

Оптимизация состава библиотеки элементов для синтеза самосинхронных схем

Л.П. Плеханов

Институт проблем информатики ФИЦ ИУ РАН, г. Москва, lplekhanov@inbox.ru

Аннотация — Самосинхронные цифровые схемы имеют ряд преимуществ перед другими типами схем. Для обеспечения этих преимуществ необходимо специальное построение и устройств с памятью, и информационной части схем и индикаторной подсхемы. Одними из главных проблем синтеза самосинхронной схемы являются снижение затрат в транзисторах и улучшение быстродействия информационной части схемы и индикаторной подсхемы. Оба показателя прямо зависят от наличия в библиотеке для синтеза однокаскадных элементов с одним выходом и большим числом входов. Целью работы было найти регулярный способ определения таких элементов. Предлагается формализованный метод выбора библиотечных логических элементов для синтеза самосинхронных схем. Метод позволяет сформировать такой состав библиотеки, который будет давать оптимальные решения при синтезе самосинхронных схем с использованием критериев минимального числа транзисторов и максимального быстродействия.

Ключевые слова — самосинхронные схемы, синтез самосинхронных схем, библиотеки элементов схем.

I. ВВЕДЕНИЕ

Вопросы формирования библиотек элементов и ячеек для синтеза микросхем являются актуальными и обсуждаются с различных точек зрения на конференциях и в публикациях.

В докладе [1] анализируются ячейки из доступных библиотек и даются рекомендации для обязательного включения в состав библиотек ряда наиболее употребительных элементов.

В работе [2] приводятся методы проектирования библиотек на этапах логического синтеза и других, рассматриваются особенности формирования библиотек с учетом архитектуры проектируемых устройств.

В докладе [3] анализируется энергопотребление элементов и предлагаются методы оптимизации структуры транзисторов для библиотечных элементов.

В докладе [4] решаются вопросы проектирования библиотечных ячеек на схемотехническом уровне с ориентацией на топологическую реализацию. Даются примеры экономичных D-триггера и полусумматора.

Проектирование самосинхронных схем существенно отличается от создания схем других

видов. Основные пользовательские показатели самосинхронных схем, создаваемых на базе библиотек элементов, значительно зависят от состава этих библиотек, что делает тему статьи актуальной.

II. ОСОБЕННОСТИ СИНТЕЗА САМОСИНХРОННЫХ СХЕМ

Самосинхронные цифровые схемы имеют уникальные свойства по сравнению с синхронными и другими асинхронными схемами:

- полное и принципиальное отсутствие состязаний при любых задержках элементов схемы,

- отказобезопасность, проявляющаяся в остановке всех переключений при возникновении хотя бы одной константной неисправности типа залипания в 0 или 1 и некоторых других неисправностей.

Эти свойства обеспечивают правильность функционирования в максимально широком диапазоне внешних условий (температуры и напряжения питания), а также дают базис для создания высоконадежных (сбое- и отказоустойчивых) устройств.

Впервые возможность создания самосинхронных схем (под названием «асинхронные») было показано в классической работе Д. Маллера [5].

Практические принципы и методы создания самосинхронных схем были разработаны В.И. Варшавским с сотрудниками [6, 7].

Приведенные выше свойства были впервые подтверждены прямыми экспериментами [8, 9].

Из-за специфики самосинхронных схем их проектирование проводится с использованием специальных методов, разработанных в публикациях [5 – 7].

Особенностями самосинхронных схем, реализующими уникальные свойства, являются [6, 7]:

- специальное, как правило, парафазное со спейсером (ПФС) кодирование информации, приводящее к удвоению аппаратных затрат и информационных сигналов,

- двухфазный порядок работы – обязательное чередование рабочей фазы, во время которой самосинхронная схема вычисляет рабочие состояния

информационных выходов под управлением входных сигналов, и спейсерной, в течение которой схема готовится к переключению в следующую рабочую фазу,

- наличие индикаторной подсхемы, цель которой – сформировать сигнал окончания работы каждой текущей фазы.

Число легитимных рабочих состояний выходов самосинхронной схемы в пределе равно 2^m , где m – число информационных выходов схемы; спейсерное состояние – одно. Именно это обстоятельство позволяет различить рабочие и спейсерные состояния и с помощью индикации определить окончание переключения схемы в обе фазы.

Перечисленные особенности прямым образом влияют на затраты реализации в транзисторах и быстродействие схемы. Оба показателя зависят как от метода синтеза, так и от состава библиотеки элементов для реализации.

Известно несколько методов создания самосинхронных схем.

Теоретические методы группы В.И. Варшавского [10, 11] характеризуются крайне громоздким исходным описанием в виде диаграммы изменений или сетей Петри, что делает нереальным синтез практических схем. Кроме того, такие методы не ориентируются на какую-либо библиотеку элементов.

Существует единственная зарубежная методология построения самосинхронных схем – *Null Convention Logic (NCL)* [12]. Эта методология была создана опытным путем, по принципу «запрос-ответ». Каждый элемент базиса проектирования NCL-схем индицирует все свои входы на своем выходе. Из-за этого схемы получаются с большой избыточностью, превышающей избыточность традиционных самосинхронных схем. NCL-методология имеет существенные недостатки: небольшой набор элементов (27), единственный конвейерный способ построения схем и другие. В результате схемы, созданные по методологии NCL, имеют в разы большие затраты в транзисторах по сравнению с аналогичными схемами, созданными по другим принципам [13].

Значительно более экономные решения можно получать с использованием функционального подхода [14, 15]. Он ориентирован на исходное описание в виде логических функций (что более удобно для разработки) и не ограничивает возможности выбора элементов.

В составе самосинхронных схем можно выделить три части: информационную, запоминающую и индикаторную. Выбор библиотечных элементов для последних двух частей не составляет проблемы, так как номенклатура уже разработанных самосинхронных триггеров весьма велика, а способы построения на них устройств с памятью хорошо разработаны.

Например, специально созданная библиотека, использованная для проектирования практических вычислительных самосинхронных устройств [16], включает 75 самосинхронных триггеров и ячеек к ним для запоминающей части и 30 гистерезисных триггеров для индикаторной части.

Остается проблема выбора элементов для информационной части (инфо-части), и она состоит в следующем.

Как показано в [14], каждый исходный (не самосинхронный) сигнал в инфо-части требует в самосинхронной схеме в среднем от 10 до 16 транзисторов в схеме индикации. Меньшее число исходных сигналов приводит также и к лучшему быстродействию индикаторной схемы. Поэтому в исходном описании в инфо-части желательно иметь как можно меньше таких сигналов. С этой целью логические функции в исходном описании укрупняются, то есть путем подстановок формируются функции с большим числом аргументов до тех пор, пока в самосинхронной библиотеке находятся элементы, необходимые для реализации этих функций.

Таким образом, ограничивающим фактором в этом процессе становится наличие логических элементов в библиотеке. Например, в упомянутой выше библиотеке есть элементы с числом входов до 9 включительно.

Практическая реализация автоматизированного синтеза самосинхронных схем учитывает особенности синтеза [17].

Для этого множество исходных (не самосинхронных) функций тем или иным способом декомпозируется до функций с числом аргументов не более трех (например, применяя известную программу логического синтеза *Yosys*). Эти функции (после ПФС-преобразования) уже должны быть обеспечены элементами самосинхронной библиотеки. Далее исходные функции укрупняются, как описано выше, и подбираются элементы из библиотеки [16].

Опыт использования такого подхода к синтезу показывает, что выявляются функции с числом аргументов 5-6, для реализации которых в имеющейся библиотеке нет элементов. В этом случае функция реализуется композицией нескольких элементов, что приводит к увеличению числа самосинхронных сигналов, подлежащих индикации, и, следовательно, к увеличению и информационной, и индикаторной частей самосинхронной схемы.

В настоящее время формирование состава логических элементов для синтеза самосинхронных схем осуществляется либо на основе ранее разработанных самосинхронных устройств [16], либо интуитивно, то есть не формализовано.

Первый подход недостаточен в силу ограниченного материала для подбора. Интуитивный подход из-за большого числа возможных реализаций (как показано ниже) может привести к элементам, вообще не

востребованным для синтеза самосинхронных устройств.

По изложенным причинам вопрос уменьшения затрат и улучшения быстродействия самосинхронных схем за счет целенаправленного подбора логических элементов для самосинхронной библиотеки является актуальным.

III. ПОСТАНОВКА ЗАДАЧИ

Поскольку предела увеличения числа логических элементов в библиотеке нет, необходимо указать детерминированный способ выбора таких элементов для самосинхронного синтеза с заданным числом входов N .

Рассматривается библиотека для проектирования самосинхронных схем, изготавливаемых по технологии комплементарный металл-диэлектрик-полупроводник (КМДП). В соответствии с особенностями самосинхронных схем требования к ее элементам состоят в следующем:

- элемент должен быть *однокаскадным* с одним выходом,
- функция элемента должна быть *монотонной*,
- ограничение на число последовательно соединенных транзисторов р-типа в элементе $ИРТ = 3$, если технологические нормы библиотеки 65 нанометров и ниже; для технологий с большими нормами допускается $ИРТ = 4$,
- ограничение на число последовательно соединенных транзисторов n-типа в элементе $ИНТ = 4$.

Однокаскадным считается элемент, каждый вход которого управляет транзисторами, непосредственно переключающими ток нагрузки. Требование однокаскадности следует из теории [7] и обеспечивает отказобезопасность. Но оно не является строго ограничительным по одному выходу и не запрещает использование, например, элемента «неравнозначность» с парафазным выходом, состоящим из двух компонентов.

Принципиальная схема такого элемента на КМДП-транзисторах, показанная на рис. 1, обеспечивает существенное сокращение числа транзисторов при совместной реализации функции «неравнозначности» и ее парафазного дополнения. Рис. 2 демонстрирует принципиальную схему элемента «неравнозначность» с двумя парафазными входами и одним парафазным выходом. Число транзисторов в ней в 1,33 раза больше, чем в схеме на рис. 1.

Монотонность функции (и элемента) [7] состоит в том, что изменение любого входа в одну сторону (увеличения или уменьшения) может изменить его выход тоже только в одну сторону при любых значениях других входов.

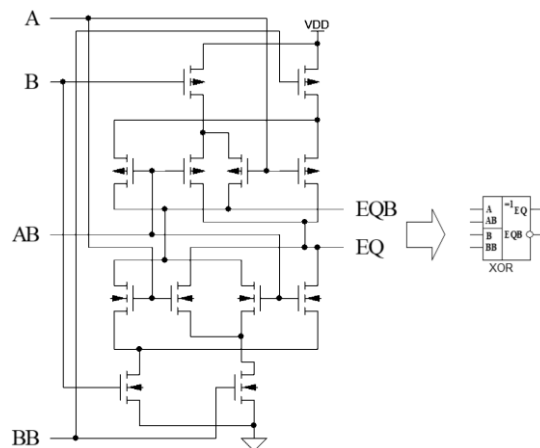


Рис. 1. Элемент «неравнозначность» с парафазным выходом на КМДП-транзисторах

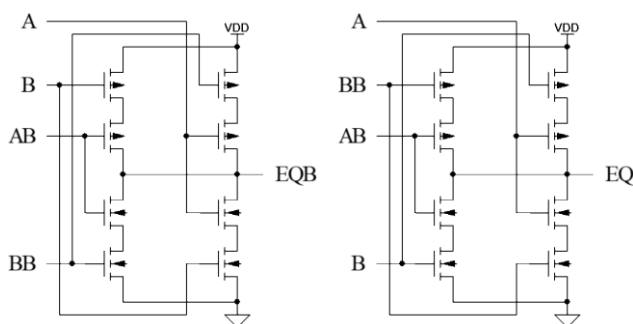


Рис. 2. Реализация функции «неравнозначности» на однокаскадных элементах

Монотонный элемент (и его функция) является *изотонным*, если увеличение входа может вызвать только увеличение выхода, и является *антитонным*, если увеличение входа может вызвать только уменьшение выхода.

Отметим, что однокаскадные элементы КМДП-библиотек являются антитонными, за исключением схем на проходных транзисторах (например, двунаправленных ключей) и элементов, на выходе которых расположен усилительный каскад в виде инвертора.

Если на выход антитонного элемента подключен инвертор, он становится изотонным. Требования самосинхронности для такого элемента не нарушаются, если выходом элемента является только выход инвертора, и его также можно включать в библиотеку.

Далее для простоты ограничимся рассмотрением только однокаскадных антитонных элементов.

IV. ВЫБОР ЛОГИЧЕСКИХ ЭЛЕМЕНТОВ ДЛЯ САМОСИНХРОННОГО СИНТЕЗА

A. Получение необходимых функций

Рассмотрим, как получаются функции (антитонные), которые необходимо реализовать в библиотечных элементах.

Обозначим антитонную функцию как *A-функцию*.

К исходной (не самосинхронной) функции добавляется ее инверсия. Обе функции приводятся к виду дизъюнктивной нормальной формы (ДНФ). В каждой ДНФ вхождение аргумента без отрицания заменяется вхождением с отрицанием, и этот аргумент становится новым аргументом. Такой процесс получения пары назовем *ПФС-преобразованием*.

Если n – число входов исходной функции, то в результате ПФС-преобразования получается пара функций с числом аргументов от n до $2n$ в зависимости от вида. Эта пара и подлежит реализации элементами библиотеки.

Как известно, общее количество разных логических функций от n аргументов равно $2^{2^n - 1}$. Возьмем для примера исходные функции с $n = 2$. Таких функций существует 8:

$$\begin{aligned}
 Y &= a b, \\
 Y &= \bar{a} b \\
 Y &= \bar{a} \bar{b}, \\
 Y &= a \vee b, \\
 Y &= \bar{a} \vee b, \\
 Y &= \bar{a} \vee \bar{b}, \\
 Y &= a \bar{b} \vee \bar{a} b, \\
 Y &= a b \vee \bar{a} \bar{b},
 \end{aligned}$$

где $\bar{}$ - символ отрицания.

После ПФС-преобразования этих функций для их реализации в самосинхронной библиотеке потребуются всего три А-функции:

$$\begin{aligned}
 Y &= \bar{c} \bar{d}, \\
 Y &= \bar{c} \vee \bar{d}, \\
 Y &= \bar{c} \bar{d} \vee \bar{e} \bar{f}.
 \end{aligned}$$

Данный пример показывает, что для реализации исходных функций с $n = 2$ нужны А-функции с тем же числом аргументов n и только одна функция с числом аргументов $2n$. Между тем, результаты расчетов, приведенные в [14], показывают, что число существующих А-функций с 4 аргументами составляет 22 (Таблица 1). То есть далеко не все возможные А-функции востребованы для синтеза.

Подобное обстоятельство справедливо и для исходных функций с $n = 3, 4$ и так далее. Поэтому можно сделать следующий вывод.

Для исходных функций с числом аргументов n требуются для реализации в самосинхронной библиотеке:

- все А-функции с n аргументами,
- малая часть А-функций (которую можно вычислить) с большим чем n числом аргументов.

Таблица 1 содержит в качестве примера числа А-функций с различными параметрами по данным из [14].

При дальнейшем увеличении числа аргументов количество существующих А-функций увеличивается до десятков и сотен тысяч. Не все возможные А-функции могут быть приняты из-за топологической реализации, и этот вопрос должен решаться в дальнейшем.

В. Алгоритм выбора логических функций

Предлагается следующий порядок.

Таблица 1

Количество А-функций в зависимости от числа аргументов n и параметра $ИПТ$

n	ИПТ		
	2	3	4
2	2	-	-
3	4	2	-
4	4	16	2
5	-	160	56
6	-	4772	14928
7	-	19936	249670

Необходимо выбрать два числа:

- M - максимальное число аргументов исходных (не самосинхронных) функций, которые должны быть полностью обеспечены элементами самосинхронной библиотеки,
- N - максимальное число входов логических элементов самосинхронной библиотеки.

Далее вычисляются все возможные исходные функции последовательно, начиная с $n = 2$ до M включительно.

Для каждого n проводится ПФС-преобразование всех исходных функций с n аргументами и получаются возможные А-функции для реализации.

Из возможных А-функций выбираются функции с числом аргументов не более N и удовлетворяющие требованиям поставленной задачи.

Дополнительным ограничивающим фактором могут быть неприемлемые паразитные емкости в узлах соединений транзисторов и этот вопрос решается на этапе проектирования топологии.

В любом случае изложенный порядок действий определяет и приоритет включения элементов в самосинхронную библиотеку.

V. ЗАКЛЮЧЕНИЕ

Для уменьшения затрат в транзисторах и улучшения быстродействия самосинхронных схем (информационных и индикаторных частей) необходимо применять в информационной части по возможности как можно более крупные (по числу входов) библиотечные логические элементы.

Вопрос выбора необходимых логических элементов для синтеза самосинхронных схем до настоящего времени решался эвристически, что не позволяло добиться более оптимальных показателей синтезируемых самосинхронных схем.

В статье предложен формализованный метод выбора логических библиотечных элементов для синтеза самосинхронных схем, который определяет приоритетность включения элементов в самосинхронную библиотеку и обеспечивает оптимальность такого выбора в реальных условиях.

ЛИТЕРАТУРА

[1] Калашников В.С., Семёнов М.Ю. Оптимизация состава библиотек стандартных ячеек // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2016. № 2. С. 217-224.

[2] Хватов В.М., Гаврилов С.В. Формирование библиотек СФ блоков в маршруте проектирования пользовательских схем на ПЛИС и РСнК // Изв. вузов. Электроника. 2021. Т. 26. № 5. С. 387-398.

[3] Ильин С.А., Коршунов А.В., Гарбулина Т.В. Сравнительный анализ энергоэффективности библиотек по технологии FinFET 7 нм // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2020. Выпуск 4. С. 169-173. doi:10.31114/2078-7707-2020-4-169-173.

[4] Баранов А.А., Сафьянников Н.М. Схемо-топологическое проектирование ячеек СБИС // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2016. № 3. С. 220-225.

[5] Muller D.E., Bartky W.C. A theory of asynchronous circuits // Proc. Int Symp. On the Theory of Switching, Part 1. Harvard University Press, - 1959. P. 204-243.

[6] Аперидические автоматы / Под ред. В.И. Варшавского. – М.: Наука, 1976. - 423 с.

[7] Автоматное управление асинхронными процессами в ЭВМ и дискретных системах / Под ред. В.И. Варшавского. - М.: Наука, 1986. - 398 с.

[8] Плеханов Л.П., Степченко Ю.А. Экспериментальная проверка некоторых свойств строго самосинхронных схем // Системы и средства информатики. - М.: Наука, 2006. – Вып. 16. - С. 476 – 485.

[9] Соколов И.А., Степченко Ю.А., Бобков С.Г. и др. Базис реализации супер-ЭВМ эксафлопного класса // Системы и средства информатики. - М.: Торус Пресс, 2014. – Т. 25. - № 1. - С. 5–34.

[10] Варшавский В.И., Кишиневский М.А., Кондратьев А.Ю., Розенблюм Л.Я., Таубин А.Р. Модели для спецификации и анализа процессов в асинхронных схемах / Техническая кибернетика. - 1988. - № 2. - С. 171–190.

[11] Kishinevsky M., Kondratyev A., Taubin A., Varshavsky V. Concurrent Hardware: The Theory and Practice of Self-Timed Design. - London: John Wiley and Sons, 1993. - 388 p.

[12] URL: <http://www.theseusresearch.com/NCLPaper01.htm>. (Дата обращения: 20.08.2022)

[13] Taubin A., Cortadella J., Lavagno L., Kondratyev A., Peeters A. Design Automation of Real-Life Asynchronous Devices and Systems // Foundations and Trends in Electronic Design Automation. - Now Publishers Inc., 2007. - V. 2. - № 1. - P. 1 – 133.

[14] Плеханов Л.П. Основы самосинхронных электронных схем. - М.: Бином. Лаборатория знаний, 2013. - 208 с.

[15] Плеханов Л.П., Синтез самосинхронных комбинационных секций функциональным методом // Системы и средства информатики. - М.: Наука, 2017. Т. 27. № 2. С. 86 – 98.

[16] Степченко Ю.А., Денисов А.Н., Дьяченко Ю.Г., Гринфельд Ф.И., Филимонов О.П., Морозов Н.В., Степченко Д.Ю., Плеханов Л.П.. Библиотека функциональных ячеек для проектирования самосинхронных полужаказных микросхем серий 5503 и 5507 / Под общ. ред. академика РАН А.Н. Саурова. М.: ТЕХНОСФЕРА, 2017. – 376 с.

[17] Плеханов Л.П., Денисов А.Н., Дьяченко Ю.Г., Мамонов Д.И., Морозов Н.В., Степченко Д.Ю. Реализация синтеза самосинхронных схем в базе БМК // Наноиндустрия, – М: Техносфера, 2022. Спецвыпуск Т.15, № S8-1 (113), 2022 – с. 274-282.

Library Composition Optimization for Self-timed Circuit Synthesis

L.P. Plekhanov

Institute of Informatics Problems of RAS, Moscow, lplekhanov@inbox.ru

Abstract — Self-timed digital circuits have a number of advantages over other circuit types. To ensure these advantages, special construction of memory units, information part of the circuits and indication subcircuit is required. One of the main problems of the self-timed circuit synthesis is to reduce costs in transistors and improve the performance of the circuit's information part and indicator subcircuit. Both factors directly depend on the availability of

single-stage cells with one output and a large number of inputs in the library for synthesis. The aim of the work was to find a regular way to define such cells. A formalized method for selecting library logic cells for the synthesis of self-timed circuits is proposed. The method makes it possible to form such a library composition that will give optimal solutions for the self-timed circuit synthesis in terms of complexity in transistors and speed.

Keywords — self-timed circuits, synthesis of self-timed circuits, cell libraries

REFERENCES

- [1] Kalashnikov V.S., Semenov M.Y. Standard cell libraries content optimization // Problems of Perspective Micro- and Nanoelectronic Systems Development - 2016. Proceedings / edited by A. Stempkovsky, Moscow, IPPM RAS, 2016. Part 2. P. 217-224.
- [2] Hvatov V.M., Gavrilov S.V. Formirovanie bibliotek SF blokov v marshrute proektirovaniq polzovatelxskih shem na PLIS i RSnK (Formation of SF block libraries in the design route of user circuits on FPGAs and SoCs) // Izv. vuzov. Elektronika. 2021. V. 26. № 5. P. 387–398.
- [3] Ilin S.A., Korshunov A.V., Garbulina T.V. Benchmarking Energy Efficiency of Libraries on FinFET 7nm // Problems of Perspective Micro- and Nanoelectronic Systems Development - 2020. Issue 4. P. 169-173. doi:10.31114/2078-7707-2020-4-169-173
- [4] Baranov A.A., Safyannikov N.M. Schematic-topological design of VLSI cells // Problems of Perspective Micro- and Nanoelectronic Systems Development - 2016. Proceedings / edited by A. Stempkovsky, Moscow, IPPM RAS, 2016. Part 3. P. 220-225.
- [5] Muller D.E., Bartky W.C. A theory of asynchronous circuits // Proc. Int Symp. On the Theory of Switching, Part 1. Harvard University Press, - 1959. P. 204-243.
- [6] Aperiodicheskie avtomaty (Aperiodic automata) / Pod red. V.I.Varshavskogo. – M.: Nauka, 1976. - 423 p.
- [7] Avtomatnoe upravlenie asinhronnymi processami v EVM i diskretnyh sistemah (Automatic control of asynchronous processes in computers and discrete systems) / Pod red. V.I.Varshavskogo. – M.: Nauka, 1986. - 398 p.
- [8] Plekhanov L.P., Stepchenkov Yu.A. Eksperimentalnaq proverka nekotoryh svojstv strogo samosinhronnyh shem (Experimental verification of some properties of strictly self-timed circuits) // Sistemy i sredstva informatiki. - M.: Nauka, 2006. – Vyp. 16. - P. 476 – 485.
- [9] Sokolov I.A., Stepchenkov Yu.A., Bobkov S.G. and oth. Baza realizacii super-EVM eksaflopnoq klassa (The basis for the implementation of an exaflop class supercomputers) // Sistemy i sredstva informatiki. - M.: Torus Press, 2014. – V. 25. - № 1. - P. 5–34.
- [10] Varshavsky V.I., Kishinevsky M.A., Kondratyev A.Yu., Rozenblum L.Ya., Taubin A.R. Modeli dlya specifikacii i analiza processov v asinhronnyh shemah (Models for the specification and analysis of processes in asynchronous circuits) / Tehnicheskaya kibernetika. - 1988. - № 2. - P. 171–190.
- [11] Kishinevsky M., Kondratyev A., Taubin A., Varshavsky V. Concurrent Hardware: The Theory and Practice of Self-Timed Design. - London: John Wiley and Sons, 1993. - 388 p.
- [12] URL: <http://www.theseusresearch.com/NCLPaper01.htm>. (access date: 20.08.2022)
- [13] Taubin A., Cortadella J., Lavagno L., Kondratyev A., Peeters A. Design Automation of Real-Life Asynchronous Devices and Systems // Foundations and Trends in Electronic Design Automation. - Now Publishers Inc., 2007. - V. 2. - № 1. - P. 1 – 133.
- [14] Plekhanov L.P. Osnovy samosinhronnyh elektronnyh shem (Fundamentals of self-timed electronic circuits). - M.: Binom. Laboratoriya znaniy, 2013. - 208 p.
- [15] Plekhanov L.P., Sintez samosinhronnyh kombinacionnyh sekcij funkcionalnym metodom (Synthesis of self-timed combination sections by the functional method) // Sistemy i sredstva informatiki. - M.: Nauka, 2017. V. 27. № 2. P. 86 – 98.
- [16] Stepchenkov Yu.A., Denisov A.N., Diachenko Yu.G., Grinfeld F.I., Filimonenko O.P., Morozov N.V., Stepchenkov D.Yu., Plekhanov L.P.. Biblioteka funkcionalnyh yacheek dlya proektirovaniya samosinhronnyh poluzakaznyh mikroshem seriy 5503 i 5507 (Library of functional cells for the design of self-timed semi-custom chips of the 5503 and 5507 series) / Pod obsh. red. akademika RAN A.N. Saurova. M.: TEHNOSFERA, 2017. – 376 p.
- [17] Plekhanov L.P., Denisov A.N., Diachenko Yu.G., Mamonov D.I., Morozov N.V., Stepchenkov D.Yu. Realizaciq sinteza samosinhronnyh shem v bazise BMK (Implementation of the synthesis of self-timed circuits in the semi-custom basis) // Nanoindustriq, – M: Tehnosfera, 2022. Specvypusk V.15, № S8-1 (113), 2022 – p. 274-282.