

Методы формирования и верификации библиотек стандартных элементов в составе маршрута проектирования ИС на базе ПЛИС отечественного производства

В.М. Хватов, Т.В. Гарбулина, О.В. Лялинская

Институт проблем проектирования в микроэлектронике РАН (ИППМ РАН),

khvatov_v@ippm.ru, garbulina_t@ippm.ru, olgaly@ippm.ru

Аннотация — В настоящее время появляется всё больше программируемых логических интегральных схем (ПЛИС) отечественного производства с различными типами архитектур. В этих условиях важной задачей становится обеспечение возможности проектирования интегральных схем (ИС) на данных ПЛИС и автоматизации за счёт собственных разработок в области программных продуктов и систем автоматизированного проектирования (САПР). В представленной статье содержится описание различных типов и форматов библиотек, используемых в составе разработанного в ИППМ РАН маршрута проектирования ИС на ПЛИС российского производства с уникальной архитектурой. Приведен используемый маршрут проектирования, место всех представленных форматов библиотек в этом маршруте, способы их формирования, исходя из архитектуры ПЛИС, и описание методов их верификации.

Ключевые слова — САПР (система автоматизированного проектирования), ПО (программное обеспечение), ПЛИС (программируемая логическая интегральная схема), ЛЭ (логический элемент), ЛЯ (логическая ячейка), библиотека стандартных ЛЭ, lib-файл, характеристика.

I. ВВЕДЕНИЕ

Отечественная микроэлектронная промышленность развивается в рамках импортозамещения, и в данных условиях важной стратегической задачей становится проектирование интегральных схем на ПЛИС. Это позволяет добиться высокого быстродействия, значительно сократить время проектирования, обеспечивает возможность отладки аппаратуры и внесение необходимых изменений. Процесс проектирования интегральной схемы на ПЛИС предусматривает наличие отлаженного маршрута и полноценной САПР. Традиционно для этих целей применялось программное обеспечение зарубежных производителей. Однако в настоящее время на рынке появляется всё большее число отечественных разработок с различными типами архитектур. Принимая во внимание данные факторы в ИППМ РАН был разработан маршрут, который на разных этапах может быть интегрирован с зарубежными программными продуктами (поскольку элементы зарубежных САПР продолжают

повсеместно использоваться на отечественных предприятиях) или использовать компоненты только отечественного производства, такие как XCAD или AlphaSim, разработанные также в ИППМ РАН.

Программный модуль XCAD предназначен для автоматизации этапов логического синтеза, размещения элементов и трассировки межсоединений микросхем, проектируемых на базе ПЛИС. AlphaSim предназначена для анализа электрических схем цифровых и аналого-цифровых КМОП БИС, а также для расчета параметров макромоделей (характеристики) КМОП библиотечных элементов, применяемых в программах временного и логического моделирования. Наряду с этим маршрутом также предложены маршруты проектирования на основе программных средств автоматического синтеза и интерактивного редактирования и моделирования общего назначения от компаний Synopsys, Cadence.

В ходе разработки ИС одним из ключевых моментов является использование библиотек стандартных элементов (это могут быть как логические элементы, так и ячейки ввода-вывода с различным функционалом) различных форматов в зависимости от конкретного этапа маршрута проектирования.

Методология проектирования на базе стандартных элементов, когда микросхема реализуется при помощи заранее спроектированных блоков, объединённых в библиотеку, является одной из наиболее распространённых. Это позволяет разработчикам существенно ускорить процесс проектирования и моделирования схем.

Разработка и верификация таких библиотек элементов – обширная трудоёмкая задача, и в данной статье рассматривается применительно к маршруту проектирования на ПЛИС с любой архитектурой [1].

Дальнейшее содержание работы включает следующие разделы:

- раздел II, в котором представлен общий принцип реализации маршрута проектирования интегральных схем на ПЛИС и место библиотек стандартных элементов в этом маршруте;

- раздел III, в котором рассмотрены особенности каждого из представленных форматов библиотек, методы и порядок их разработки, примеры описания логических элементов, составляющих данные библиотеки.

- раздел IV, включающий в себя описание как разработанных, так и уже имеющихся методов и способов верификации всех типов библиотек.

II. ОПИСАНИЕ ОБЩЕГО МАРШРУТА ПРОЕКТИРОВАНИЯ ИС НА ПЛИС

Маршрут проектирования ИС на базе ПЛИС включает в себя ряд обязательных этапов (рис. 1):

- 1) разработка RTL-описания;
- 2) верификация RTL-описания;
- 3) проведение логического синтеза в САПР фирмы Synopsys – Design Compiler, в САПР фирмы Cadence – RTL Compiler, Genius или в ПО с открытой лицензией Yosys;
- 4) декомпозиция;
- 5) планировка / размещение на ПЛИС;
- 6) трассировка / конфигурирование элементов маршрутизации;
- 7) выполнение функциональной и временной верификации спроектированной схемы на базе ПЛИС.

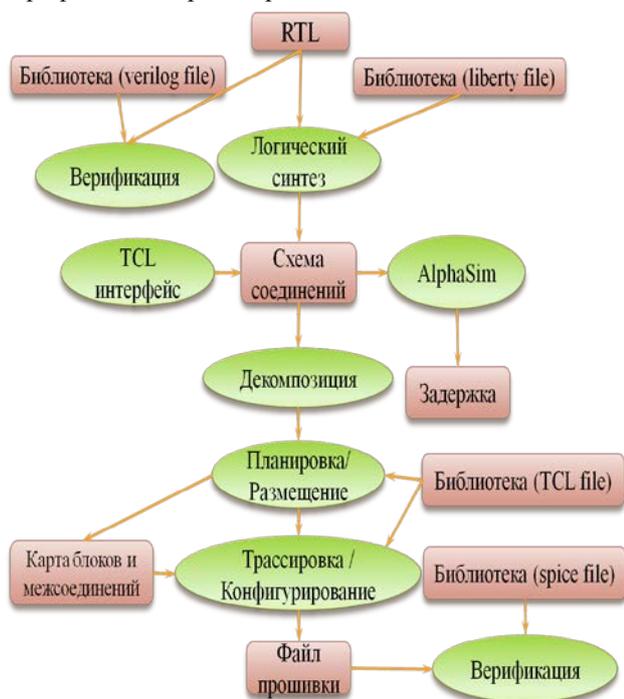


Рис. 1. Маршрут проектирования ИС

Порядок и суть каждого из этапов не зависят от используемой архитектуры ПЛИС. Но в то же время архитектура, ее особенности и ограничения влияют как на алгоритмы, применяемые на этапах

декомпозиции, размещения и трассировки, так и на структуру, содержимое и объем всех библиотек стандартных ячеек, используемых в маршруте [2-4].

Так, на разных этапах используются различные типы и форматы библиотек: библиотека с описанием ЛЭ с помощью языка Tcl, библиотека Spice-описаний и RTL-описаний логических элементов, liberty файл. Все они должны быть четко согласованы между собой и пройти этап верификации.

III. ТИПЫ БИБЛИОТЕК СТАНДАРТНЫХ ЭЛЕМЕНТОВ, ИСПОЛЬЗУЕМЫХ В МАРШРУТЕ ПРОЕКТИРОВАНИЯ ИС

A. Библиотека в формате TCL

Библиотека стандартных элементов данного типа формируется с помощью лингвистических средств, разработанных на основе языка Tcl – специальных команд, вызывающих функции из программного модуля XCAD [5-6]. Является основополагающей в маршруте проектирования устройств на основе российских ПЛИС, так как именно в ней учитывается структура одной логической ячейки ПЛИС и именно в ней происходит конфигурирование этой ячейки под конкретную логическую функцию с помощью подачи сигналов земли/питания на управляющие входы. На её основе формируется конечный список соединений схемы на языке Tcl, который далее участвует в декомпозиции, размещении и трассировке межсоединений.

Синтаксис команды `xc_lib_cell` для добавления модели библиотечного элемента выглядит следующим образом:

```
xc_lib_cell <cell_name> <subckt name>
    {<input nodes>} {<set nodes>} {<function>},
```

где:

- <cell_name> - имя библиотечной ячейки;
- <subckt name> - имя подсхемы;
- <input nodes> - входные узлы;
- <set nodes> - установить значения внутренних узлов;
- <function> - функция ячейки.

После описания в терминах данной библиотеки всех необходимых элементов и их верификации, сформированный файл подается на вход программы XCAD, которая генерирует библиотеку заданных элементов на языке схемотехнического описания Spice.

Формирование данной библиотеки не автоматизировано в связи с бесчисленным количеством возможных вариаций структуры ЛЭ ПЛИС и производится полностью разработчиком библиотек.

Применение данной команды на примере добавления в библиотеку одного ЛЭ – `al_aoi21` на языке Tcl:

```

xc_lib_cell al_aoi21 LE {
  a=A_Data
  b=B_Clk
  c=C_Reset
  y=ELLR=LR
} {
  c_A_g=0
  c_B_g=0
  c_C=0
  c_CH=1
  c_CL=0
  c_CTr=0
  c_Rst=1
  c_RstX=0
  c_Tr=0
  c_nA=0
  c_nB=0
  c_nC=0
  c_nQ=0
  c_nRst=0
} {
  +f "y = !(( a & b ) / c)"
}

```

Синтаксис команды добавления элемента в список соединений xc_inst выглядит следующим образом:

```
xc_inst <instance name> <cell_name> {<list nodes>}
```

{<inout>}, где:

- <instance name> - имя элемента;
- <cell_name> - имя библиотечной ячейки;
- <list nodes> - список узлов;
- <inout> - идентификация IO-элемента.

Пример добавления элемента в список соединений:

```
xc_inst U1 al_aoi21 {a=In1 b=In2 x=Out}
```

В. Библиотека в формате Spice

Библиотека стандартных элементов в Spice формате представляет собой совокупность подсхем (subckt), внутри каждой из которых вызывается Spice-описание логической ячейки ПЛИС из полного описания всей схемы в формате cd1. При вызове ячейки на её управляющие входы, аналогично Tc1 библиотеке, подается напряжение земли или напряжение питания, тем самым конфигурируя её под нужный ЛЭ или ячейку ввода-вывода [7].

Данная библиотека используется для дальнейшей характеристики стандартных элементов, а также для верификации результатов прошивки ПЛИС.

Пример ЛЭ из библиотеки в формате Spice:

```

.subckt xd:al_aoi21 a<A_Data> b<B_Clk>
c<C_Reset> y<ELLR> y<LR>

x_al_aoi21 a<A_Data> b<B_Clk> c<C_Reset>
+ y<ELLR> y<LR> gnd! gnd! gnd! vddh! gnd! gnd!

```

```

+ gnd! gnd! gnd! gnd! gnd! gnd! vddh! gnd! gnd!
+ vddh! vddh! vddh! vddh! LE_4p0
.ends xd:al_aoi21

```

С. Библиотека в формате LIBERTY

Для получения синтезированного описания разработанного устройства соответствующим программам на вход необходимо подать файл библиотеки в формате Liberty. Это один из наиболее распространенных форматов описания библиотеки стандартных ячеек, разработанный фирмой Synopsys. Формат является открытым, что способствует его развитию и широкому распространению. Описание библиотеки содержит информацию, необходимую для временного анализа. Использование формата Liberty позволяет точно моделировать быстродействие, мощность и помехи [8].

Формирование библиотеки данного формата осуществляется на основе уже полученной библиотеки в формате Spice и ПО AlphaSim [9-11].

На рис. 2 представлен перечень входных и выходных файлов программы AlphaSim. Для каждого логического элемента необходимо описать задание на характеристику, реализуемую логическую функцию и занимаемую площадь, указать модели транзисторов и условия характеристики (Process, Voltage, Temperature).

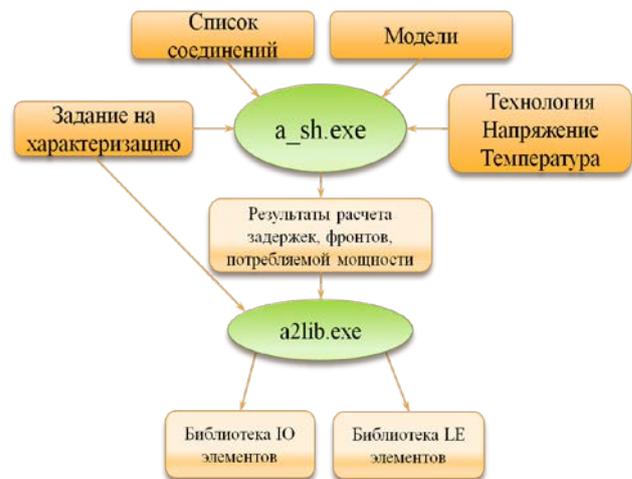


Рис. 2. Входные и выходные данные программы характеристики AlphaSim

В качестве примера представлена часть входного файла, включающего задание на характеристику элемента al_aoi21:

```

set asim_ckt_nodes(al_aoi21) { a b c y y1 }
set asim_ckt_outputs(al_aoi21) { y y1 }
set asim_ckt_inputs(al_aoi21) { a b c }
a_ckt_include al_aoi21
a_loop_testcase delay delay1
a_loop_testcase_control dynamic_power cap
a_loop_waveform a 00000rfr1111
a_loop_waveform b 00r1111f00rf

```

После получения файлов с описанием различных характеристик элементов (результатами расчёта задержек, фронтов и потребляемой мощности), программный модуль a2lib генерирует библиотеку элементов в формате Liberty. Ниже представлен фрагмент описания элемента al_aoi21 из данной библиотеки.

```
cell ( al_aoi21 ) {
  area : 2.0;
  cell_footprint : al;
  pin ( y ) {
    direction : output;
    function : "!((a&b)/c)";
    max_capacitance : 0.2;
    timing () {
      related_pin : "c";
      timing_sense : negative_unate;
      cell_rise {
        values( ... ); }
      rise_transition {
        values( ... ); }
      cell_fall {
        values( ... ); }
      fall_transition ( ) {
        values( ... ); }
    }
    timing () {
      related_pin : "a";...
```

D. Библиотека в формате Verilog

Библиотека ЛЭ в формате Verilog имеет четко определенную структуру, состоящую из следующих частей:

- Заголовок модуля;
- Описание входных узлов;
- Описание выходных узлов;
- Описание логики (logic section);
- Проверка логики с задержками (path delays).

Данный тип библиотеки используется для проверки результатов логического синтеза в целях верификации разработанного устройства. Файл библиотеки со стандартными элементами, описанными на языке Verilog, генерируется автоматически на основе liberty файла с помощью ПО, способного выполнять временной анализ схем, такого как PrimeTime фирмы Synopsys или Spectre фирмы Cadence в соединении с программным модулем a2lib, разработанным в ИППМ РАН.

Описание ЛЭ из рассмотренной библиотеки выглядит следующим образом:

```
module al_aoi21 (y, a, b, c);
  output y;
  input a, b, c;
// Function
  wire int_fwire_0, int_fwire_1;
```

```
and (int_fwire_0, a, b);
or (int_fwire_1, int_fwire_0, c);
not (y, int_fwire_1);
```

```
//Timing
specify
  (a => y) = 0.01;
  (b => y) = 0.01;
  (c => y) = 0.01;
endspecify
endmodule
```

IV. СПОСОБЫ ВЕРИФИКАЦИИ БИБЛИОТЕК РАЗЛИЧНЫХ ФОРМАТОВ

При разработке ИС должна быть гарантирована полная работоспособность всех типов библиотек, используемых на различных этапах маршрута проектирования на ПЛИС. Для этого после формирования каждого из форматов необходимо выполнить ряд действий, позволяющих произвести функциональную верификацию полного набора ЛЭ библиотеки.

Верификация выполняется как для отдельного типа библиотеки, где минимальным критерием годности является правильное выполнение каждым элементом своей логической функции, так и путем сравнения функционала двух различных типов библиотек. Критерием согласования логических элементов из различных типов библиотек является совпадение их логических функций и временных параметров.

Так, на первом этапе верифицируется библиотека стандартных ЛЭ на языке Tcl. После конфигурирования каждого ЛЭ или ячейки ввода-вывода выполняется их моделирование, что позволяет проверить работоспособность каждого библиотечного элемента.

Библиотека в формате Spice генерируется автоматически, исходя из верифицированной Tcl-библиотеки, поэтому этот тип библиотеки отдельно не проверяется, но существует возможность выполнить Spice-моделирование всех библиотечных элементов в программе AlphaSim или любом другом Spice-симуляторе. Spice-библиотека используется в дальнейшем для верификации двух других типов библиотек.

Генерация библиотеки в формате liberty выполняется на основе полученного ранее Spice-описания. Однако при формировании разработчиком liberty файла, а именно при задании входных воздействий, логических функций элементов или других параметров характеристики, могут быть допущены ошибки. Верификация данного типа библиотеки выполняется с участием библиотеки в формате Verilog, которая автоматически генерируется из liberty файла.

Первый этап верификации библиотеки в формате liberty осуществляется при помощи

программы Conformal фирмы Cadence, которой на вход подаются две библиотеки: в формате Verilog и в формате Liberty. Результатом работы программы является сообщение об эквивалентности или неэквивалентности всех представленных логических элементов. После успешного выполнения данного этапа верификации, библиотека на языке Verilog сравнивается со Spice-библиотекой с помощью специально разработанных программных средств на основе языка Tcl. На вход программе передаются две библиотеки – на языке Spice и на языке Verilog. Выполняется декомпозиция этих библиотек, описания всех стандартных элементов записываются в индивидуальные файлы с расширениями sr и v. После этого для каждого из элементов библиотеки создается файл в формате .ves с тестовыми входными воздействиями. Для комбинационной логики на входы подаются все возможные комбинации воздействий с использованием кода Грея, для последовательностной логики – комбинация, учитывающая все представленные в библиотеке типы триггеров (d-триггер, d-триггер со входом установкой, d-триггер с инверсным выходом и т.д.). Результатом моделирования Spice-описаний в программе AlphaSim и RTL-описаний в программе IcarusVerilog. являются временные диаграммы в виде .raw (AlphaSim) и .vcd (IcarusVerilog) файлов. После их обработки программа производит сравнение переключений. Для повышения достоверности результатов сравнения значения выходов триггеров предварительно устанавливаются в “0”. Две данные библиотеки считаются идентичными и верифицированными, если все ЛЭ в них работают верно, т.е. если программа по завершению работы не сообщила об ошибке переключения ни в одном из элементов этих библиотек.

Описанный набор действий обеспечивает выполнение верификации библиотеки ЛЭ на языке Tcl, библиотеки Spice-описаний, RTL-описаний ЛЭ и библиотеки в формате liberty, что позволяет использовать описанные библиотеки на разных этапах маршрута проектирования.

V. ЗАКЛЮЧЕНИЕ

Все библиотеки, использующиеся на этапах описанного маршрута проектирования ИС на ПЛИС, одинаково важны. Поэтому разработчику библиотек необходимо знать, каким образом можно сформировать каждую из них и представить заказчику полный работоспособный набор библиотек.

Для этого в данной работе были представлены методы формирования библиотек на языке Tcl, Spice, Verilog и общая структура характеристики библиотечных элементов и получения liberty файла.

Были описаны существующие способы верификации Tcl, Verilog и liberty библиотек, а также дано описание программных средств верификации библиотек формата Spice и Verilog.

Цель на будущее – максимально автоматизировать этап разработки и верификации библиотек. Прделанная работа составляет основу для дальнейших исследований в области автоматизации проектирования интегральных схем на ПЛИС отечественного производства в рамках исследований, проводимых в ИППМ РАН.

ЛИТЕРАТУРА

- [1] Угрюмов Е.П. Цифровая схемотехника 2-е изд. 2007, С.533-567.
- [2] Гарбулина Т.В., Хватов В.М. Исследование автоматического и полуавтоматического маршрутов проектирования интегральных схем на примере регулярных сложно-функциональных блоков в базисе ПЛИС 5510XC3Т // Конференция «Микроэлектроника и Информатика-2017», МИЭТ, апр. 2017г., с. 100.
- [3] Гарбулина Т.В., Железников Д.А., Липатов И.А., Тиунов И.В. Маршрут проектирования для отечественных программируемых интегральных схем специального назначения: интеграция с существующими промышленными средствами автоматизированного проектирования и решение проблем импортозамещения // Электронная техника. Серия 3. Микроэлектроника. – 2017. – № 4 (168). – С. 5-11.
- [4] Гаврилов С.В., Железников Д.А., Хватов В.М. Совместное решение задач трассировки межсоединений с ресинтезом для реконфигурируемых систем на кристалле // Известия ВУЗов. Электроника. – 2017. – Т. 22. – № 3. – С. 266-275.
- [5] Б. Уэлш, К. Джонс, Практическое программирование на Tcl и Tk [Текст] // 4-е издание.: Пер.с англ. –М.: Вильямс, 2004. С.937-985.
- [6] Дэвид М. Харрис, Сара Л. Харрис, Цифровая схемотехника и архитектуракомпьютера [Текст] // 2-е издание.: Пер.с англ. –М.: ДМК-Пресс, 2018. С.3-420.
- [7] Spectre Circuit Simulator User Guide // Cadence Design Systems, Inc. [Электронныйресурс]. Системные требования: AdobeAcrobat Reader. Режим доступа:https://www.ee.columbia.edu/~harish/uploads/2/6/9/2/26925901/spectre_reference.pdf (дата обращения: 28.11.2017).
- [8] Liberty User Guide // Synopsys, Inc. [Электронныйресурс]. Системные требования: AdobeAcrobat Reader. Режим доступа:<ftp://doc.nit.ac.ir/cee/m.gholipour/CAD%20Tools/Leonardo2003/XlibCreator/doc/liberty.pdf> (дата обращения: 01.12.2017).
- [9] AlphaSim User Guide // AlphaCHIP. [Электронныйресурс]. Системные требования: AdobeAcrobat Reader. Режим доступа:http://www.alphachip.ru/doc/AlphaSim_U_G_eng.pdf (дата обращения: 01.12.2017).
- [10] S. V. Gavrilov, O. N. Gudkova and Yu. B. Egorov. Methods of accelerated characterization of VLSI cell libraries with prescribed accuracy control // Russian Microelectronics, Vol.40, N7, 2011, P.476-482.
- [11] С.В. Гаврилов, О.Н. Гудкова, Ю.Б. Егоров “Метод аппроксимации сечений для оптимизации характеристической сетки при проектировании библиотек элементов”// IV Всероссийская научно-техническая конференция “Проблемы разработки перспективных микроэлектронных систем – 2010”: сб. научн. тр. / под общей ред. А.Л. Стемпковского. – М.: ИППМ РАН, 2010. – С.120-125.

Formation and Verification of Standard Element Libraries in the Design Flow for the Domestic FPGAs

V. M. Khvatov, T.V. Garbulina, O.V. Ljalinskaja

Institute for Design Problems in Microelectronics (IPPM RAS)

khvatov_v@ippm.ru, garbulina_t@ippm.ru, olgaly@ippm.ru

Abstract – Domestic microelectronics industry develops within the framework of import substitution. The market of programmable logic integrated circuits (FPGAs) develops and offers various types of architectures. Using FPGA allows you to achieve high performance, significantly reduce development time, provide the ability to debug equipment and make the necessary changes during the development phase. Traditionally the software of foreign manufacturers was used for that purposes. In these conditions, an important task is to provide the possibility of designing integrated circuits (ICs) on FPGAs and automating this process through own developments in the field of software products and CAD systems. IPPM RAS has developed a route that can be integrated with foreign software products at different stages or use only components of domestic production, such as XCAD or AlphaSim, developed also in IPPM RAS. The program module XCAD is designed to automate the stages of logical synthesis, placement of elements and routing of interconnections of integrated circuits designed on the basis of FPGA. AlphaSim is intended for the analysis of electrical circuits of digital and analog-digital CMOS BIS. One of the key points is the use of standard cell libraries, whose design and verification is an extensive and time-consuming task. This article describes the various types and formats of libraries used as part of the IC design flow designed for Russian FPGAs with different architectures. The design flow, all library formats and methods of their formation and verification are also presented.

Keywords – CAD (Computer-Aided Design), Software, FPGA (field-programmable gate array), LE (logical element), LC (logical cell), library of the standard LE, liberty-file, characterization.

REFERENCES

- [1] Ugrjumov E.P. Digital circuitry 2th ed. 2007 pp. 533-567.
- [2] Garbulina T.V., Hvatov V.M. Issledovanie avtomaticheskogo i poluavtomaticheskogo marshrutov proektirovaniya integral'nyh skhem na primere reguljarnyh slozhno-funkcional'nyh blokov v bazise PLIS 5510HS3T // Konferenciya «Mikroelektronika i Informatika-2017», MIET, apr. 2017, -100 p. (in Russian)
- [3] Garbulina T.V., Zheleznikov D.A., Lipatov I.A., Tiunov I.V. Marshrut proektirovaniya dlya otechestvenny`x programmiruemy`x integral`ny`x sxem special`nogo naznacheniya: integraciya s sushhestvuyushhimi promy`shlenny`mi sredstvami avtomatizirovannogo proektirovaniya i reshenie problem importozameshheniya // Elektronnyaya tekhnika. Seriya 3. Mikroelektronika. – 2017. – № 4 (168). – pp. 5-11. (in Russian)
- [4] Gavrilov S.V., Zheleznikov D.A., Hvatov V.M. Sovmestnoe reshenie zadach trassirovki mezhsoedinenij s resintezom dlya rekonfiguriruemyh sistem na kristalle // Izvestiya VUZov. Elektronika. – 2017. – T. 22. – № 3. – pp. 266-275. (in Russian)
- [5] B. Ujelsh, K. Dzhons Practical Programming in Tcl and Tk. 4th ed. –M.: Vil'jams, 2004, pp. 937-985. (in Russian).
- [6] David . M. Harris, Sarah M. Harris Digital Design and Computer Architecture. 2th ed. –N.Y.: Elsevier, 2013, pp. 3-420.
- [7] Spectre Circuit Simulator User Guide // Cadence Design Systems, Inc. requirements: AdobeAcrobat Reader. Available at:https://www.ee.columbia.edu/~harish/uploads/2/6/9/2/26925901/spectre_reference.pdf (accessed: 28.11.2017).
- [8] Liberty User Guide // Synopsys, Inc. requirements: AdobeAcrobat Reader. Available at:<ftp://doc.nit.ac.ir/cee/m.gholipour/CAD%20Tools/Leonardo2003/XlibCreator/doc/liberty.pdf> (accessed: 01.12.2017).
- [9] AlphaSim User Guide // AlphaCHIP. requirements: AdobeAcrobat Reader. Available at:http://www.alphachip.ru/doc/AlphaSim_U_G_eng.pdf (accessed: 01.12.2017).
- [10] S. V. Gavrilov, O. N. Gudkova and Yu. B. Egorov. Methods of accelerated characterization of VLSI cell libraries with prescribed accuracy control // Russian Microelectronics, Vol.40, N7, 2011, P.476-482.
- [11] S.V. Gavrilov, O.N. Gudkova, Yu.B. Egorov “Metod approksimacii sechenij dlya optimizacii karakterizacionnoj setki pri proektirovanii bibliotek e`lementov”// IV Vserossijskaya nauchno-tekhnicheskaya konferenciya “Problemy` razrabotki perspektivny`x mikroelektronny`x sistem – 2010”: sb. nauchn. tr. / pod obshej red. A.L. Stempkovskogo. – M.: IPPM RAN, 2010. – pp.120-125. (in Russian)