

## Электронные системы

УДК 621.38

П. Арянпур, А.А. Приходько

### Использование программ семейства MicroCap для изучения и проектирования цифровых компонентов и схем

Рассмотрены функциональные возможности программ семейства MicroCap предназначенных для изучения и проектирования цифровых компонентов, схем и систем в курсах по цифровой схемотехнике. Изложены общие принципы проектирования для любых цифровых объектов и особенности моделирования цифровых объектов комбинационного типа.

Functional possibilities of the programs of family MicroCap are considered for a study and design of digital components, circuits and systems in courses for digital to circuit technique. General principles design is expounded for all digital objects and features of modeling digital objects of combinative type.

**Ключевые слова:** электронные цифровые компоненты, цифровые интегральные схемы, электронные цифровые схемы и системы жёсткой логики, булева алгебра, системы автоматизированного проектирования, схемотехническое моделирование, логическое моделирование.

#### Введение

Изучение цифровой схемотехники в технических университетах и институтах во всём её многообразии и овладение современными методологиями проектирования цифровых электронных схем и систем имеет большое практическое значение как для студентов специальности «Электронные системы» так и для студентов смежных специальностей. Однако эти процессы на данный момент не имеют общей методологии и «оторваны» друг от друга. Проблеме формирования единой методологии изучения и схемотехнического проектирования цифровых электронных схем и систем с использованием современного программного обеспечения посвящена данная работа. В основу такой методологии положим идеологию построения систем автоматизированного проектирования (САПР). Предметом исследования являются объекты и процессы изучения и проектирования в области цифровой электроники.

Объекты изучения – это логические элементы и схемы, цифровые компоненты, схемы и системы, а объекты процесса проектирования – цифровые и цифроаналоговые компоненты (интегральные схемы (ИС), микросхемы). Принято эти ИС разделять на следующие группы:

- ИС малой и средней степени интеграции жёсткой логики;
- БИС/СБИС микропроцессоров и микроконтроллеров;
- БИС/СБИС элементов памяти;
- программируемые логические ИС (ПЛИС);
- аналого-цифровые и цифро-аналоговые ИС.

Проектируемые цифровые электронные схемы и системы также являются объектами проектирования.

Отметим, что принцип непрерывности образовательного процесса обуславливает единую методологию, как изучения, так и проектирования цифровых объектов. Кроме того, в учебном процессе должны использоваться современные аппаратные и программные средства. Такой подход совпадает с тенденциями развития высшего образования, как на западе, так и в России, и идёт в русле болонского процесса [1].

С точки зрения требований, предъявляемых как к изучению, так и к проектированию, цифровые схемы и системы разделим на следующие классы:

- на компонентах жёсткой логики (классическая цифровая схемотехника) [2–7];
- на базе микропроцессоров (вычислительные цифровые системы) [8];
- на базе микроконтроллеров (цифровые управляющие системы) [9];
- ПЛИС и цифровые схемы с их использованием [10].

Цифровые и цифроаналоговые компоненты и схемы жёсткой логики изучаются студентами специальности «Электронные системы» в нескольких курсах по традиционной инженерной методике: лекции, задачи стандартной методики, лабораторный практикум на стендах и курсовые работы и (или) проекты. При этом программы логического моделирования либо со-

всем не используются, либо используются ограниченно.

На базе знаний первого класса цифровой схемотехники построено изучение курсов по микропроцессорной технике и микроконтроллерам в системах управления. Эти курсы поддерживаются специальным программным обеспечением, что очень важно для подготовки специалистов современного уровня. Однако материал этих курсов сложный и поэтому трудно воспринимается студентами.

Цифровые системы с ПЛИС не изучаются студентами специальности «Электронные системы», в то время как большая часть цифровых систем в электронике реализуются на них. В современном самолёте АН-148 конструкторы использовали около 30 таких систем. При этом важно подчеркнуть, что проектирование проводится внутри кристалла микросхемы ПЛИС, и оно существенно отличается от проектирования цифровых схем на дискретных компонентах.

Проектирование любых технических объектов выполняют по одной из двух методологий: неавтоматизированного (часто используют термин «инженерного») и автоматизированного проектирования. Инженерное проектирование выполняется по упрощённым методикам для большого количества несложных технических объектов без использования современных программных и аппаратных средств и преимущественно применяется при выполнении курсовых работ и проектов в схемотехнических курсах. Автоматизированное проектирование проводится согласно методологии САПР.

Выбираем следующую стратегию решения поставленной проблемы: на первом этапе ограничимся рассмотрением класса схем жёсткой логики, и только после завершения этого этапа можно будет, используя полученные результаты, приступить к решению этой проблемы для остальных трёх классов схем. Теперь поставим первую задачу данной работы: сформулировать базовые положения методологии изучения и проектирования цифровых схем и систем на компонентах жёсткой логики согласно методологии САПР.

Цифровые компоненты и схемы жёсткой логики разделяют на два типа: комбинационного и последовательностного. К цифровым компонентам и схемам комбинационного типа относятся: стандартные и тристабильные логические элементы, дешифраторы и шифраторы, мультиплексоры и демultipлексоры, схемы сравнения и контроля, преобразователи кодов, логические схемы для выполнения арифметических и логических операций. К цифровым компонентам и

схемам последовательностного типа относятся: триггеры, регистры, счётчики и элементы памяти.

Цифровые компоненты жёсткой логики выпускаются в виде промышленных серий. В этой работе использовались ИС серий КР1533 и КР1554 (зарубежный аналог – серия SN74). Для Украины отсутствие отечественных ИС современного уровня компенсируется доступностью зарубежной (прежде всего западной) элементной базы.

Вторая задача работы – изложение общей методологии логического моделирования класса схем жёсткой логики и частной методологии логического моделирования подкласса схем жёсткой логики комбинационного типа.

## **1. Современная методология изучения и проектирования цифровых схем и систем**

Разработка или выбор программы схемотехнического моделирования цифровых схем – это первый вопрос, который необходимо было качественно рассмотреть. В настоящее время заниматься разработкой такой программы с нуля контрпродуктивно, поэтому следовало выбрать из числа доступных и распространённых программ. Отметим, что в СССР в области логического моделирования наиболее успешно работали научные группы под руководством д. т. н. Норенкова И.П. [11] и д. т. н. Петренко А.И. [12, 13]. Однако на сегодняшний день в мире широко используются только три семейства программ: MicroCap [14], Electronics Workbench [15] и PSpice [16]. При этом в учебном процессе чаще всего используются программы семейства MicroCap (в интернете любой желающий может скачать бесплатно студенческую версию). В Национальном исследовательском университете «Московский электроэнергетический институт (технический университет)» на кафедре промышленной электроники установлено следующее общее требование к выполнению большинства курсовых работ и проектов и в обязательном порядке бакалаврских и магистерских работ: должны быть включены разделы с использованием программ из семейства MicroCap.

Современное проектирование цифровых систем базируется на методологии САПР электронных схем и систем. В работе в качестве базового математического обеспечения использованы методы и средства логического моделирования, реализованные в программах семейства MicroCap, что позволило авторам разработать свои инструменты логического моделирования и реализовать их в среде программ этого семейства. Такой подход позволяет, во-первых,

построить процесс изучения курсов, связанных с цифровыми схемами и компонентами, с методологией логического моделирования и, во-вторых, автоматически подключиться к процессу автоматизированного проектирования цифровых систем. Однако процесс освоения современных инструментальных и методологических средств логического моделирования требует более высокого качества математической подготовки студентов. Кроме того, выбранный подход позволяет надеяться, что через некоторое время можно будет построить логические модели сложных цифровых компонентов: микропроцессоров и микроконтроллеров, что значительно повысит качество усвоения изучаемого сложного материала.

При построении методологии изучения классических цифровых схем жёсткой логики необходимо учитывать тот факт, что реализация электронных цифровых систем проводится, как правило, на ПЛИС. Это позволяет отказаться (или значительно сократить) от изучения вопросов, относящихся к натурному эксперименту или проектированию на дискретных ИС. В первую очередь необходимо тщательно излагать вопросы функционального моделирования базовых логических схем, а во вторых – показывать моделирование реальных цифровых микросхем. Далее лабораторные практикумы на стендах можно сократить до минимума и больше времени уделять изучению свойств цифровых объектов на моделях, используя универсальные программы, например, семейства MicroCap.

Несколько замечаний о математическом обеспечении. Аксиомы и теоремы булевой алгебры и методы минимизации булевых функций являются математическим обеспечением для структурного синтеза функциональных схем всех цифровых объектов. Поэтому важно выработать как систему аксиом и теорем с точки зрения математической логики подачи материала, так и систему приобретения навыков доказательства при кратком изложении материала. На взгляд авторов более рационально остановиться на варианте 5 аксиом и 15 теорем (из них 4 дуальных теоремы для булевых функций  $n$  логических переменных). Аксиомы вводят понятие логической переменной, принимающей только одно из двух значений – 0 или 1, и основные логические операции: НЕ, И, ИЛИ. Традиционное изложение этого материала основано на нестрогом соответствии аксиомам и теоремам алгебры логики и теории множеств. Без серьёзного понимания законов алгебры и глубокого овладения методами минимизации булевых функций невозможно достичь хороших зна-

ний как в области изучения свойств цифровых объектов, так и в области создания их цифровых моделей. Другими словами, не будут освоены общие методики анализа и синтеза цифровых объектов.

Кроме отмеченного выше следует обратить внимание на полноту изложения и математическую и методологическую строгость получения результатов, ведь на сегодняшний день необходимо не только подавать студентам новую для них информацию о цифровых объектах и методах их моделирования. Современный специалист обязан уметь если не создавать новые модели, то, по крайней мере, уметь грамотно использовать существующие модели и знать, как их можно модифицировать. Этого можно достичь только при одном условии – высоком качестве уровня подготовки (в первую очередь в области математики).

Следует также обратить внимание на инструментальные средства логического моделирования, так как на сегодняшний день уже недостаточно беглого знакомства с современными программами схемотехнического моделирования, а необходимо сформировать математически и технически грамотного пользователя.

В заключении отметим, что имеющаяся доступная литература по универсальным программам [14–16] описывает только графический пользовательский интерфейс без детального анализа используемых моделей и методик работы с ними, а также реализованных методов логического моделирования. Кроме того отсутствует литература с профессиональными рекомендациями по моделированию и проектированию различных классов цифровых схем и систем.

Указанные выше замечания частично устранены авторами в двух тематических разработках: «Математическое обеспечение для анализа и синтеза логических комбинационных схем» и «Моделирование цифровых схем с помощью программ семейства MicroCap», которые включены в магистерскую работу выполненную Арянпур П. на кафедре промышленной электроники Национального технического университета Украины «Киевский политехнический институт» под руководством Приходько А.А. в 2008 году.

## **2. Моделирование и проектирование цифровых схем и систем**

Проектирование цифровых схем и систем осуществляется на основании одного из двух подходов:

- на основе стандартных блоков с помощью универсальных программ схемотехническо-

го моделирования семейства MicroCap, Electronics Workbench и OrCad с подсистемой PSpice;

- с помощью языков описания схем и синтеза ABEL, VHDL и Verilog.

Первый подход гармонично реализуется в предлагаемом подходе, а второй – чрезвычайно важен для подготовки современных специалистов высокой квалификации в области цифровой электроники. Ввиду большого объема и сложности материала второго подхода необходимо для его изучения предусмотреть отдельный курс. Кроме того, без знания языков описания логических схем невозможно изучать класс схем ПЛИС.

Будем различать два семейства цифровых компонентов (как и в программах семейства MicroCap): цифровые примитивы – это функционально-логические модели, выполняющие или отражающие основные свойства цифровых компонентов, схем или систем, и библиотеку реальных цифровых ИС.

Для логического моделирования используют два вида моделей компонентов:

- функционально-логические, которые отражают основную функцию, выполняемую логическим компонентом;
- структурно-логические, когда сложные цифровые компоненты представляют в виде структурных схем из более простых компонентов, для которых имеются функционально-логические модели.

Модели цифровых компонентов, а также логических элементов, схем и систем, строим используя принцип логического макро моделирования:

- на первом уровне разрабатываем библиотеку логических схемных элементов, отражающих базовые свойства и операции булевой алгебры и базовых логических элементов;
- на втором уровне формируем стандартный набор структурно– логических моделей для всех базовых цифровых объектов;
- на третьем уровне на основе моделей первого и второго уровней и методики логического макро моделирования можно разрабатывать модели новых сложных цифровых схем и систем.

На всех этапах разработки моделей цифровых компонентов и схем необходимо использовать соответствующие методики анализа и синтеза логических схем. Анализ и синтез – базовые процедуры процесса проектирования цифровых схем и систем. Анализ заключается в получении формального описания реализуемых логических функций для заданной схемы и системы, а синтез – в получении логической схемы,

обеспечивающей требуемое функциональное поведение.

Под проектированием цифровых схем и систем будем понимать многоэтапный процесс создания описаний проектируемого объекта от выдачи технического задания до создания полного комплекта конструкторской и технологической документации.

Укрупнённо проектирование цифровой схемы или системы в соответствии с положениями построения САПР включает семь этапов:

- неформальное (словесное или мысленное) описание логической схемы или системы в соответствии с выданным техническим заданием;
- формализация описания схемы (системы) – состоит из указания входных и выходных сигналов схемы и задания её функционального поведения посредством специальных таблиц и (или) логических уравнений;
- синтез логической схемы (системы), который заключается в получении эквивалентной логической схемы как вручную, так и с помощью программ (основан на специальном математическом обеспечении);
- тестирование логической схемы и системы (используются два метода: по принципу «годен – не годен» или с помощью диагностического тестирования);
- анализ логической схемы и системы (можно проводить для простых схем вручную, но лучше использовать программы схемотехнического моделирования);
- конструкторское проектирование (заканчивается описанием схем электрических принципиальных);
- технологическое проектирование (заканчивается описанием монтажных плат).

Если результаты этапов тестирования и анализа неудовлетворительные, то корректируются параметры объекта на этапах формализации описания и синтеза.

Детализация процедур синтеза и анализа для цифровых схем и систем жёсткой логики сильно зависит от их типа: комбинационного или последовательностного.

По уровню детализации работы цифровых компонентов и схемы (системы) в целом различают следующие варианты логического моделирования:

- функционально-логическое, когда предполагают, что все цифровые компоненты идеализированы для статического уровня сигнала (либо 0, либо 1) и динамического (компоненты задаются с нулевой временной задержкой) режимов;

- временное функциональное, когда вычисляются и учитываются задержки для всех путей прохождения логических сигналов в наихудшем случае;
- аналого-цифровое, когда исследуемая эквивалентная схема имеет три части – аналоговую, цифровую и схемы сопряжения между аналоговыми и цифровыми частями.

### 3. Логическое моделирование цифровых схем комбинационного типа

Разработка моделей цифровых объектов первого и второго уровней для логического моделирования схем комбинационного типа выполнялась по методике абстрактного синтеза.

Абстрактным синтезом комбинационных логических схем называют последовательность этапов, в результате которых получают функциональную схему базовых логических элементов или функциональных узлов комбинационного типа по результатам синтеза таблицы истинности и логических уравнений.

Функциональные схемы – это схемы, в которых показаны связи между логическими схемными элементами, а сами логические схемные элементы представлены в виде условных графических обозначений.

Логическими схемными элементами (ЛСЭ) комбинационного типа будем называть элементы, которые реализуют элементарные булевы функции. При этом реализация каждого ЛСЭ должна быть представлена в трёх вариантах: полном базисе И, ИЛИ, НЕ, основном базисе И–НЕ и дуальном базисе ИЛИ–НЕ. Все варианты равноправны, переход от полного базиса к основному или дуальному базису осуществляется с помощью правил де Моргана [3].

Абстрактный синтез ЛСЭ комбинационного типа выполняем в соответствии со следующим алгоритмом:

Этап 1. Словесная формулировка элементарных булевых функций.

Этап 2. Построение таблицы истинности ЛСЭ комбинационного типа по словесной формулировке.

Этап 3. Запись базового логического уравнения.

Этап 4. Выбор базового условного графического обозначения.

Этап 5. Задание второго и третьего базисов, отличных от базового базиса.

Этап 6. Преобразование базового логического уравнения для записи логических уравнений для второго и третьего базисов.

Этап 7. Построение эквивалентной функциональной схемы для второго и третьего базисов, отличных от базового базиса.

Согласно приведенному выше алгоритму разработан базовый набор ЛСЭ (БНЛСЭ) комбинационного типа для базовых логических элементов одной и двух переменных во всех трёх логических базисах. Свойства этих элементов представлены в виде таблиц истинности, на основе которых строятся логические уравнения для трёх базисов, а затем реализуются в виде эквивалентной функциональной схемы для каждого базиса.

БНЛСЭ включает 8 стандартных и 4 нестандартных базовых логических элемента: инвертор (НЕ), буфер (повторитель), И, ИЛИ, И–НЕ, ИЛИ–НЕ, Исключающее ИЛИ, Исключающее ИЛИ–НЕ, Запрет по X1, Запрет по X2, Импликация от X1 к X2, Импликация от X2 к X1 (всего 36 логических моделей).

Трестабильные логические элементы одной и двух переменных отличаются от обычных тем, что позволяют совмещать стандартный режим работы логических схем и режим высокого импеданса, а также аналогично моделируются. Моделирование трестабильных логических элементов проверено на примере трестабильной буферной шины, пропускающей или не пропускающей сигналы с входа шины на выход.

Двухуровневую логику, то есть последовательное выполнение двух логических операций (И–ИЛИ–НЕ, ИЛИ–И–НЕ) можно реализовать, если, используя модели логических элементов И, ИЛИ, И–НЕ и ИЛИ–НЕ, построить макромодели элементов двухуровневой логики (третий уровень принципа логического макромоделирования).

Синтез и анализ логических схем комбинационного типа выполнялся на базе общего математического обеспечения – булевой алгебры.

Абстрактный синтез комбинационной логической схемы состоит из следующих этапов:

- словесная формулировка функций логической схемы;
- составление таблицы истинности по словесной формулировке;
- запись логического уравнения по таблице истинности в виде канонической суммы или канонического произведения логической функции;
- минимизация логического уравнения;
- выбор одного из логических базисов: полного И, ИЛИ, НЕ, основного И–НЕ, дуального ИЛИ–НЕ;
- преобразование логического уравнения с использованием аксиом и теорем булевой

алгебры (в первую очередь правил де Моргана) для реализации в выбранном базисе;

- построение функциональной схемы по последнему варианту логического уравнения для выбранного базиса.

Для минимизации (упрощения) логического уравнения применяют четыре метода: алгебраическое упрощение с помощью аксиом и теорем булевой алгебры, метод минимизации с помощью Карт Карно, табличный и программный методы Квайна-Мак-Класки. Первые два метода рекомендуются использовать для логических функций с количеством логических переменных не более пяти, а два последующих – теоретически не имеют ограничений.

Изложенный выше алгоритм синтеза апробирован для всех классов функциональных узлов и компонентов комбинационного типа:

- Для дешифраторов на таких объектах: двоичных линейных дешифраторах  $2 \times 4$  и  $3 \times 8$  и на микросхеме  $74 \times 138$ , матричном дешифраторе на 64 выхода, пирамидальном дешифраторе на 8 выходов, семисегментном дешифраторе на микросхеме  $74 \times 49$ .
- Для шифраторов на таких объектах: двоичном шифраторе  $8 \times 3$ , восьмивходовом приоритетном шифраторе и таком же шифраторе на микросхеме  $74 \times 148$ .
- Для мультиплексоров на таких объектах: функциональных узлах  $2 \times 1$  и  $4 \times 1$  и мультиплексорах на микросхемах  $74 \times 151$  и  $74 \times 157$ .
- Для демультиплексоров на таких объектах: функциональных узлах  $1 \times 2$  и  $1 \times 4$  и демультиплексора на микросхеме  $74 \times 138$ .
- Для преобразователей кодов – на примере преобразователя кода 7442.
- Для комбинационных арифметических схем на таких объектах: одноразрядных сумматорах (четвертьсумматоре, полусумматоре, полном сумматоре) и вычитателях (полувычитателях, полном вычитателе), многоразрядных сумматорах и вычитателях, сумматоре на микросхеме  $74 \times 283$ , арифметико-логическом устройстве на микросхемах  $74 \times 381$  и  $74 \times 382$ .
- Для цифровых компараторов на таких объектах: однобитовом и двухбитовом компараторе и многоразрядных компараторах на микросхемах  $74 \times 85$  и  $74 \times 682$ .

Всего примеров: 19 для функциональных узлов и 12 на цифровых компонентах. Все модели функциональных узлов, цифровых компонентов, а также логических элементов, упомянутых в статье, протестированы с помощью

программы MicroCap версии 6.0 и подробно описаны в магистерской работе Арянпур П.

## Выводы

1. Предложена методология изучения курсов цифровой схемотехники в технических университетах, разработанная на идеологии САПР, которая позволяет на единых принципах объединить процессы изучения и проектирования электронных цифровых компонентов, схем и систем.

2. На основе анализа процессов, как традиционного обучения, так и современного проектирования четырёх классов схем цифровой схемотехники определена последовательность решения поставленной проблемы: на первом этапе рассматривается только класс схем жёсткой логики с учётом свойств оставшихся трёх классов схем, а затем полученные результаты логического моделирования этого этапа применить для других классов схем. Такой подход возможен, так как все классы цифровых схем имеют единое базовое математическое обеспечение.

3. Для класса схем жёсткой логики выделены все классы и подклассы функциональных узлов и компонентов комбинационного и последовательностного типов.

4. В качестве инструментального средства изучения и проектирования цифровых объектов обоснован выбор программ семейства MicroCap.

5. Даны рекомендации по выбору системы аксиом и тождеств булевой алгебры и по методике преподавания математического обеспечения.

6. Сформулирована методология для логического моделирования и проектирования схем жёсткой логики в соответствии с методологией САПР.

7. Сформулирована методика разработки функционально-логических моделей логических элементов, на основе которой сформирована библиотека логических моделей (БНЛЭСЭ), которая на данный момент имеет 36 моделей. Показаны методики расширения библиотеки за счёт тристабильных логических элементов и элементов двухуровневой логики.

8. Сформулирована методика абстрактного синтеза комбинационной логической схемы, на основании которой апробированы функционально-логические модели 19 функциональных узлов и 12 цифровых компонентов комбинационного типа из всех 9 подклассов, из которых можно сформировать библиотеку типовых моделей функциональных узлов и цифровых компонентов подобно как в САПР создают библиотеку стандартных проектных решений.

9. Тестирование всех приведенных в статье логических моделей доказывает справедливость принятой методологии изучения и проектирования цифровых компонентов, схем и систем в курсах цифровой схемотехники.

### Литература

1. *Згуровський М.З.* Болонський процес: головні принципи та шляхи структурного реформування вищої освіти України. – К.: НТТУ «КПІ», 2006. – 544 с.
2. *Схемотехніка* електронних систем: У 3 кн. Кн. 2. Цифрова схемотехніка: Підручник / В.І. Бойко, А.М. Гуржій, В.Я. Жуйков та ін. – 2-ге вид., допов. і переробл. – К.: Вища школа, 2004. – 423 с.
3. *Справочник* по цифровой схемотехнике. Зубчук В.И., Сигорский В.П., Шкуро А.Н. – К.: Техника, 1990. – 448 с.
4. *Рябенський В.М., Жуйков В.Я., Гулий В.Д.* Цифрова схемотехніка: Навч. посіб. – Львів: «Новий Світ-2000», 2009. – 736 с.
5. *Угрюмов Е.П.* Цифровая схемотехника. – СПб.: БХВ–Петербург, 2004. – 528 с.
6. *Фрике К.* Вводный курс цифровой электроники. – М.: Техносфера 2004. – 432 с.
7. *Бабич Н.П., Жуков И.А.* Компьютерная схемотехника. Методы построения и проектирования: Учебное пособие. – К.: «МК–Пресс», 2004. – 576 с.
8. *Мікропроцесорна* техніка: Підручник / Ю.І. Якименко, Т.О. Терещенко, Є.І. Сокол, В.Я. Жуйков, Ю.С. Петергеря; За ред. Т.О. Терещенко. – К.: Видавництво «Політехнік», 2003. – 440 с.
9. *Трамперт В.* AVR-RISC микроконтроллеры: Пер. с нем. – К.: «МК–Пресс», 2006. – 464 с.
10. *Грушвицкий Р.И., Мурсиев А.Х., Угрюмов Е.П.* Проектирование систем на микро-схемах программируемой логики. – СПб.: БХВ–Петербург, 2002. – 608 с.
11. *Системы* автоматизированного проектирования в радиоэлектронике: Справочник / Е.В. Авдеев, А.Т. Ерёмин, И.П. Норенков, М.И. Песков; Под ред. И.П. Норенкова. – М.: Радио и связи, 1986. – 368 с.
12. *Петренко А.И.* Основы автоматизации проектирования. – К.: Техника, 1982. – 295 с.
13. *Петренко А.И., Цурин О.Ф., Киселёв Г.Д.* Автоматизация проектирования цифровых схем. – К.: Вища школа, 1978. – 150 с.
14. *Разевиг В.Д.* Система моделирования Micro–Cap 6. – М.: Горячая линия–Телеком, 2001. – 344 с.
15. *Карлащук В.И.* Электронная лаборатория на IBM PC. Программа Electronics Workbench и её применение. – 2-е изд., доп. и перераб. – М.: Солон–Р, 2001. – 726 с.
16. *Разевиг В.Д.* Система проектирования OrCAD 9.2 – М.: Солон–Р. 2001. – 519 с.