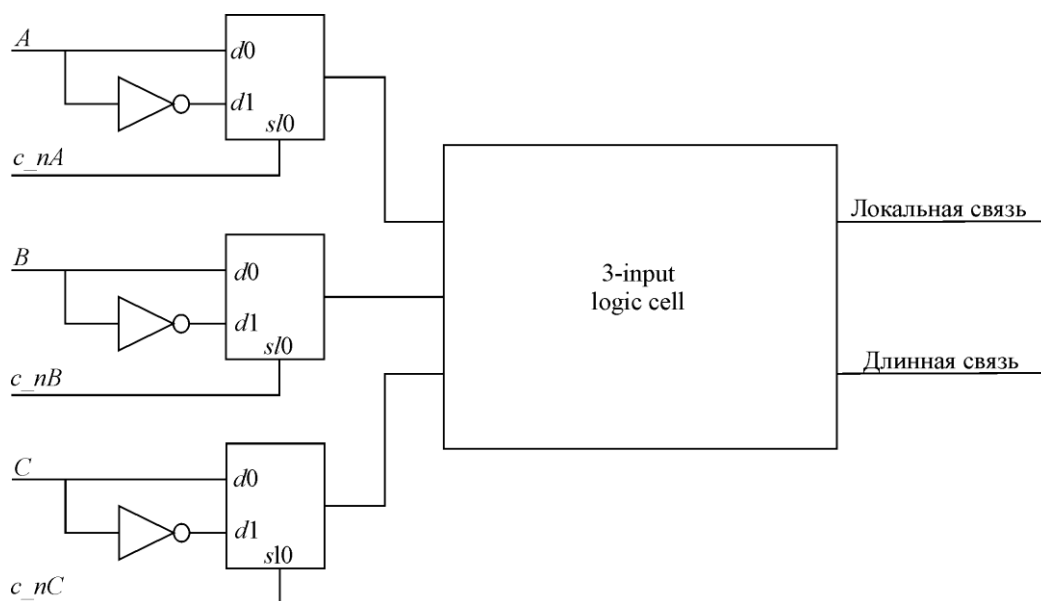


Рис.1. Структура логической ячейки ПЛИС Алмаз-14

По периметру ПЛИС расположены ячейки ввода/вывода, позволяющие внешней периферии обмениваться сигналами со схемой. Каждая из ячеек может работать как в обычном режиме (только на вход или только на выход), так и в двунаправленном. Передача сигналов между всеми компонентами происходит с помощью специализированных высокоэффективных шин, обеспечивающих целостность передаваемых данных и эффективную трассировку межсоединений.

Программирование осуществляется загрузкой пользователем необходимой информации в конфигурационную память ПЛИС (конфигурационное ОЗУ), которая управляет

ет коммутационными ресурсами ПЛИС. Коммутационные элементы распределены по всему устройству для обеспечения необходимых соединений между блоками ПЛИС и для установки режимов их функционирования.

К коммутационным ресурсам данного кристалла относятся не только стандартные ключи, но и дополнительные элементы для конфигурирования и трассировки, расположенные как снаружи, так и внутри библиотечных элементов, что отличает архитектуру ПЛИС Алмаз-14 от архитектуры классических иерархических ПЛИС. Широкий спектр дополнительных элементов конфигурирования и трассировки межсоединений кристалла ПЛИС (рис.2) является исключительной особенностью данной схемы, что обуславливает необходимость разработки новых методов трассировки межсоединений.

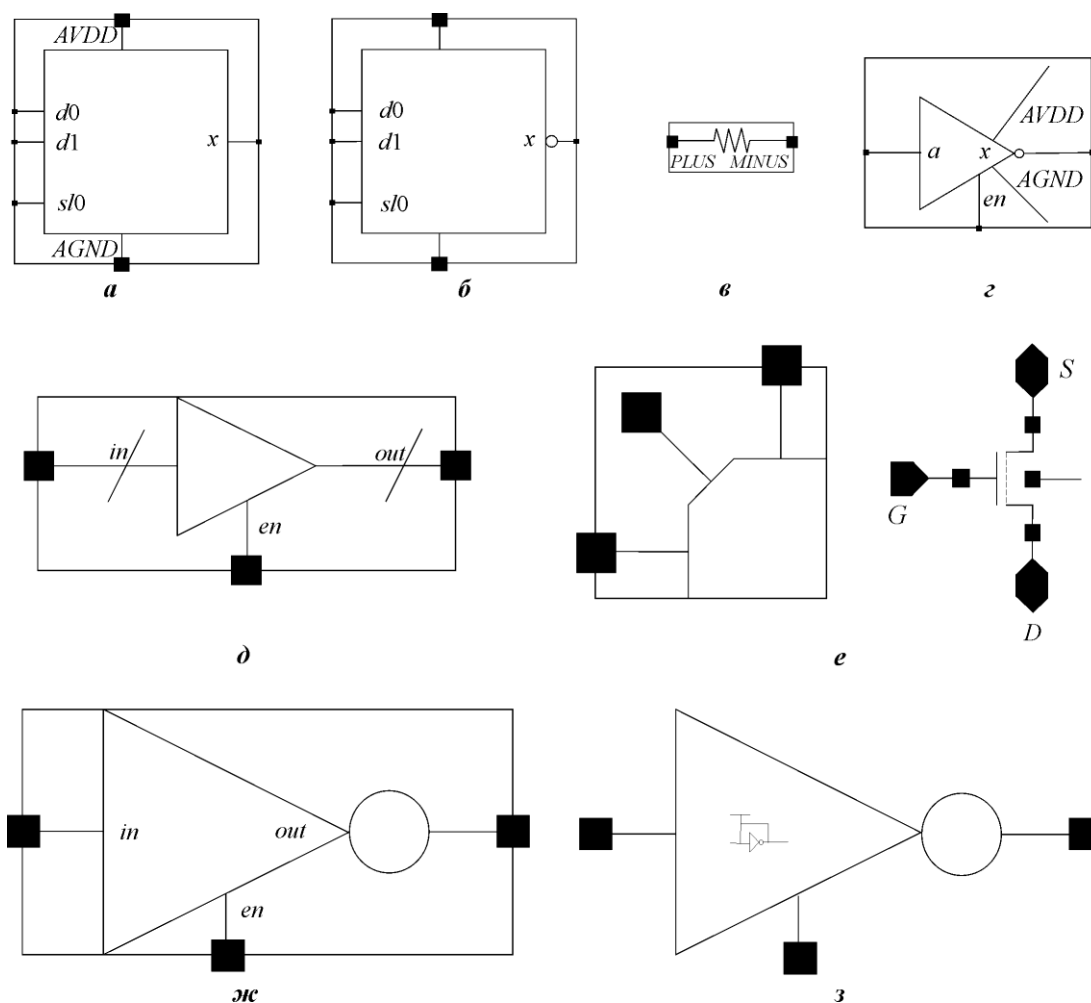


Рис.2. Элементы конфигурирования и трассировки ПЛИС Алмаз-14: а – мультиплексор без инверсии; б – мультиплексор с инверсией; в – интегральный резистор; г – усиленный инвертор; д – усиленный управляемый буфер; е – *n*-канальный МОП-транзистор; ж – инвертор с третьим состоянием; з – инвертор без управления

Математическая модель для дополнительных элементов конфигурирования и трассировки ПЛИС. Разнообразие дополнительных элементов для конфигурирования и трассировки схемы Алмаз-14 требует разработки специальной обобщенной математической модели для их описания. Среди всех дополнительных элементов для конфигурирования можно выделить различные типы коммутирования сигнала: по условию, направлению, логике и по нагрузке (усилению). В соответствии с этим предлагается

ввести следующую систему для обозначения различных типов коммутирования: $=$ – двунаправленный ключ; $<\#$ – однонаправленный ключ с инверсией; $<=$ – однонаправленный ключ без инверсии; $:\#$ – специализированный ключ с инверсией и усилением сигнала; $:=$ – специализированный ключ без инверсии и усиления сигнала.

На основании данной системы разработан специальный интерфейс на командном языке Tcl [5], позволяющий описывать взаимодействие со специальными элементами для конфигурирования и трассировки межсоединений ПЛИС, включающий в себя перечень элементов, их описание и функциональное назначение.

Разработанный интерфейс имеет следующий вид:

route_elem <имя элемента> {функциональное описание}

Функциональное описание включает в себя:

- определение условий управления элементом, который может управляться по специальному разрешающему сигналу (управляемый ключ, буфер и т.д.) или иметь безусловный тип включения в цепи (инвертор, резистор и т.д.);
- функциональную характеристику ключа, которая определяет логическую функцию работы данного элемента и может быть реализована с помощью специальных обозначений различных типов коммутирования;
- весовую характеристику ключа w , необходимую для оценки стоимости пути в алгоритме автоматической трассировки межсоединений.

Практическое использование данного интерфейса можно увидеть на примере описания некоторых дополнительных элементов для конфигурирования ПЛИС:

```
route_elem mux2_1      { !sI0 x <= d0 } { sI0 x <= d1 }
route_elem inv         { en x :# a w=1 }
route_elem buf         { en x := a }
route_elem sw          { G D == S }
route_elem res         { 1 MINUS == PLUS w=0.1 }
```

Элемент *mux2_1* – мультиплексор (рис.2,а), имеющий один управляющий вход $sI0$ и два входа данных $d0$, $d1$, выполняющий логическую функцию $x = \neg sI0 \wedge d0 \vee sI0 \wedge d1$. Элемент *inv* – усиленный инвертор (рис.2,б), управляемый сигналом en , со стандартной логической функцией $x = \neg a$ и заданным весовым коэффициентом. Элемент *buf* – буфер (рис.2,в), повторяющий и усиливающий сигнал a на выходе x , управляемый сигналом en и имеющий логическую функцию $x = en \wedge a \vee \neg en \wedge z$. Элемент *sw* – пример обычного управляемого ключа (рис.2,д) n -канального МОП-транзистора. Элемент *res* – резистор (рис.2,з), имеющий безусловное включение, передающий импульс от минуса к плюсу, с заданным весовым коэффициентом.

Кроме описанных дополнительных элементов для конфигурирования и трассировки межсоединений ПЛИС Алмаз-14 имеет еще одну отличительную особенность: коммутационные сигналы могут приходиться на информационные входы ЛЭ с инверсией. Исходя из этого, для соблюдения логики работы схемы после успешного завершения трассировки требуется логический ресинтез библиотечного элемента, входы данных которого могут быть инвертированы с помощью специальных управляющих сигналов. Для реализации возможности инвертирования входных сигналов разработан специальный интерфейс на командном языке Tcl, с помощью которого задается соответствие между сигналом, управляющим инвертированием, и инвертируемым входным сигналом данных. Синтаксис интерфейса имеет следующий вид:

set Inv(<имя библи. элемента>, <вход данных>) <управляющий сигнал>

Приведем пример из командного скрипта на языке Tcl:

```
set Inv(LE, A)      c_nA
set Inv(LE, B)      c_nB
set Inv(LE, C)      c_nC
set Inv(IO, Fast_in) "-"
```

Специальный символ "-" означает отсутствие управляющего сигнала для управления инверсией входа данных.

Схемотехнические особенности ПЛИС Алмаз-14 обусловили необходимость разработки новых алгоритмов и программных средств для автоматической трассировки.

Алгоритм трассировки межсоединений ПЛИС. Основная задача трассировки межсоединений – достижение полной трассируемости цепей с учетом минимально возможной задержки передачи сигналов, т.е. использование минимального допустимого пути. Таким образом, успешная трассировка межсоединений требует выполнения двух условий, входящих в противоречие друг с другом.

Особенности рассматриваемой ПЛИС налагают на уже имеющиеся условия трассировки дополнительные ограничения. Во-первых, ресурсы трассировки в ПЛИС дискретны и ограничены, что требует комплексного подхода с использованием как глобальной, так и детальной трассировки. Во-вторых, ПЛИС содержат в своей структуре переключатели, имеющие направление и произвольную (но фиксированную) позицию мест их соединений, что требует построения ориентированного графа. Эти особенности не позволяют напрямую применять для ПЛИС большую часть алгоритмов для решения задачи трассировки заказных ИС [6–8].

Для работы алгоритма трассировки необходим ориентированный граф, представляющий в ПЛИС ресурсы трассировки и их соединения (рис. 3), граф такого вида обозначается как $G := (V, E)$. Множество вершин V соответствует электрическим узлам или сигналам схемы, а множество ребер $E = V \times V$ – переключателям или коммутационным элементам, которые соединяют эти узлы. Из-за разнообразия типов коммутационных элементов в архитектуре ПЛИС Алмаз-14 граф трассировки становится смешанным и начинает включать в себя как ориентированные дуги для однонаправленных элементов, так и неориентированные ребра для двунаправленных.

С каждой вершиной $v_i \in V, i = \{1, 2, \dots, n\}$ в графе ассоциированы базовая стоимость w_v , которая, как правило, пропорциональна длине межсоединения; задержка D_v , которая может быть учтена в алгоритме; длина пути L_v , вычисляемая в процессе работы алгоритма. Кроме того, с каждым ребром или дугой графа $e_j \in E, j = \{1, 2, \dots, m\}$ ассоциирована весовая характеристика ключа w_e , представляющая собой относительную стоимость прохождения сигнала через данный ключ.

За основу алгоритма автоматической трассировки межсоединений для ПЛИС Алмаз-14 взят алгоритм A^* – модификация классического алгоритма поиска кратчайшего пути на графе, алгоритма Дейкстры [9–11]. Алгоритм A^* отличается от классического алгоритма Дейкстры модифицированной целевой функцией, в которой текущая стоимость пути вычисляется как сумма уже пройденного и оценки оставшегося пути.

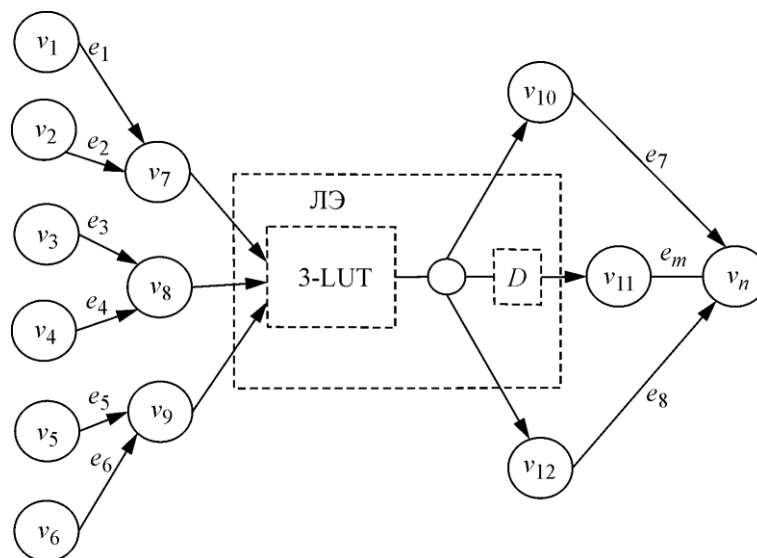


Рис.3. Пример ориентированного графа для трассировки межсоединений (3-LUT – функциональная таблица истинности)

При корректной оценке оставшегося пути такой подход позволяет целенаправленно искать путь от источника к приемнику сигнала. Для каждой рассматриваемой вершины $v_i \in V, i = \{1, 2, \dots, n\}$ алгоритм вычисляет функцию стоимости прохождения сигнала через данную вершину:

$$f(v_i) = g(v_i) + \alpha \cdot h(v_i),$$

где $g(v_i)$ – наименьший пройденный путь на момент нахождения в текущей вершине v_i ; α – весовой коэффициент, по умолчанию равный 1; $h(v_i)$ – эвристическая оценка оставшегося пути от текущей вершины v_i . Эвристическая оценка проводится с помощью предварительного запуска волнового алгоритма с расчетом минимального расстояния от всех вершин приемников сигнала $t_j \in V$ до текущей вершины v_i по следующей рекурсивной формуле:

$$h(v_i) = \min_{v_j: (v_i, v_j) \in E} (h(v_j) + 1).$$

Отметим, что эвристическая оценка оставшегося пути для всех вершин приемников $t_j \in V$ устанавливается как $h(t_j) = 0$, а в случае неориентированных ребер при $(v_i, v_j) \in E$ также верно и $(v_j, v_i) \in E$.

В данный алгоритм внесены дополнительные изменения, а именно: автоматическая генерация наряду с направленным графом графа трассировки, учет одно- и двунаправленных ключей, а также дополнительных элементов конфигурирования, описанных ранее. Помимо того, алгоритм может использовать возможность инвертирования сигнала на информационных входах элементов. Граф трассировки для работы алгоритма настраивается на основе автоматической загрузки схемотехнического описания схемы ПЛИС в формате *.cdl.

Алгоритм трассировки реализован на языке программирования C, но действует в совокупности со средой управления, разработанной на основе лингвистических средств языка Tcl, используя одно из его основных преимуществ – объединение в одной программе

управляющих скриптов на языке Tcl и программного кода на языке C. Реализованный алгоритм позволяет трассировать проектируемую схему несколькими способами:

1) полностью автоматический, приоритетный для пользователя, принимающий на вход описание схемы в виде списка цепей на языке Verilog или Tcl, схемотехническое описание схемы ПЛИС в формате *.cdl и карту размещения элементов;

2) полуавтоматический, с указанием через командный интерфейс языка Tcl особых условий для трассировки, таких как приоритет или порядок трассировки узлов, ручное назначение источника и приемника сигнала для выделенных узлов, а также использование специальных параметров самого алгоритма.

Приоритет трассировки узлов задается через командный интерфейс Tcl в следующем виде:

```
net_order high { <узел_1><узел_2> ... <узел_n> }
```

Здесь приоритет трассировки в строке убывает, т.е. чем раньше описан узел, тем раньше алгоритм его трассирует, и наоборот.

Механизм ручного назначения источника и приемника сигнала предназначен для формирования параметризованных IP-блоков в составе кристалла и выглядит следующим образом:

```
find_path <имя цепи><узел приемник><=<узел источник>
find_path <имя цепи><узел приемник><#<узел источник>
find_path <имя цепи><узел приемник>.:=<узел источник>
find_path <имя цепи><узел приемник><=0/1
find_path <имя цепи><узел приемник><#0/1
find_path <имя цепи><узел источник/приемник>==<узел приемник/источник>
```

Здесь «<=», «<#», «:=» и «==» – различные типы коммутации от узла к узлу, которые задаются по тому же принципу, что и типы прохождения сигнала через специальные элементы для конфигурирования. В случае использования данного механизма путь между узлами продумывает сам пользователь, а алгоритм не гарантирует его нахождения.

Изменение специальных параметров алгоритма трассировки также осуществляется с помощью специального командного интерфейса на языке Tcl, например:

```
set_param      maxPathL      <значение>
set_param      maxPathW      <значение>
```

Здесь *maxPathL* и *maxPathW* означают максимально возможную длину пути и максимально возможную стоимость пути соответственно.

В качестве результата работы представленных лингвистических средств проектирования на языке Tcl можно привести отрывок из разработанного описания параметризованного синхронного блока памяти:

```
proc Memory { $Irb WCLK ... WA0 ... WA7 ... WRB ... WPE DI0 ... DI8
              RCLK ... RA0 ... RA7 ... RDB ... RPE DO0 ... DO8 ... } {
  inst ramblock synchronous "wclk=$WCLK ... do8=$DO8"
  map_inst ramblock I$Irb
  find_path wclks      WCLKS := Bus_In1<10>
  ...
  find_path do8        Bus_Out2<14> <= Q2<14> }
```


Заключение. Разработана модель смешанного коммутационного графа для трассировки межсоединений, а также командный интерфейс на языке Tcl, позволяющий использовать различные элементы для конфигурирования. Предлагаемый алгоритм трассировки межсоединений ПЛИС Алмаз-14, реализованный на языке С, по сравнению с существующими алгоритмами имеет следующие отличительные особенности: возможность задания нескольких узлов-источников для одного дерева; возможность использования широкого спектра элементов для конфигурирования и трассировки межсоединений за счет разработанной модели смешанного коммутационного графа; встроенный логический ресинтез с возможностью инвертирования входного сигнала на входе логического элемента; улучшенная целевая функция для ускорения поиска пути от источника до приемника сигнала; трассировка сигналов с приоритетом, задаваемым пользователем.

Разработанный командный интерфейс и алгоритм автоматической трассировки межсоединений показали свою эффективность на схемах, проектируемых на основе семейства ПЛИС Алмаз-14 и включающих сложные конструкции, такие как параметризованные макроблоки и блоки памяти.

Литература

1. Гаврилов С.В., Гудкова О.Н., Стемпковский А.Л. Анализ быстродействия нанометровых сложнотехнологических блоков на основе интервального моделирования // Изв. вузов. Электроника. – 2012. – № 4. – С. 40–49.
2. Гаврилов С.В., Иванова Г.А. Анализ быстродействия сложных цифровых схем с учетом неопределенности технологических и схемных параметров // Вестник Рязанского государственного радиотехнического университета. – 2015. – Вып. 53. – С. 29–35.
3. Гарбулина Т.В., Лялинская О.В., Хватов В.М. Повышение эффективности проектирования интегральных схем на ПЛИС с ограниченными трассировочными ресурсами // VII Всеросс. науч.-техн. конф. «Проблемы разработки перспективных микро- и наноэлектронных систем – 2016»: сб. научн. тр. / Под общ. ред. А.Л. Стемпковского. – М.: ИППМ РАН, 2016. – Ч. I. – С. 165–171.
4. Железников Д.А., Лялинский А.А. Метод оптимизации быстродействия ПЛИС на микроархитектурном уровне с помощью механизма конвейеризации // VII Всеросс. науч.-техн. конф. «Проблемы разработки перспективных микро- и наноэлектронных систем – 2016»: сб. научн. тр. / Под общ. ред. А.Л. Стемпковского. – М.: ИППМ РАН, 2016. – Ч. III. – С. 206–211.
5. Уэлш Б., Джонс К. Практическое программирование на Tcl и Tk. – 4-е изд.: пер. с англ. – М.: Вильямс, 2004. – 1125 с.
6. Hauck S., DeHon A. Reconfigurable computing: the theory and practice of FPGA-based computation. San Francisco, CA: Morgan Kaufmann Publishers Inc., 2007. – P. 944.
7. Kuon I., Rose J. Measuring the gap between FPGAs and ASICs // IEEE Trans. on Computer-Aided Design. – 2007. – Vol. 26. – No. 2. – P. 203–215.
8. Gomez-Prado D., Ciesielski M. A tutorial on FPGA routing // Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, USA. – 2005.
9. Dijkstra E.W. A note on two problems in connexion with graphs // Numerische Mathematik. – 1959. – Vol. 2. – P. 269–271.
10. Hart P. E., Nilsson N. J., Raphael B. A. Formal basis for the heuristic determination of minimum cost paths // IEEE Trans. on Systems Science and Cybernetics SSC4. – 1968. – No. 2. – P. 100–107.
11. Sharma A., Hauck S. Accelerating FPGA routing using architecture-adaptive A* techniques // Proc. Field-Programmable Technology. – 2005. – P. 225–232.

Поступила 21.02.2017 г.; принята к публикации 21.03.2017 г.

Гаврилов Сергей Витальевич – доктор технических наук, профессор, заведующий отделом САПР Института проблем проектирования в микроэлектронике РАН (ИППМ РАН) (Россия, 124365, г. Москва, г. Зеленоград, Советская ул., д. 3), Sergey_G@ippm.ru

Железников Даниил Александрович – младший научный сотрудник ИППМ РАН (Россия, 124365, г. Москва, г. Зеленоград, Советская ул., д. 3), аспирант кафедры проектирования интегральных микросхем (ПКИМС) МИЭТ (Россия, 124498, г. Москва, г. Зеленоград, пл. Шокина, д. 1), zheleznikov_d@ippm.ru

Хватов Василий Михайлович – инженер-исследователь ИППМ РАН (Россия, 124365, г. Москва, г. Зеленоград, Советская ул., д. 3), магистр кафедры ПКИМС МИЭТ (Россия, 124498, г. Москва, г. Зеленоград, пл. Шокина, д. 1), khvatov_v@ippm.ru

References

1. Gavrilov S.V., Gudkova O.N., Stempkovskij A.L. Analiz bystrodejstviya nanometrovykh slozhnofunktsional'nykh blokov na osnove interval'nogo modelirovaniya [Nanometr IP-Blocks Performance Analysis Using Interval Simulation]. *Izvestiya vuzov. Elektronika – Proceedings of Universities. Electronic*, 2012, no.4, pp. 40–49. (In Russian).
2. Gavrilov S.V., Ivanova G.A. Analiz bystrodejstviya slozhnykh tsifrovyykh skhem s uchetom neopredelennosti tekhnologicheskikh i skhemnykh parametrov [Performance analysis of complex digital circuits with uncertainty of technological and circuit parameters]. *Vestnik Ryazanskogo gosudarstvennogo radiotekhnicheskogo universiteta – Vestnik of Ryazan State Radio Engineering University*, 2015, iss. 53, pp. 29–35. (In Russian).
3. Garbulina T.V., Lyalinskaya O.V., Khvatov V.M. Povyshenie ehffektivnosti proektirovaniya integral'nykh skhem na PLIS s ogranichennymi trassirovochnymi resursami [Improving the Efficiency of the Integrated Circuit Design on FPGA with Limited Routing Resources]. *VII Vseross. nauch.-tekhn. konf. «Problemy razrabotki perspektivnykh mikro- i nanoehlektronnykh sistem – 2016»: sb. nauchn. tr./ Pod obshh. red. A.L. Stempkovskog* [Problems of Perspective Micro- and Nanoelectronic Systems Development - 2016. Proceedings. Edited by A. Stempkovsky]. Moscow, IPPM RAN, 2016, p.I, pp. 165–171. (In Russian).
4. Zheleznikov D.A., Lyalinskij A.A. Metod optimizatsii bystrodejstviya PLIS na mikroarkhitekturnom urovne s pomoshh'yu mekhanizma konvejerizatsii [The method of timing optimization for FPGA at the microarchitecture level using the pipelining mechanism]. *VII Vseross. nauch.-tekhn. konf. «Problemy razrabotki perspektivnykh mikro- i nanoehlektronnykh sistem – 2016»: sb. nauchn. tr. Pod obshh. red. A.L. Stempkovskogo* [Problems of Perspective Micro- and Nanoelectronic Systems Development. Edited by A. Stempkovsky]. Moscow, IPPM RAN, 2016, p. III, pp. 206–211. (In Russian).
5. Welch B., Jones K. *Practical Programming in Tcl and Tk*. 4th ed. Prentice Hall PTR, 2003. 960 p. (Russ. ed.: Uehlsch B., Dzhons K. *Prakticheskoe programmirovaniye na Tcl i Tk*. 4-e izd.: per. s angl. Moscow, Vil'yams, 2004. 1125 p.).
6. Hauck S., DeHon A. *Reconfigurable computing: the theory and practice of FPGA-based computation*. San Francisco, CA: Morgan Kaufmann Publishers Inc, 2007. 944 p.
7. Kuon I., Rose J. Measuring the gap between FPGAs and ASICs. *IEEE Trans. on Computer-Aided Design*, 2007, vol. 26, no. 2, pp. 203–215.
8. Gomez-Prado D., Ciesielski M. *A Tutorial on FPGA routing*. Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, USA, 2005.
9. Dijkstra E.W. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1959, vol. 2, pp. 269–271.
10. Hart P. E., Nilsson N. J., Raphael B. A. Formal basis for the heuristic determination of minimum cost paths. *IEEE Trans. on Systems Science and Cybernetics SSC4*, 1968, no. 2, pp. 100–107.
11. Sharma A., Hauck S. Accelerating FPGA routing using architecture-adaptive A* techniques. *In Proc. Field-Programmable Technology*, 2005, pp. 225–232.

Submitted 21.02.2017; accepted 21.03.2017.