

## ФОРМАЛИЗАЦИЯ ПРОЦЕССА ПРОЕКТИРОВАНИЯ ПРОБЛЕМНО-ОРИЕНТИРОВАННЫХ УСТРОЙСТВ НА БАЗЕ FPGA

Владимир Опанасенко, Сергей Крывый,  
Станислав Завьялов

***Аннотация:** В работе предложены два метода к формализации процесса проектирования проблемно-ориентированных устройств: первый представлен прямонаправленным графом, второй – графом с подзадачей выбора множества оптимально реализованных функций. Разработан обобщенный алгоритм процесса проектирования на ПЛИС, который соответствует процессу формализации, позволяет выбрать одну из множества разработанных реализаций устройств в зависимости от заданных ограничений по параметрам либо в зависимости от постановки задачи оптимизации.*

***Ключевые слова:** проблемно-ориентированный процессор, синтез структуры, оптимизация, множество Парето, FPGA.*

***ITHEA Keywords:** C. Computer Systems Organization - C.1 Processor Architectures - C.1.0 - General.*

---

### **Введение**

---

Базовой компонентой современных реконфигурируемых компьютеров являются кристаллы FPGA (Field-programmable gate array). Во внутренней области кристаллов расположены: матрица конфигурируемых логических блоков – Configurable logic blocks (CLB) между которыми проходят каналы трассировок, блоки статической памяти (Block RAM), модули цифровой обработки сигналов Digital Signal Processing (DSP), а также быстродействующие приемники-передатчики Rocket IO GTP/GTX transceivers (MGT). На границах кристалла расположены блоки

ввода/вывода Input/output Blocks (IOB) и средства синхронизации Clocks & Delay-Locked Loop (DLL).

Архитектура современных кристаллов FPGA семейства Virtex 7, Spartan 7 [Oranassenko, 2016] оптимизированы для использования hard core и soft core, например, кристалл XCE7VX865T семейства Virtex 7 имеет следующие встроенные блоки: 3960 усовершенствованных модулей цифровой обработки сигналов DSP48E1, контроллер шины PCI-Express и другие.

На процесс проектирования вычислительных устройств непосредственно влияют характеристики выбранного кристалла (имеющиеся логические ресурсы, память, hard core, и т.д.) и набор инструментальных средств для проектирования, которые могут включать библиотеки готовых технических решений - soft core и другие библиотеки.

В известных методах формализованного проектирования вычислительных устройств [Капитонова, 1988] сам процесс отображается последовательностью этапов, на каждом из которых проект представлен совокупностью математических моделей, описывающих их различные части. Различают три основных вида моделей - функциональные, динамические, структурные. Функциональные модели устанавливают функции, которые выполняет проектируемая система, динамические модели устанавливают процессы функционирования системы или процессы вычислений, структурные модели отражают систему в виде композиции взаимосвязанных компонентов.

Согласно [Палагин, 2006, 2007], модель проектируемого устройства можно представить, как  $S = \langle M, A, B, P \rangle$ , где  $M$  – множество математических методов,  $A$  – множество алгоритмов реализации метода,  $B = \{b\}$  – алфавит конструктивов, из которых строится структура,  $P$  – процедура описания проекта.

Процесс проектирования состоит в решении задачи синтеза структуры на базе конструктивов  $\{b\}$  алфавита  $B$  для выполнения алгоритма  $A$  по методу  $M$ .

Результатом процедуры  $P$  является описание проекта во входном языке САПР. Заметим, что критерием эффективности метода (алгоритма) являются обобщенные характеристики разработанного устройства: аппаратные затраты, быстродействие, погрешность вычислений, сложность структуры, надежность или специальные критерии.

## 1. Постановка задачи выбора оптимальной структуры

Задача оптимального синтеза структуры сводится к задаче оптимального выбора на предварительно сформированном (и постоянно расширяемом) множестве решений [Оранасенко, 2014, 2017] для модели  $S$ . При этом необходимо учитывать наличие весьма мощного ряда кристаллов ПЛИС: каждая из имеющихся серий представлена множеством кристаллов, отличающихся быстродействием, потребляемой мощностью, логической емкостью, типом корпуса, количеством выводов и другими важными параметрами.

Целевая функция в аналитическом виде находится одним из приближенных методов, например линейной или нелинейной интерполяции или экстраполяции, по нескольким опорным точкам (структурные реализации алгоритма), которые получают путем предварительного формирования вариантов реализаций алгоритма или берут из набора готовых CORE-ядер, которые входят в состав САПР ПЛИС [Palagin, 2017 A, B]. Из множества этих точек, где каждой  $r$ -ой точке ( $r = 1 \div m$ ) соответствует реализация с параметрами  $\langle T_r, Q_r \rangle$ , формируется множество Парето на плоскости  $T - Q$  с учетом соотношений [Петросян, 2012]:

$$\begin{aligned} T_1 &\leq T_2 \leq \dots \leq T_r \leq \dots \leq T_m; \\ Q_1 &\geq Q_2 \geq \dots \geq Q_r \geq \dots \geq Q_m. \end{aligned}$$

В общем виде, задача выбора оптимального варианта реализации алгоритма сводится к минимизации функционала

$$L_r = \alpha T_r + \beta Q_r \Rightarrow \min ,$$

с учетом ограничений:

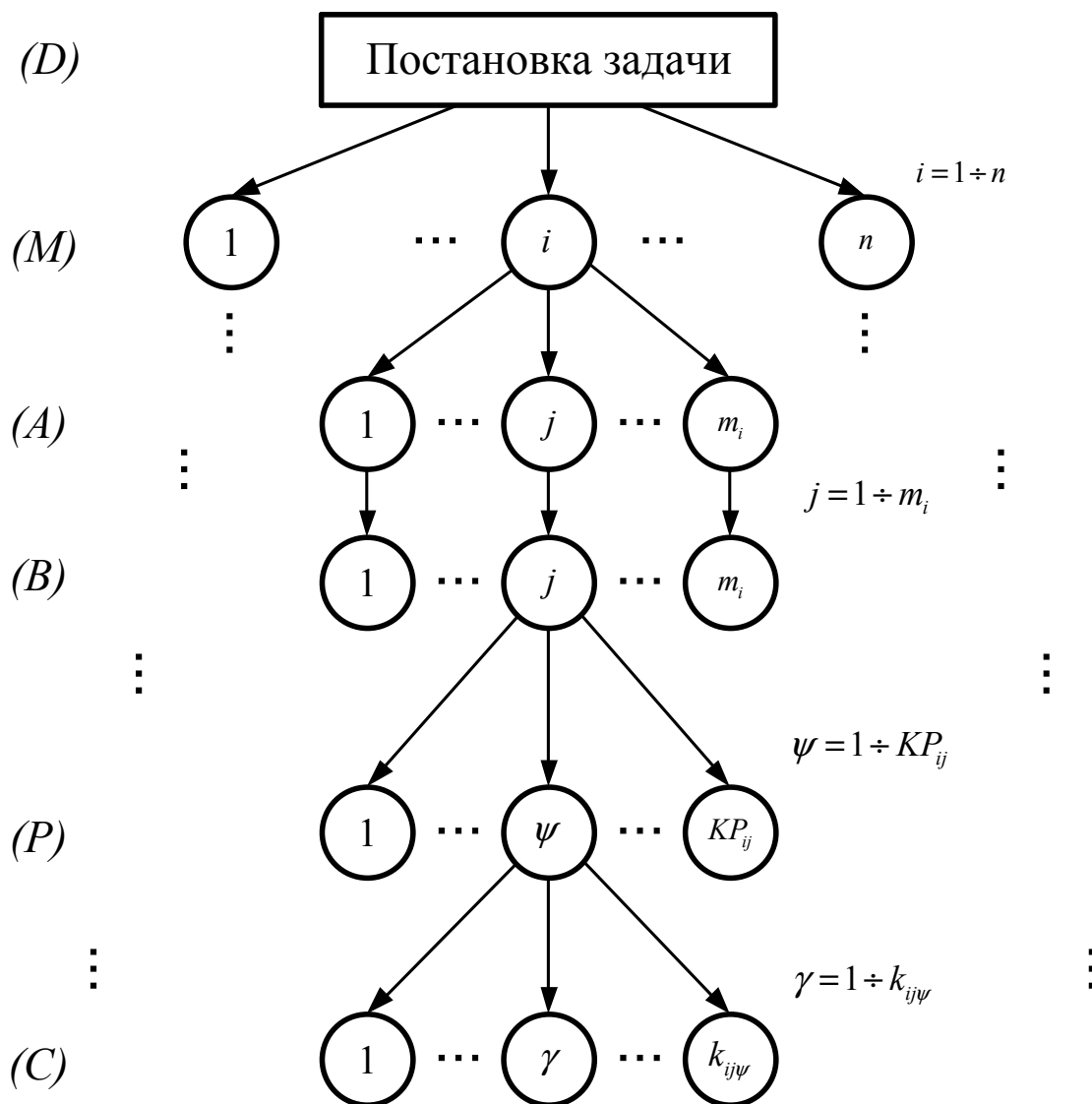
$$\begin{cases} T_r \leq T_0; \\ Q_r \leq Q_0. \end{cases}$$

где:  $\alpha, \beta$  - весовые коэффициенты, которые могут быть определены, например, методом экспертных оценок;  $T_0, Q_0$  - заданные предельные значения параметров  $T_r$  и  $Q_r$ .

Если заданным ограничениям удовлетворяет единственная точка множества Парето, то реализация, соответствующая этой точке, является оптимальной. Если ограничениям удовлетворяют несколько таких точек, то необходимо минимизировать функционал.

Рассмотрим два подхода к формализации процесса проектирования: первый представлен прямонаправленным графом, второй - графом с подзадачей выбора множества оптимально реализованных функций.

**Первый метод формализации** представлен графом (рис. 1.). Пусть задана постановка задачи  $D$  из предметной области, которая может быть решена несколькими методами  $D = \bigcup M_i, (\forall i = 1 \div n)$ . Для каждого из методов существует некоторое множество алгоритмов реализации  $A(M_i) = \bigcup A_{ij}, (\forall j = 1 \div m_i)$ . Каждый из алгоритмов реализуется на базе множества  $B = \{F_\lambda\}, (\forall \lambda = 1 \div s)$  заданных функций (например, сложение, умножение и другие). Каждая функция конструктива  $B$  может быть реализована несколькими вариантами  $F_{ij} = \bigcup C_\gamma(P_\psi(B_j)), (\forall \psi = 1 \div KP_{ij}, \forall \gamma = 1 \div k_{ij\psi})$ . Количество вариантов, которые удовлетворяют постановке задачи, будут определяться величиной  $KR_{ij} = KR'_{ij} - KR''_{ij}$ , где  $KR'_{ij}$  - общее количество возможных вариантов реализаций множеств функций,  $KR''_{ij}$  - количество вариантов реализаций множеств функций, которые не удовлетворяют постановке задачи оптимизации, и в работе не рассматриваются, поскольку для любой реализации структуры устройства характеристики устройства не будут удовлетворять постановке задачи.



**Рис. 1.** Граф первого метода формализации.

Решение задачи синтеза структуры устройства заключается в выборе из набора существующих  $\{C_\gamma\}$  ( $\forall \gamma = 1 \div k_{ij\psi}$ ) одной оптимальной структуры, удовлетворяющей постановке задачи оптимизации. Рассмотрим постановку задачи оптимизации для первого метода формализации, которая может быть сформулирована как задача минимизации целевой функции с комплексными параметрам:

$$\alpha \sum_i \sum_j \sum_\psi \sum_\gamma q_{ij\psi\gamma} x_{ij\psi\gamma} + \beta \sum_i \sum_j \sum_\psi \sum_\gamma t_{ij\psi\gamma} x_{ij\psi\gamma} \Rightarrow \min ,$$

$$(\forall i = 1 \div n); (\forall j = 1 \div m_i); (\forall \psi = 1 \div KR_{ij\psi}); (\forall \gamma = 1 \div k_{ij\psi})$$

Учитывая ограничения:

$$\sum_i \sum_j \sum_\psi \sum_\gamma q_{ij\psi\gamma} x_{ij\psi\gamma} \leq Q_0 ,$$

$$\sum_i \sum_j \sum_\psi \sum_\gamma t_{ij\psi\gamma} x_{ij\psi\gamma} \leq T_0 ,$$

$$(\alpha + \beta) = 1, \sum_\gamma x_{ij\psi\gamma} = 1 ,$$

где  $\alpha, \beta$  - весовые коэффициенты, которые могут быть определены методом экспертных оценок ( $\alpha, \beta \in [0, 1]$ );  $q_{ij\psi\gamma}, t_{ij\psi\gamma}$  - аппаратные и временные оценки (относительные)  $\gamma$ -го варианта структуры устройства построенного на функциях из  $\psi$ -го множества описаний реализаций функций для  $j$ -го алгоритма  $i$ -го метода поставленной задачи;  $x_{ij\psi\gamma}$  - булева переменная, которая определяет подходящий вариант реализации структуры устройства ( $x_{ij\psi\gamma} \in \{0, 1\}$ );  $Q_0, T_0$  - (относительные) аппаратные и временные ограничения в постановке задачи  $D$ .

Тогда решение задачи может быть получено методами целочисленного математического программирования. Методы решения таких задач достаточно хорошо разработаны и позволяют за допустимое время получать приемлемое решение [Сергиенко, 2003].

Для такой постановки задачи оптимизации  $KR''$  равно количеству тех вариантов реализаций множеств функций, которые не удовлетворяют системе ограничений:

$$\begin{cases} \sum_\lambda Q_\lambda < Q_0; \\ \max \{T_\lambda\} < T_0, \end{cases}$$

где  $\sum_{\lambda} Q_{\lambda}$  - сумма аппаратных характеристик реализаций функций  $\psi$ -го множества,  $\max\{T_{\lambda}\}$  - максимальный временной параметр реализации из всех функций  $\psi$ -го множества.

Отличие **второго метода формализации** от первого (рис. 2) состоит в том, что из множества заданных функций  $B_{ij} = \{F_{ij\lambda}\}$ , ( $\forall \lambda = 1 \div s$ ) формируется множество оптимально реализованных функций  $B'_{ij} = \{F'_{ij\lambda}\}$ , ( $\forall \lambda = 1 \div s$ ), то есть множество из перебора ( $1 \div KR_{ij}$ ), которое для конкретной реализации структуры устройства будет соответствовать постановке задачи оптимизации.

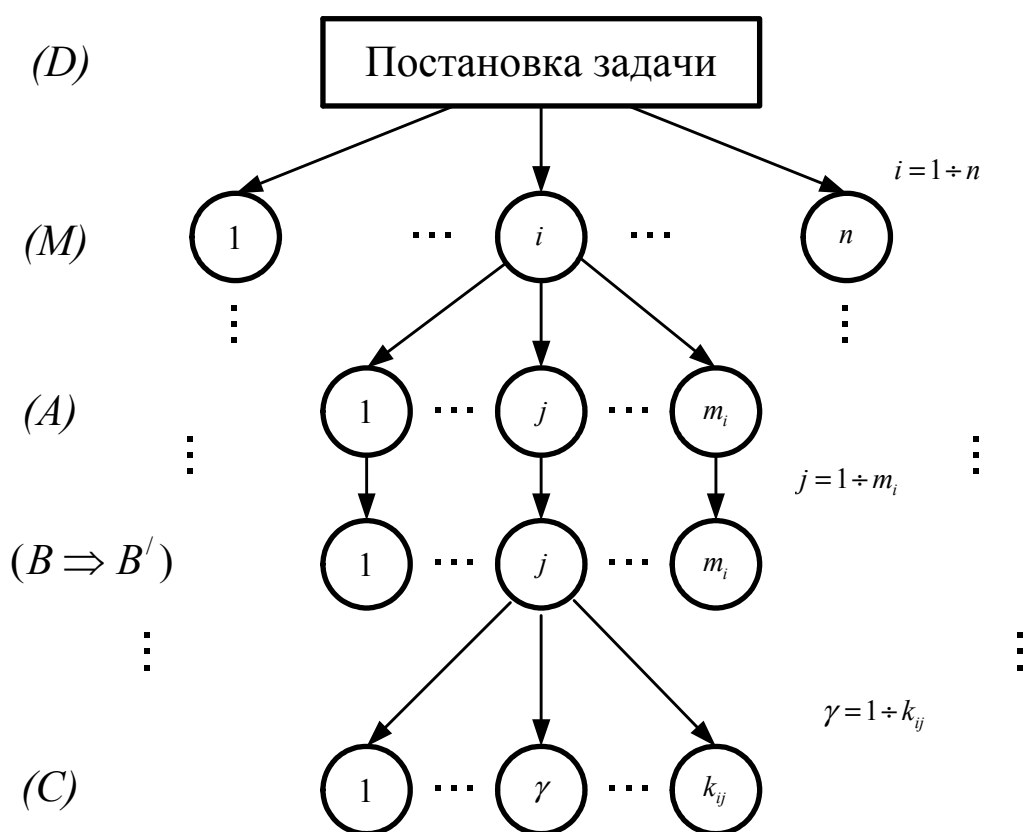
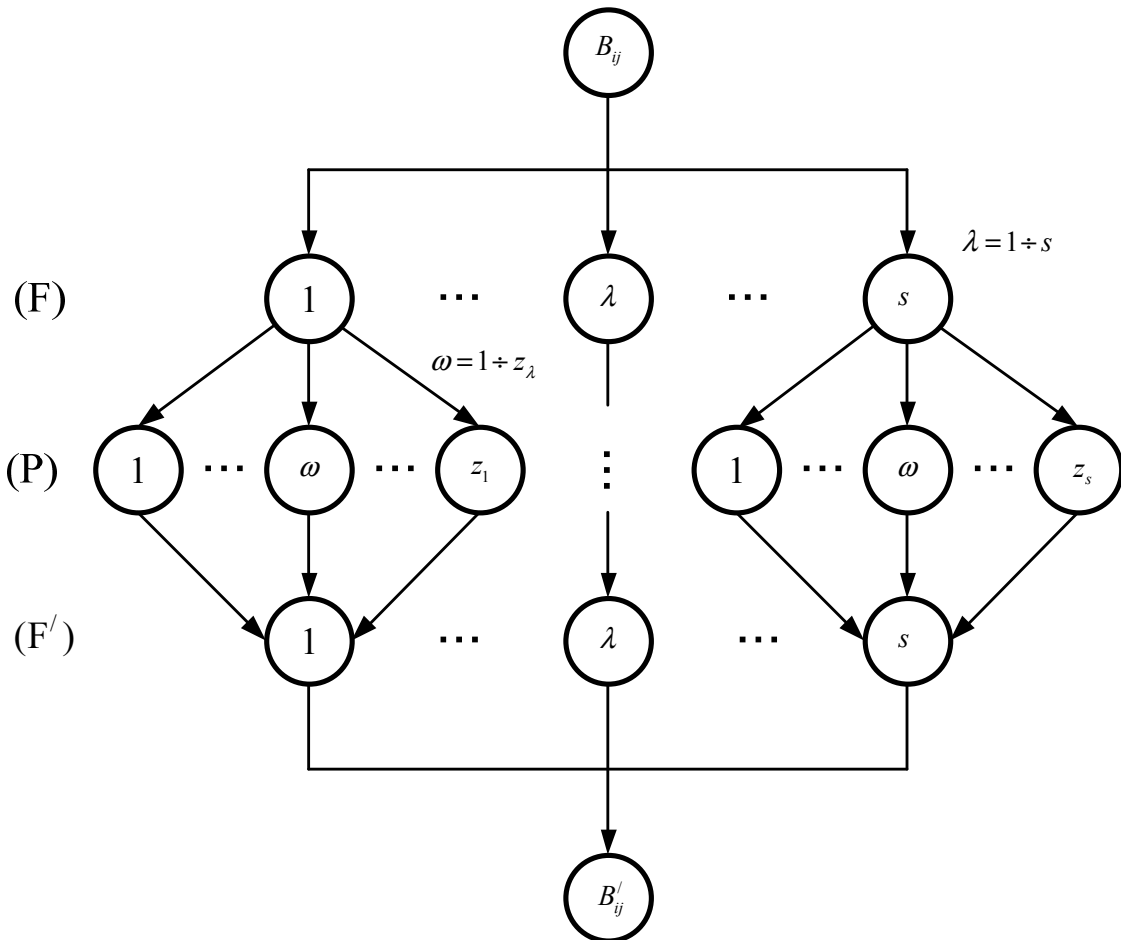


Рис. 2. Формализация процесса проектирования вторым методом.

Отображение  $B \Rightarrow B'$  - решение задачи выбора множества оптимально реализованных функций (рис. 3), которое в свою очередь зависит от постановки задачи оптимизации, алгоритма и реализации структуры устройства.



**Рис. 3.** Задача выбора множества оптимально реализованных функций

Решение задачи синтеза реализации структуры устройства будет заключаться в выборе оптимальной структуры из набора существующих  $\{C_{ij\gamma}\}$ ,  $(\forall \gamma = 1 \div k_{ij})$ , с отметкой, что структура  $C_{ij\gamma}$  будет строиться на множестве оптимально реализованных функций  $F'$ , а задача выбора множества оптимально реализованных функций рассматривается отдельно.



Рассмотрим постановку задачи оптимизации для второго типа формализации:

$$\alpha \sum_i \sum_j \sum_\gamma q_{ij\gamma} x_{ij\gamma} + \beta \sum_i \sum_j \sum_\gamma t_{ij\gamma} x_{ij\gamma} \Rightarrow \min ,$$

$$(\forall i = 1 \div n) (\forall j = 1 \div m_i) (\forall \gamma = 1 \div k_{ij})$$

учитывая ограничения:

$$\sum_i \sum_j \sum_\gamma q_{ij\gamma} x_{ij\gamma} \leq Q_0, \sum_i \sum_j \sum_\gamma t_{ij\gamma} x_{ij\gamma} \leq T_0, \sum_\gamma x_{ij\gamma} = 1,$$

$$(\alpha + \beta) = 1, \sum_\gamma x_{ij\gamma} = 1$$

Задача выбора множества оптимально реализованных функций будет заключаться в переборе из  $KR_{ij}$  комбинаций множеств функций и выборе множества оптимально реализованных функций таким образом, чтобы устройство соответствовало задаче оптимизации.

## 2. Обобщенный алгоритм процесса проектирования

На рис. 4 приведена блок-схема алгоритма процесса проектирования, где ВР - процедура верификации, условия в вершинах 3, 5 и 7 (01): успешное выполнение верификации; (00), (10) - неуспешное выполнения верификации по двум признаками верификации.

Алгоритм процесса проектирования включает следующие:

- 1: Постановка задачи включает разработку технического задания и формулировку задачи оптимизации параметров устройства.
- 2: Выполняется анализ существующих методов из множества  $M$  решения рассматриваемой задачи; в случае отсутствия таковых, разрабатываются новые методы; если имеются, то выбирается один из методов  $M_i \subset M, (\forall i = 1 \div n)$ .
- 3: Проверяется соответствие погрешностей, полученных в результате применения выбранного метода, ограничениям задачи оптимизации; используется верификация методом технического обзора документации. В

случае положительного решения верификации (10) выполняется переход к вершине 4. Если погрешности всех существующих и новых методов не удовлетворяют ограничениям задачи оптимизации (10), то выполняется переход ( $i = i + 1$ ) на вершину 2. Если проанализированы все методы  $M_i \subset M$  и новые разработанные методы, то устройство не может быть реализовано (10) и выполняется переход на вершину 15.

4: Выполняется анализ существующих алгоритмов реализации выбранного метода и разработка новых алгоритмов реализации; выбирается один из алгоритмов.

5: Проверяется соответствие погрешностей, полученных при использовании выбранного алгоритма, заданным ограничениям задачи оптимизации. В случае несоответствия ограничениям задачи оптимизации, и когда не все алгоритмы рассмотрены, то выполняется переход (10) на вершину 4 ( $j = j + 1$ ). В случае несоответствия, и когда рассмотрены все алгоритмы, то выполняется переход (00) на вершину 2 для выбора другого метода ( $i = i + 1$ ). Иначе переход (01) на вершину 6.

6: Выполняется анализ функций, используемых в выбранном алгоритме, поскольку источников получения функций может быть достаточно большое количество, то для каждой из функции будем иметь одну или более реализаций.

7: В случае отсутствия хотя бы одной функции, используемой в алгоритме, происходит переход на вершину 4 (10) для выбора другого алгоритма ( $j = j + 1$ ). Если выбранные реализации функций не могут быть использованы для данного устройства (00) вследствие различных причин, избирается другая реализация данной функции ( $\psi = \psi + 1$ ).

8: Выбирается или разрабатывается структура устройства.

9: В случае невозможности построить структуру устройства на выбранных реализациях функций, выбираются новые реализации ( $\psi = \psi + 1$ ).

10: Выполняется реализация поведенческой модели и синтез устройства средствами САПР.

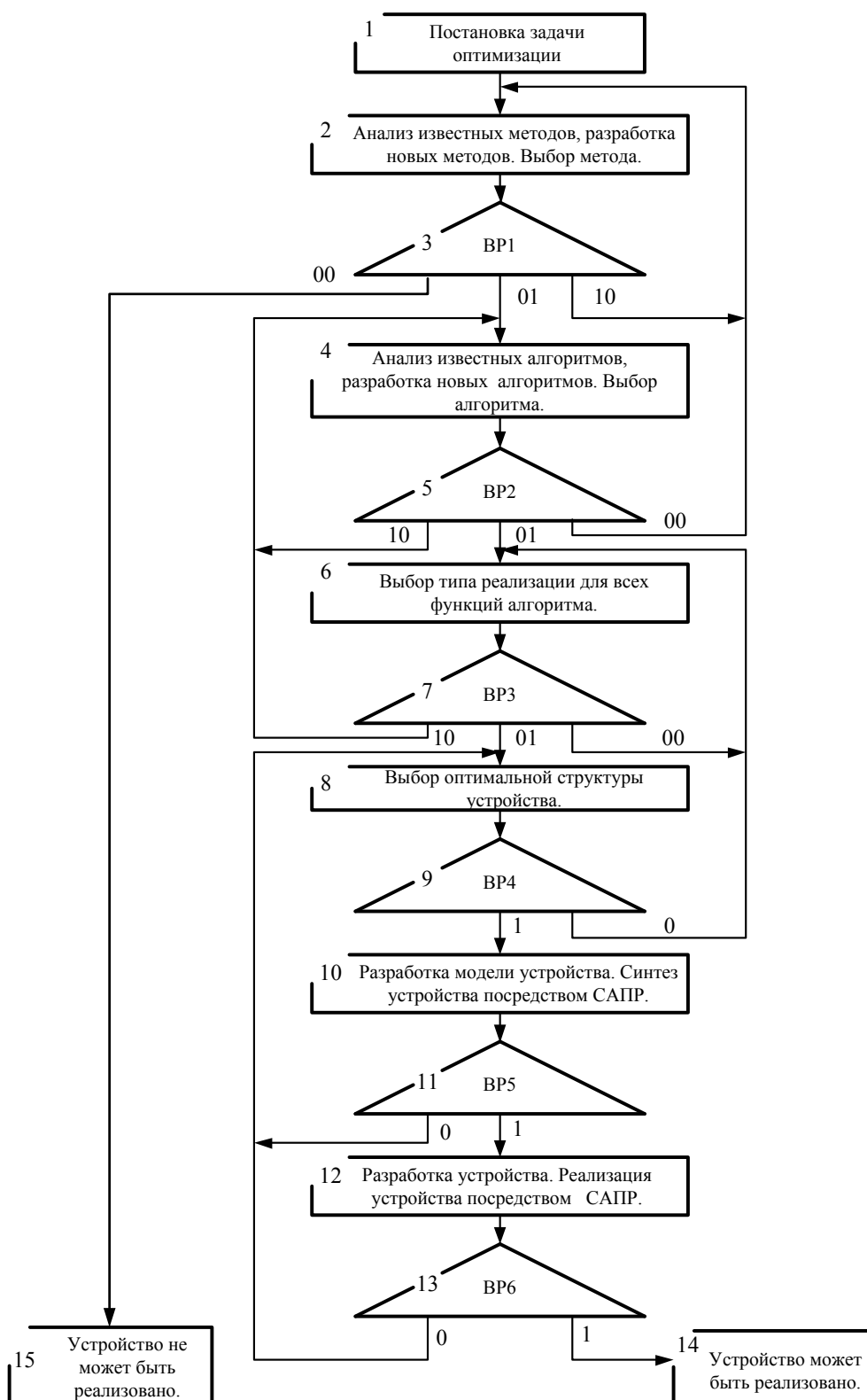


Рис. 5. Блок-схема алгоритму процесса проектирования

11: Проверяется функциональность устройства методом поведенческого моделирования.

12: Выполняются этапы: компоновка и трассировка устройства (MAP, Place and Route, Trace).

13: Проверяется функциональность устройства методом моделирования с реальными задержками. Полученные характеристики проверяются на соответствие ограничениям постановки задачи. В случае несоответствия выполняется переход к вершине  $\delta$  ( $\gamma = \gamma + 1$ ). Если же подтверждена функциональность и характеристики устройства, соответствуют постановке задачи оптимизации, то проект устройства считается реализованным.

---

### **Заключение**

---

Предложено два метода формализации процесса проектирования, которые предназначены для описания процесса разработки цифрового устройства на ПЛИС и выбора одного варианта структурной реализации путем перебора в соответствии с одной из постановок задачи оптимизации. Первый метод рассматривает все варианты структурных реализаций, то есть для каждой из  $C_{ij}$  структуры  $KR_{ij}$  вариантов.

Во втором методе уменьшается количество структурных реализаций для решения задачи синтеза структуры за счет решения задачи выбора множества оптимально реализованных функций. Разработана блок-схема алгоритма процесса проектирования на ПЛИС, которая соответствует формализации процесса проектирования, и позволяет установить реализацию устройства по заданным ограничениями параметров или постановке задачи оптимизации.

---

### **Библиография**

---

[Капитонова, 1988] Капитонова Ю.В., Летичевский А.А. Математическая теория проектирования вычислительных систем. М.: Наука, 1988. 296 с.

- 
- [Сергиенко, 2003] Сергиенко И.В., Шило В.П. Задачи дискретной оптимизации. Проблемы, методы решения, исследования. Киев: Наукова Думка, 2003. 261 с.
- [Палагин, 2006] Палагин А.В., Опанасенко В.Н. Реконфигурируемые вычислительные системы. Киев: Просвіта, 2006. 295 с.
- [Палагин, 2007] Палагин А.В., Опанасенко В.Н. Проектирование реконфигурируемых систем на ПЛИС. *Технология и конструирование в электронной аппаратуре*. 2007. №3. С. 15-19.
- [Петросян, 2012] Петросян Л.А., Зенкевич Н. А., Шевкопляс Е. В. Теория игр. СПб: БХВ-Петербург, 2012, 432 с.
- [Опанасенко, 2014] Опанасенко В.М., Крывий С.Л. Прямая задача синтеза адаптивных логических сетей. *International Journal "Information Technologies & Knowledge"*. София: ИТЕА, 2014, Vol. 8, N.1. P. 3–12.
- [Опанасенко, 2016] Опанасенко V., Kryvyi S. Synthesis of multilevel structures with multiple outputs. CEUR Workshop Proceeding of 10<sup>th</sup> International Conference of Programming, UkrPROG 2016; Kyiv. Ukraine. 2016. Vol. 1631, Code 122904. P. 32–37.
- [Опанасенко, 2017] Опанасенко V.N., Kryvyi S.L. Synthesis of neural-like networks on the basis of conversion of cyclic Hamming codes. *Cybernetics and Systems Analysis*. 2017. Vol. 53, N.4. P. 627–635. DOI: DOI 10.1007/s10559-017-9965-z.
- [Palagin, 2017A] Palagin A.V., Опанасенко V.N., and S.L. Kryvyi, Resource and Energy Optimization Oriented Development of FPGA-Based Adaptive Logical Networks for Classification Problem. [In: Green IT Engineering: Components, Networks and Systems Implementation, V. Kharchenko, Y. Kondratenko, J. Kacprzyk \(Eds.\). 2017. Vol. 105. Berlin, Heidelberg: Springer International Publishing. P. 195–218.](#) DOI: DOI 10.1007/978-3-319-55595-9\_10.
- [Palagin, 2017B] Palagin A., Опанасенко V. The implementation of extended arithmetic's on FPGA-based structures. Proceedings of the 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS'2017. (21–23

September 2017, Bucharest, Romania). 2017. Vol. 2. P. 1014–1019. DOI: DOI 10.1109/IDAACS.2017.8095239.

---

#### **Сведения об авторах**

---

**Опанасенко Владимир Николаевич** – профессор, доктор технических наук, ведущий научный сотрудник Институту кибернетики им. В.М. Глушкова НАН Украины, Украина, Киев, 03187, просп. Глушкова, 40; **e-mail:** [opanasenkovm@nas.gov.ua](mailto:opanasenkovm@nas.gov.ua)

**Кривый Сергей Лукьянович** – профессор, доктор физико-математических наук, профессор Киевского национального университета им. Тараса Шевченко, Украина, Киев, 03187, просп. Глушкова, 4д, Факультет кибернетики; **e-mail:** [krivoi@i.com.ua](mailto:krivoi@i.com.ua)

**Завьялов Станислав Борисович** – кандидат технических наук, директор ООО «Радионикс», Украина, Киев; **e-mail:** [radionix13@gmail.com](mailto:radionix13@gmail.com).

### **Formalization of the process of designing the FPGA-based problem-oriented devices**

**Volodymyr Opanasenko, Sergii Kryvyi, Stanislaw Zavyalov**

**Abstract:** *The two formalization methods of the design process of problem-oriented devices are proposed. The first one is represented by a direct-directed graph, the second one - is a graph with a subtask of selecting a set of optimally implemented functions. A general algorithm for the FPGA design process is developed. That algorithm corresponds to the formalization process, and allows you to select one of the set developed device implementations depending on the specified constraints on the parameters or on the basis of the optimization problem.*

**Keywords:** *problem-oriented processor, structure synthesis, optimization, Pareto set, FPGA.*