

Министерство образования и науки РФ

Санкт-Петербургский государственный
электротехнический университет «ЛЭТИ»

Проектирование цифровых узлов

Методические указания
к курсовому проектированию

Санкт-Петербург
Издательство СПбГЭТУ «ЛЭТИ»
2011

УДК 000000000

Проектирование цифровых узлов: Методические указания к курсовому проектированию / Сост.: Р.И. Грушвицкий, Е.П. Угрюмов. СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2011. 00 с.

Содержат рекомендации по разработке специализированного цифрового узла и набор заданий на курсовое проектирование. Разработка узла на элементной базе ПЛИС предполагает рассмотрение возможных вариантов его реализации, моделирование работы вариантов и анализ их свойств средствами пакета Quartus II, выбор наилучшего варианта по заданному критерию качества и оформление для него технической документации. Предназначены для студентов по направлению подготовки 230100.62 "Информатика и вычислительная техника", профиль "ЭВМ, комплексы, системы и сети".

Утверждено
редакционно-издательским советом университета
в качестве методических указаний

© СПбГЭТУ «ЛЭТИ», 2011

Разработка специализированного цифрового узла (устройства), выполняемая в рамках курсового проекта, имеет целью:

- научить студента ориентироваться в разнообразии возможностей цифровой техники при проектировании специализированных устройств и эффективно применять современную элементную базу;
- показать многовариантность решений при синтезе цифровых узлов и устройств, научить выбирать вариант их реализации, отвечающий требованиям ТЗ и наилучший по заданным критериям качества;
- научить студента пользоваться возможностями современных САПР при проектировании устройств на базе микросхем программируемой логики;
- научить студента технически грамотно описывать функционирование разрабатываемых узлов и устройств и правильно оформлять техническую документацию.

1. Состав и порядок выполнения курсового проекта

Содержание работ при курсовом проектировании иллюстрируется рис.1.

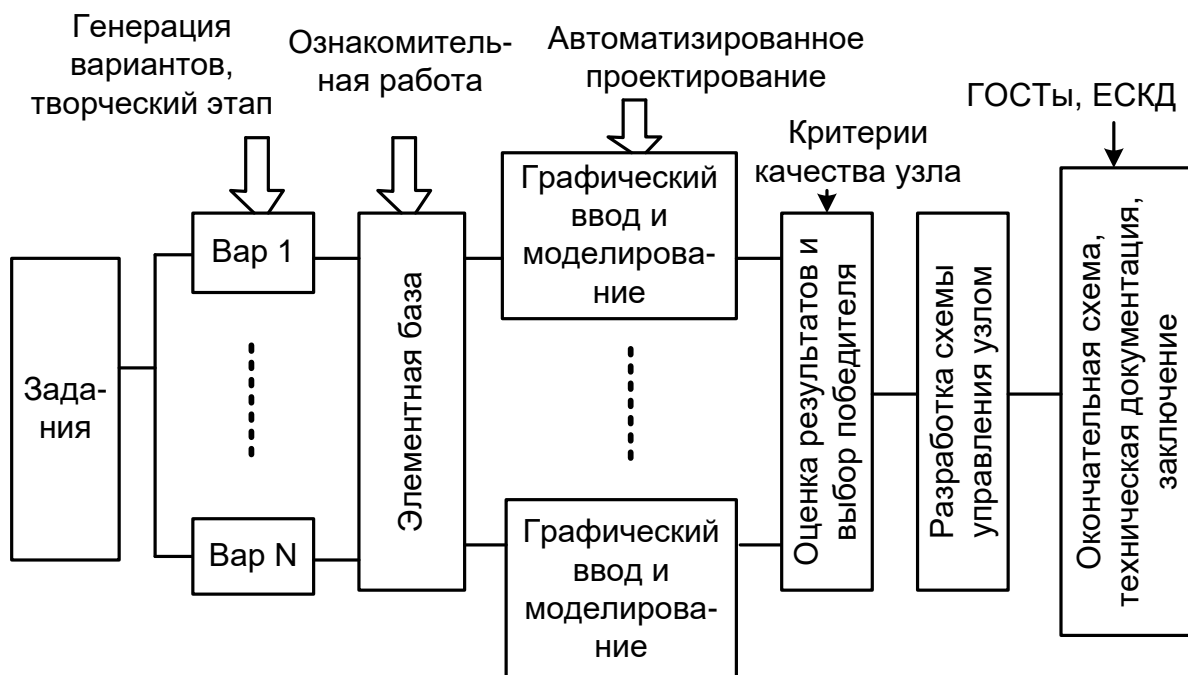


Рис.1.

Выполнение задания на курсовой проект предполагает:

- Разработку функциональной схемы проектируемого узла в нескольких вариантах;
- Разработку функциональной схемы интерфейса узла с ведущей микропроцессорной системой;
- Ввод принципиальной электрической схемы узла с интерфейсным блоком для реализации в ПЛИС с помощью САПР;
- Компиляцию полученного проекта;
- Тестирование разработанного узла в рамках САПР;
- Выбор наилучшего варианта узла с учетом заданного критерия качества;
- Разработку принципиальной электрической схемы типового элемента замены (ТЭЗа), содержащего спроектированный узел и вспомогательные схемы.

Курсовой проект состоит из принципиальной электрической схемы, разработанной для заданного варианта проектируемого узла, и пояснительной записки.

Материал пояснительной записки рекомендуется располагать в следующем порядке:

1. Задание на проектирование узла.
2. Описание предлагаемых вариантов реализации узла. Варианты следует рассматривать на уровне укрупненных функциональных схем. Их число N в зависимости от задания равно $2 \dots 3$.
3. Описание основных элементов библиотеки САПР Quartus II и стандартных микросхем, необходимых для реализации вариантов узла. В пояснительной записке следует дать описание применяемых микросхем, указать назначение их выводов. Не следует перегружать этот раздел проекта. Его объем не должен превышать $3 \dots 4$ страниц.
4. Описание процесса синтеза и моделирования работы предложенных вариантов средствами САПР Quartus II при графическом вводе проектов и полученных при этом результатов.
5. Определение предельной допустимой частоты генератора тактовых импульсов (ГТИ).
6. Разработка интерфейса сопряжения схемы узла с процессорной системой, для которой проектируемый узел является внешним устройством.

Устройство, содержащее интерфейс вместе с проектируемым узлом, размещается в ПЛИС.

7. Подробное описание функционирования узла с использованием необходимых временных диаграмм.
8. Полная принципиальная электрическая схема типового элемента замены с учетом реализации узла на выбранной ПЛИС, разработки генератора тактовых импульсов, разводки цепей питания, фильтрации напряжений питания и т.д.
9. Краткое заключение по проделанной работе.
10. Список использованной литературы, оформленный согласно требованиям стандарта.

Техническая документация, относящаяся к проекту, оформляется в соответствии с требованиями ГОСТов.

2. Рекомендации по выполнению основных этапов проектирования узла

2.1. Задание на проектирование узла.

В пояснительной записке приводится полный текст задания, выдаваемого студенту на первом практическом занятии по курсовому проектированию.

2.2. Сравнительный анализ предлагаемых вариантов реализации узла.

Рассмотрение возможных вариантов реализации узла является этапом работы, имеющим творческий изобретательский характер. Выполнение этого этапа требует знания предмета, технической эрудиции и инженерной интуиции разработчика. Преимущественно эвристический подход к поиску вариантов решения задачи объясняется тем, что формальный синтез устройства в целом даже при его теоретической возможности имел бы огромную трудоемкость.

Обычно при поиске вариантов построения узла используют известные из предыдущего опыта разработчиков решения (подсхемы) с их сочетаниями и модификациями применительно к поставленной задаче. В то же время формальные методы синтеза могут и должны быть применены при проектировании отдельных локальных частей устройства.

На этапе рассмотрения вариантов решения задачи важно продемонстрировать принципы их работы, не загромождая изложение второстепенными деталями, т. к. для этого этапа важно показать принципиальную работоспособ-

ность и основные качества вариантов. Схема каждого варианта должна быть раскрыта до уровня типовых функциональных узлов вычислительной техники типа "регистр", "счетчик", "дешифратор", "мультиплексор", "триггер" без их излишней детализации, которая затемняет главные процессы функционирования схемы. При определенной сложности проект целесообразно строить в форме нескольких схем, построенных по иерархическому принципу (ведущие и ведомые фрагменты). Главная задача этого этапа изложения материала — наглядное представление *идеи построения и функционирования схемы*.

На уровне укрупненных функциональных схем следует разработать 2...3 варианта реализации узла и описать их работу, пользуясь при необходимости упрощенными временными диаграммами.

2.3. О критериях качества разрабатываемых вариантов

Результаты синтеза технических устройств, как правило, неоднозначны и возможна их реализация в нескольких сопоставимых по качеству вариантах.

Выбор наилучшего для данных условий варианта определяется критериями качества, согласованными с заказчиками проекта. В качестве критерия качества, экстремального значения которого добивается разработчик, может выступать какая-либо одна особо важная для пользователя техническая характеристика — *частный* критерий качества (минимум аппаратных затрат, максимум быстродействия, минимум потребляемой мощности, максимум помехоустойчивости и т. п.). Однако обычно требуется учет нескольких характеристик и оценка для их совокупности — *интегральный* критерий качества.

Формирование интегрального критерия — чрезвычайно ответственная задача, не имеющая формального решения. *В любую форму интегрального критерия качества входят коэффициенты, назначаемые субъективно*. Поэтому в практике проектирования сложные интегральные критерии качества не популярны. Достаточно признанным интегральным критерием можно, пожалуй, считать лишь критерий AT , где A — аппаратная сложность устройства, T — время получения результата. Наилучшей по интегральному критерию AT или A/F является схема с его минимальным значением. Но и здесь проявляется общий недостаток интегральных критериев — в них уменьшение одного частного критерия может быть скомпенсировано ростом другого, что формально равноценно, но далеко не всегда разумно.

Аппаратная сложность схемы (А) при реализации на ПЛИС выявляется с помощью САПР после компиляции проекта и оценивается количеством использованных схемных ресурсов кристалла. По первоначальной функциональной схеме аппаратную сложность реализации в ПЛИС можно оценить лишь очень приближенно.

Время решения задачи T оценивается минимальным временным интервалом между сменами наборов входных данных. После подачи входных данных идет их обработка, по окончании которой на выходе формируется результат. Только после завершения этого процесса можно подавать на входы схемы новый набор входных данных (сквозная синхронизация).

Для оценки интервала T необходимо найти по цепи задержек наиболее длинный по времени путь распространения сигналов от входа ПЛИС к её выходу. На этом пути суммируются значения задержек элементов, соответствующие выбранной ПЛИС, и учитываются (при их наличии) интервалы, связанные с тактированием процессов в схеме. Результат можно получить при моделировании работы схемы с помощью САПР.

Специально учитываются ситуации неготовности устройств к обмену данными с микропроцессорной системой, увеличивающие интервал T . Для блокировки получения неверных результатов или пропуска запроса на обработку устройство сообщает о своей готовности сигналом Ready.

Для схем, содержащих тактируемые элементы памяти, важной характеристикой быстродействия служит предельно допустимая частота повторения импульсов F .

2.4. Применение программируемых ЗУ

Известно, что ЗУ, реализующие табличные методы представления функций, могут служить средствами решения многих задач обработки информации. Табличные формы представления функций достаточно универсальны, то же можно сказать и о возможностях построения цифровых узлов на основе программируемых ЗУ. Приводимое ниже пояснение применимости ЗУ для построения проектируемых узлов связано с возможно недостаточными сведениями об этом, полученными из лекционного курса.

Для воспроизведения таблиц служат микросхемы программируемых ЗУ (PROM, EPROM-OTP, EPROM, EEPROM, FLASH). В эти ЗУ содержимое записывается в специальном режиме, а в рабочем режиме производится только чтение данных. В микросхемах ПЛИС, кроме того, функции таблиц моде-

лируются с помощью ОЗУ, в которые загружается и сохраняется на время решения таблица (хотя и исчезает при снятии питания). Для применения ЗУ в данном курсовом проекте можно использовать типовые элементы памяти (ROM, RAM, FIFO и т.д. в закладке megafunctions – storage САПР).

Внешняя организация ROM приведена на рис. 2, а. В режиме с постоянным разрешением работы по входу EN память работает как комбинационное устройство. При этом (рис. 2, б) смена адреса вызывает изменение выходных данных с некоторой задержкой доступа по адресу $t_{A(A)}$ ($t_{CY(A)}$ — длительность цикла чтения). Код адреса $A_{n-1}A_{n-2}...A_0$ определяет ячейку памяти, из которой считываются данные $D_{m-1}D_{m-2}...D_0$. Число адресов памяти равно 2^n , а емкость ЗУ составляет $2^n \times m$ бит.

Для воспроизведения функций времени генератор частоты задает отметки времени. Сигналы генератора подаются на счетчик, преобразующий последовательность отметок времени в последовательность кодов. Коды служат адресами памяти, так что для каждого интервала времени с выходов ЗУ считываются значения функций соответственно его запрограммированному содержанию.

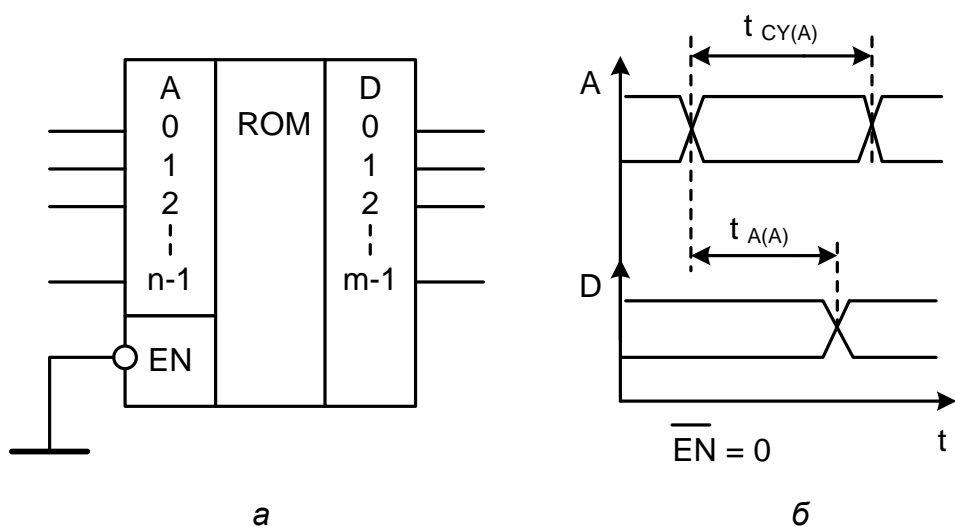


Рис.2.

2.5. Сопряжение проектируемого узла с устройством управления (интерфейсом процессорной системы)

Сигналы устройства управления (УУ) определяют такие действия как адресация объектов, приведение схемы в исходное состояние, задание режима работы, пуск и останов, прием промежуточных и окончательных результатов (рис. 3).



Рис.3

Реакция узла на команду останова предусматривается такой: при циклической работе устройства оно останавливается не мгновенно, а *по завершении текущего цикла работы*.

Сигнал сброса приводит схему в исходное состояние (если это необходимо). Адреса подаются для доступа к регистрам схемы (в том числе и одноразрядным), которым предназначен соответствующий сигнал (код режима, пуск, останов и др. Для этого восьмиразрядные адреса декодируются схемой селектора адреса, выходной сигнал, которого SEL открывает сигналу путь к соответствующему регистру. Сигнал Ready (готовность) поступает от управляемого устройства, при его отсутствии процессы обмена с устройством приостанавливаются до появления сигнала Ready. При выполнении некоторых заданий сигнал готовности может понадобиться. Способы формирования этого сигнала рассмотрены в работе [1].

Проектируемый узел рассматривается как внешнее устройство (ВУ) процессорной системы. В этом случае функции УУ выполняет интерфейс системы (в данном случае Microbus [3]). Для взаимодействия с ВУ процессор реализует две команды — команду ввода (чтения порта) IN(address_port) и команду вывода (записи в порт) OUT(address_port). Выработка сигналов, выполняющих требуемые действия, может быть проиллюстрирована временной диаграммой цикла вывода (рис.4).

Цикл ввода отличается от цикла вывода подачей строба $\overline{\text{IOR}}$ вместо строба $\overline{\text{IOW}}$ и более поздним появлением вводимых данных. Для выработки управляющих импульсов по командам IN и OUT можно использовать совпадения активных значений сигналов пары SEL и $\overline{\text{IOR}}$ либо SEL и $\overline{\text{IOW}}$ (см. схемы безусловного и условного программного ввода/вывода в работе

[1]). Повторяя команды, можно выработать требуемую последовательность управляющих импульсов. Каждый последующий импульс появляется в следующем командном цикле.

Разработанный вариант управления узлом следует проиллюстрировать полной функциональной схемой устройства с описанием его работы и временными диаграммами. Составление подобного «скелета» проекта (его архитектуры) завершает этап раннего планирования и позволяет перейти к реализации проекта в ПЛИС с помощью САПР.

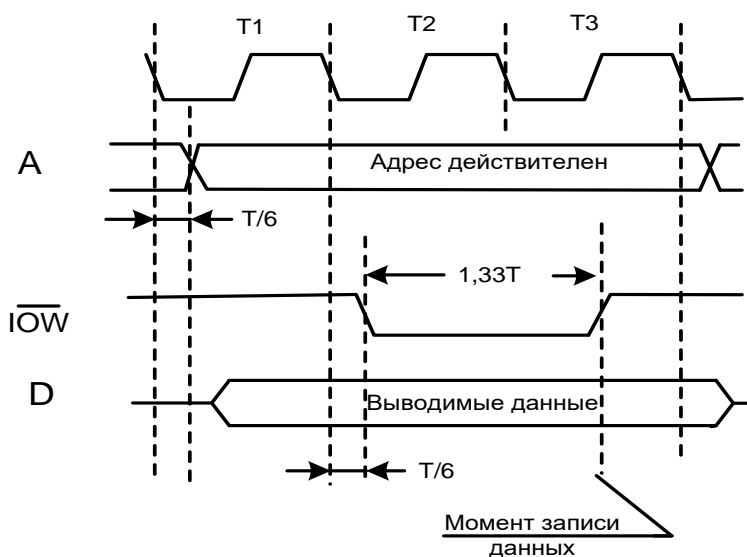


Рис.4

2.6. Этап технической реализации проектируемого устройства

Выбор элементной базы. После подготовительного этапа можно переходить к технической реализации проекта. Одним из первоочередных вопросов, встающих перед разработчиком электронной аппаратуры, является вопрос о выборе элементной базы. Существует много схемотехнических и технологических вариантов ее реализации. С этой точки зрения выбор мог бы быть разнообразным. Однако практически разработка устройств ведется в условиях заданной элементной базы. (обычно техническое задание на проект заставляет остановиться на применении определенных элементов).

Разработчику устройств, реализованных на кристалле, приходится, прежде всего, выбирать способ реализации устройства: микросхемы программируемой логики (ПЛИС) или специализированные ИС (ASIC). Технология ПЛИС используется не только самостоятельно, но и в качестве предварительной (прототипной) реализации даже при разработке ASIC. Поэтому

рассмотрение процесса разработки цифровых устройств на ПЛИС средствами САПР представляется особенно актуальным.

Кроме выбора типа ИС при планировании выполняется и ряд подготовительных действий: выбор расположения входных и выходных контактов, определение временных и мощностных ограничений и др. Опираясь на возможности САПР, разработчик еще до этапа конструкторско-технологического проектирования может, например, получить предварительные оценки затрачиваемой мощности и временных характеристик.

Выбор ПЛИС как средств реализации проекта предопределяет допустимые семейства ИС и возможные САПР. Для курсового проекта круг возможных решений искусственно сокращен. Рекомендуемый выбор — фирма производитель Altera, САПР — Quartus II, семейство микросхем — MAX3000 или MAX II.

. **Реализация проекта в ПЛИС.** Последовательность работы в САПР носит название проектный поток (Design Flow) и практически не зависит от сложности проекта. В основе проектного потока лежат этапы: ввод в той или иной форме информации о фрагментах схемы и их соединениях между собой, компиляция проекта и анализ полученных результатов. Типовая последовательность работы в САПР приведена в разделе 3. Подробные материалы о работе в среде Quartus II можно найти в литературе [2].

В курсовом проекте в основном используются два типа редакторов исходных фрагментов: графический редактор (Block Editor) и редактор создания конечных автоматов (State Machine Editor). В отдельных случаях может использоваться символьный редактор (Block Symbol Editor). Подробно работа с редакторами описана в [2], кратко изложена ниже.

Для выбора элементов САПР предлагает три типа библиотек: мегафункции (megafunctions), примитивы (primitives) и прочие (others). В свою очередь мегафункции содержат библиотеки элементов: арифметических, вентильных, ввода/вывода и памяти. Отличительная особенность мегафункций — настройка параметров типовых узлов. Мегафункции позволяют использовать многоразрядные (шинные) подключения. Примитивы предлагают выбор среди библиотек: буферных и логических элементов, различных типов контактов, элементов памяти и оформления электрических схем.

Особо интересна библиотека прочих элементов (others). В ней, в частности, расположена библиотека под именем maxplus2, в которой помимо

набора типовых элементов вычислительной техники расположены адаптированные к реализации на ПЛИС аналоги ИС малой степени интеграции серий 74/54 фирмы Texas Instruments (TI), популярных среди разработчиков электронной аппаратуры.

В пояснительной записке должно быть дано описание наиболее значимых для проекта применяемых элементов с их условными графическими обозначениями и указанием назначений их выводов. Объем такого описания *не должен превышать 3...4 страниц.*

2.7. Разработка принципиальной электрической схемы интерфейсной платы МП системы

Разработка устройства в курсовом проекте завершается проектированием электрической схемы, содержащей сконфигурированную ПЛИС, ее интерфейс с микропроцессорной системой, а также элементы обеспечивающие работу ПЛИС (генератор тактовых импульсов, разъем для входных, выходных и интерфейсных сигналов, элементы загрузки или реконфигурирования (разъем JTAG), элементы образования питающих напряжений (преобразователи DC-DC) и их фильтрации).

На этом этапе производится переход от составленной ранее функциональной схемы к схеме принципиальной электрической. При этом необходимо решить ряд вопросов. Схема принципиальная электрическая (именно такой порядок слов применяют *в официальных документах* согласно ГОСТу) должна исчерпывающим образом описывать состав всех имеющихся в ней элементов и их межсоединения.

Отдельно рассматривается вопрос организации цепей питания схемы, который не затрагивается на уровне разработки функциональных схем. В связи с существованием помех из-за связей между элементами по цепям питания, напряжения питания в цифровых устройствах *обязательно фильтруются* [1]. Самый распространенный способ фильтрации напряжений питания — подключение к шине питания сглаживающих конденсаторов. Обычно для каждого ТЭЗ (на входе схемы) вводится один низкочастотный электролитический конденсатор большой емкости (10 или более микрофарад), и около каждой микросхемы (или для группы из определенного числа микросхем) устанавливается высокочастотный (керамический) конденсатор с емкостью порядка долей микрофарады. Уточненные цифры для конкретных серий элементов приводятся в справочниках.

Элементы с открытым стоком или открытым коллектором требуют подключения на выходах *внешних резисторов*, сопротивление которых определяется расчетом [1]. Резисторы можно выполнять навесными или использовать резисторные матрицы или блоки. Выходы микросхем с третьим состоянием для повышения их помехоустойчивости также обычно снабжаются резисторами, подключенными к источникам питания, сопротивления этих резисторов не рассчитываются и принимаются достаточно большими (порядка сотен килоом).

При применении в одной и той же схеме элементов разных схемотехнологий (ТТЛ и КМОП) необходимо *электрическое согласование логических уровней*. Непосредственная подача сигналов от элементов КМОП на элементы ТТЛШ разрешается, а при подаче сигналов от элементов ТТЛШ на элементы КМОП требуется подтянуть уровни логической единицы к напряжению питания U_{CC} , для чего между элементами включается специальная резисторная цепочка [1].

2.8. Выбор и разработка генератора тактовых импульсов.

Большинство узлов, разрабатываемых в курсовом проекте, содержат генератор тактовых импульсов ГТИ. Выбор технической реализации генераторов тактовых импульсов зависит от целого ряда факторов. Определяющим фактором является требуемая стабильность частоты выходных импульсов.

Наибольшую стабильность обеспечивают схемы, содержащие в своем составе кварцевые резонаторы. Промышленностью выпускается достаточно обширный круг таких ИС. Основной недостаток этих схем – достаточно высокая стоимость (несколько долларов).

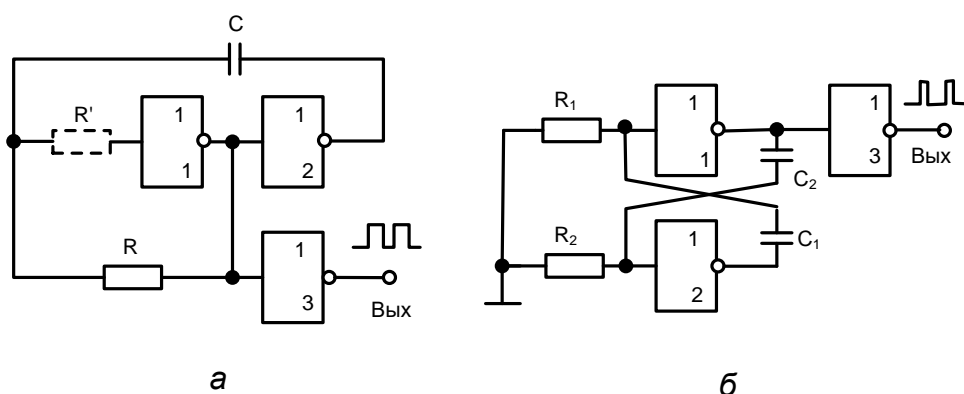


Рис. 5

При допустимой нестабильности частоты порядка единиц процентов применяют простые кольцевые генераторы [1] или RC-генераторы, с вре-

мязадающими элементами в виде цепочек, содержащих конденсаторы и резисторы.

В схеме рис. 5, *а* времязадающая цепочка RC определяет частоту выходных импульсов, равную приблизительно $0,53/RC$ при скважности около 0,6. Для стандартных элементов ТТЛ вследствие значительных входных токов сопротивление R должно быть малым. Рекомендуется выбирать его в пределах 200...400 Ом. Резистор R' в схемах на элементах ТТЛ не требуется.

При реализации схемы рис. 5, *а* на элементах КМОП сопротивление резистора R ввиду практического отсутствия входных токов инверторов может варьироваться в широких пределах (до сотен килоом). Резистор R' включается для защиты инвертора 1 от пробоя, его номинал выбирается в 2...3 раза больше номинала R . Времязадающей цепью по-прежнему является цепочка RC . При необходимости получить импульсы с другой скважностью (несимметричные) можно применить схему рис. 5, *б* с двумя времязадающими цепочками R_1C_1 и R_2C_2 . Для этой схемы длительности импульса и паузы равны соответственно $0,8R_1C_1$ и $0,8R_2C_2$. Рекомендации по диапазону изменения сопротивлений для схем на элементах ТТЛ и КМОП остаются теми же, что и для схемы рис. 5, *а*. В обеих схемах генераторов инверторы 3 не входят в их основную часть, а играют роль формирователей, улучшающих форму генерируемых колебаний, приближая ее к прямоугольной.

2.9. О некоторых правилах оформления схем

Конструктивно проектируемый узел выполняется в виде печатной платы, которая для связей узла с другими устройствами и подключения питания снабжается соединителем (разъемом). В курсовом проекте разъем отображается табличкой с двумя столбцами (рис. 6, *а*). Все входные и выходные связи, питающие напряжения и, возможно, контрольные точки выводятся на разъем.

При выполнении принципиальной электрической схемы допустимо и целесообразно применение групповых линий связи (рис. 6, *б*). Проводники следует отмечать именами сигналов (на рисунке указано, что проводник передает сигнал $A0$). Вблизи микросхемы над линией проводника указывается номер вывода, к которому подключается проводник.

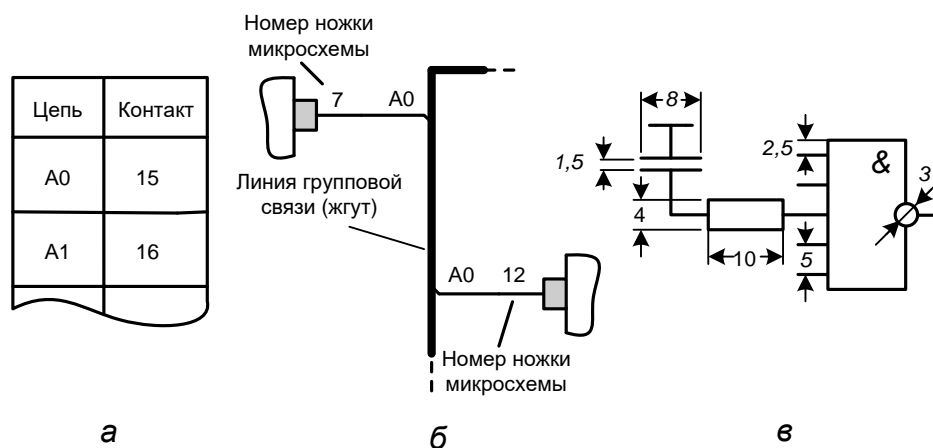


Рис. 6.

Размеры микросхем и навесных элементов должны соответствовать рис. 6, в. Принципиальная электрическая схема выполняется согласно требованиям стандарта ЕСКД ГОСТ 2.743-91.

Схема размещается на листе, формат которого определяется ее сложностью. Перечень элементов допускается выполнять и на отдельных листах формата А4. Перечисление элементов в перечне идет по начальной букве обозначения элемента в порядке латинского алфавита.

3. Работа в САПР Quartus II

3.1. Последовательность работы в САПР Quartus II

Прежде всего следует стандартными средствами ОС подготовить директорию для файлов проекта. Далее выполняются следующие действия:

1. Запустить САПР Quartus II.
2. Создать новый проект – File\ New Project Wizard. Необходимо избегать путей и наименований с русскими символами.
3. Выбрать семейство и тип микросхемы. Рекомендуется семейство ПЛИС МАХ3000А (а при наличии в схеме ЗУ семейство МАХ II) и режим автоматического выбора конкретной ИС.
4. Выбрать графический редактор – File/ New/Block Diagram.
5. Если проект строится по иерархическому принципу и содержит встроенные блоки, то выполнять далее по тексту, в противном случае сразу переходить к пункту 7. Для большинства заданий в этом курсовом проекте приемлемо рассмотрение схем в виде одного блока.

6. Рисовать схемы (работу в редакторе рассмотрим ниже) встроенных блоков, опираясь на средства графического редактора, соответствующие построенным функциональным схемам блоков. При составлении схем использовать стандартные компоненты.
7. Сохранить созданную схему блока – File/ Save
8. Создать символ– File/ “Create/Update” / Create Symbol File.
9. Рисовать схему блока верхнего уровня иерархии, опираясь на средства графического (схемного) редактора, соответствующие функциональной схеме узла. При составлении схем использовать стандартные компоненты и созданные символы фрагментов нижнего уровня.
10. Сохранить схему – File / Save.
11. Компилировать проект (Processing/Start Compilation), опираясь на данные о ПЛИС – Processing/Start Compilation.
12. Оценить затраты на реализацию проекта Processing / Compilation Report.
13. Просмотреть RTL вид проекта и его топологическую реализацию – Tools / RTL View и Tools / Technology Map View соответственно.

3.2. Работа в схемном редакторе

Помимо традиционных средств работы с файлами, иконки которых расположены сверху рабочего поля, работа в редакторе базируется на использовании рабочих инструментов, иконки вызова которых расположены слева от рабочего поля. Назначение инструментов традиционно:

- вызов символов схемы (Symbol Tools) и символов блоков (Block Symbol); рисование соединений символов проводами (Node Tool), шинами (Bus Tool), кондуитами (Conduit Tool);
- средства корректировки схем: перемещение (Selection Tool), использование резиновой нити (Use Rubber-Banding), при котором провод ведется непрерывно от выбранной начальной точки до конечной, или использование средств частичного выбора линий (Use Partial Line Selection);
- управление масштабом (Zoom Tool);
- переход к полному экрану (Full Screen);

- управление внешним или внутренним расположением окна редактора (Attach\Detach Window);
- инструменты традиционной графики.

Для графического редактора Quartus приняты следующие правила работы с шинами. Для любых действий шины и их фрагменты (включая одиночные линии) должны быть поименованы (например, address[7..0], address[3..0], address6, address[4]) . Для задания имен курсором выбирается шина или линия (выбранный элемент изменит цвет), далее набирается и вводится по Enter имя фрагмента. Имя будет закреплено за данным соединением и может перемещаться для удобства чтения. Места подключения фрагментов автоматически обозначаются точками.

3. 3. Работа в редакторе конечных автоматов

Работа в редакторе конечных автоматов выполняется в основном окне редактора при помощи инструментов, расположенных в окне инструментов. Задача редактора в интерактивном режиме разместить в основном окне символы состояний с их названиям и дуги переходов из состояния в состояние с указанием логического условия перехода. Кроме того рабочие инструменты позволяют добавлять входные и выходные сигналы автомата в окнах Input Table и Output Table соответственно. Требуемые значения выходных сигналов могут устанавливаться различными способами. Один из вариантов заключается в записи имен выходных сигналов и соответствующих значений в всплывающем окне State Properties\Actions, появляющемся после двойного щелчка на выбранном символе состояния автомата.

Запись логических выражений, соответствующих условиям перехода из состояния в состояние осуществляется по правилам языка Си (символ операции логического присвоения ==, логические операции &, |, ^). Набор логических выражений для каждого перехода (включая переход на исходное состояние) должен быть полным (включать все входные сигналы, обеспечивающие переходы).

3.4. Процесс моделирования работы узлов средствами пакета Quartus II

Разработанные узлы очень редко оказываются функционирующими в полном соответствии с техническим заданием на проект. Опыт показывает, что некоторые погрешности обнаруживаются даже после нескольких лет практического использования проекта. Основная задача проектировщика со-

стоит в уменьшении вероятности проникновения погрешностей в окончательный проект. Существенную роль в решении этой задачи играют САПР.

Программные пакеты автоматизируют процесс создания временных диаграмм, соответствующих воздействию на программную модель проекта тестовых воздействий. Различают функциональные и временные программные модели. Первый тип предполагает нулевые задержки элементов, второй — задержки, определяемые выбранной элементной базой. Функциональное моделирование обычно применяется на начальных этапах проектирования, когда важно убедиться в правильности работы проекта. Затем переходят к моделированию с учетом задержек элементов.

В рассматриваемой САПР Quartus II построение требуемой временной диаграммы выходных сигналов выполняется при последовательном выполнении двух программных пакетов. При помощи встроенного в САПР редактора временных диаграмм (Waveform Editor) вначале создается шаблон будущей диаграммы (MAX+PLUS II\ Waveform Editor). В шаблоне задается длина диаграммы, шаг диаграммы, перечень входных и выходных сигналов, планируемое поведение входных сигналов. На следующем этапе необходимо запустить программу моделировщика (Processing\Start Simulation). После успешного завершения работы симулятора можно вернуться в редактор временных диаграмм, чтобы посмотреть поведение выходных сигналов проекта.

Рекомендуемый порядок работы в редакторе временных диаграмм приведен ниже. Наиболее проблематичной является отказ системы моделирования подготавливать выходные данные для некоторых промежуточных сигналов схем (сообщения и предупреждения системы недостаточно четко определяют причину отказа). Кардинальным методом борьбы с такой проблемой является вывод таких сигналов на выходные контакты. Конечно, подобный вывод сигналов (и требуемая последующая компиляция) обычно приводит к некоторому изменению схемы и некоторых её параметров, чаще всего после определения работоспособности интересующего фрагмента схемы введенные дополнительные контакты могут быть убраны.

3.5. Работа в редакторе временных диаграмм

Временные диаграммы создаются в редакторе временных диаграмм (File/New/Verification\Debugging Files/Vector Waveform File). После этих действий появляется окно редактора. Обычно в начале работы с редактором задается общее время моделирования (Edit/End Time) и шаг временной сетки

(Edit/Grid Size). Для получения доступа к данным проекта целесообразно сразу выполнить команду File/Save при этом в выпадающем окне выбрать рекомендуемое имя файла. В окне редактора изменяется имя файла *.vwf.

Следующий шаг заключается в выборе входных и выходных сигналов проекта, предназначенных для отображения во временной диаграмме. После щелчка в поле Name в выпадающем окне Insert Node or Bus выбрать Node Finder и далее для окна Filter выбрать интересующий тип сигналов (Pin, Registers: pre-synthesis, ...). После нажатия клавиши List выбирать в списке окна Nodes Foundes интересующие сигналы и пересылать их в окно Selected Nodes. Завершаются действия выбора сигналов традиционно последовательным нажатием клавиши ОК.

Последующая работа связана с формированием требуемой последовательности входных данных (для выходных данных аналогичными действиями можно определить ожидаемые значения). Для формирования значений используется интерактивный режим работы в окне временной диаграммы с применением рабочих инструментов, расположенных слева от рабочего поля. Выделяя требуемый диапазон задания сигнала (группы сигналов), а затем и инструмент, соответствующий желаемому значения сигнала, можно строить желаемую временную диаграмму.

3.6. Оценка временных характеристик разработки.

САПР позволяет автоматизировать расчет различных временных характеристик проекта при помощи специального пакета – Processing/ Classic Timing Analyzer Tool.

Анализатор позволяет определить три типа временных характеристик: допустимую скорость изменения тактовых сигналов (Registered Performance), задержку выбранных фрагментов комбинационных схем (Custom Delays) и различные временные характеристики (tpd – время требуемое для прохождения сигнала от входного контакта через комбинационную логику до выходного, tsu – время предустановки входного сигнала для триггерных схем, tco – максимальное время требующееся для получения устойчивого сигнала на выходном контакте, th – время предустановки входных данных триггера).

4. Задания

Шифр задания состоит из **пяти** разделенных точками цифр (четыре десятичных и одна шестнадцатиричная), например 1.2.1.2. 80h.

Первая цифра указывает номер задания, вторая — вариант (для каждого задания предусмотрены несколько вариантов, в которых тип проектируемого узла сохраняется, а конкретные данные проекта варьируются), **третья цифра обозначает заданное для реализации проекта семейство ПЛИС**, четвертая определяет критерий качества, используемый для сравнения предлагаемых вариантов построения узла. Числа пятой позиции определяют начальный адрес зоны адресов, отведенной для адресации регистров устройства.

Таким образом, шифр задания имеет структуру:

Номер задания	Номер варианта	Семейство ПЛИС	Критерий качества	Начало зоны адресов
---------------	----------------	-----------------------	-------------------	---------------------

Числа третьей позиции задают критерий качества схемы в виде:

- 1 — А (минимум аппаратных затрат на реализацию узла);
- 2 — АТ (произведение аппаратных затрат на время получения результата или на период синхроимпульсов, тактирующих работу узла);

Варианты заданий

1. Многорежимный формирователь импульсных последовательностей

Разработать принципиальную электрическую схему устройства, формирующего заданные последовательности импульсов. Входные сигналы поступают от генератора ГТИ (генератор разрабатывается). Выходные последовательности цикличны. Длина цикла N периодов тактирующих импульсов, на выходе должны формироваться импульсы с указанными в задании номерами и заданной скважностью $\theta = T/t_{\text{и}}$.

Код, задающий режим работы схемы, а также сигналы ее пуска и останова поступают из управляющего устройства (процессора). Формирователь рассматривается как внешнее устройство процессорной системы, восьмиразрядные адреса его регистров расположены в адресном пространстве, содержащем 256 адресов. Начальный адрес зоны адресов указан в задании.

Варианты задания приведены в табл. 1

Таблица 1

Ва	N	Номера импульсов, проходящих на выход формирователя	Нача	θ
----	---	---	------	----------

ри- ант		Режимы						льны й ад- рес	
		1	2	3	4	5	6		
1.1	20	1,5,9,18	1,3,10, 19	4,7,12, 15	6,11,13, 16,20	2,3,10, 15	—	40h	8
1.2	20	2,5,7,18,2 0	1,3,15, 17	4,9,12, 15,18	3,13,16, 20	7,9,10, 17	1,2,7,9, 18,20	80h	4
1.3	24	1,9,15, 18	3,5,10, 13,19	4,8,12, 15,17,24	6,11,14, 16,20	2,7,10, 15,19	—	30h	6
1.4	22	3,8,11, 20	2,4,12, 21	5,10,15, 16	8,9,13, 17	1,6,11, 19	1,2,7,9,1 8,22	C2h	8

2. Формирователь импульсной последовательности с линейно-изменяющимися интервалами между импульсами

Разработать узел, формирующий заданную последовательность импульсов. Входные сигналы частоты $f = 1/T$ поступают от генератора ГТИ (генератор разрабатывается). Выходная последовательность периодична с периодом T_p . Узел рассматривается как внешнее устройство процессорной системы. Сигналы пуска и останова поступают из управляющего устройства (процессора), количество периодов работы от пуска до останова фиксируется счетчиком (их максимальное число 100).

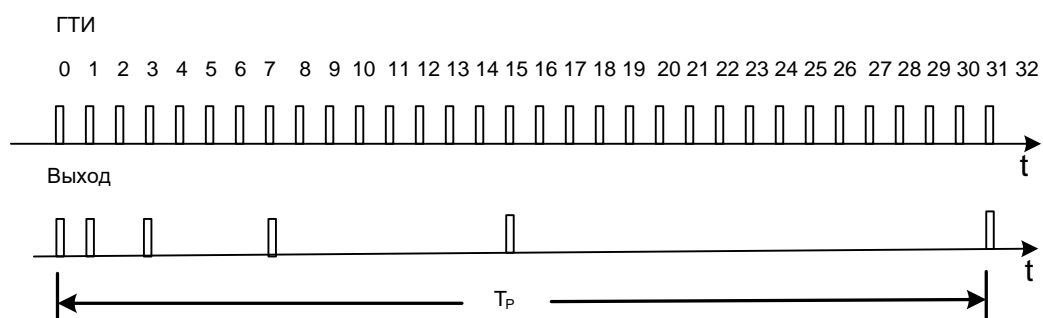


Рис. 7

Пример временных диаграмм сигналов для варианта с линейным ростом интервалов между импульсами показан на рис. 7. Варианты заданий приведены в табл.2.

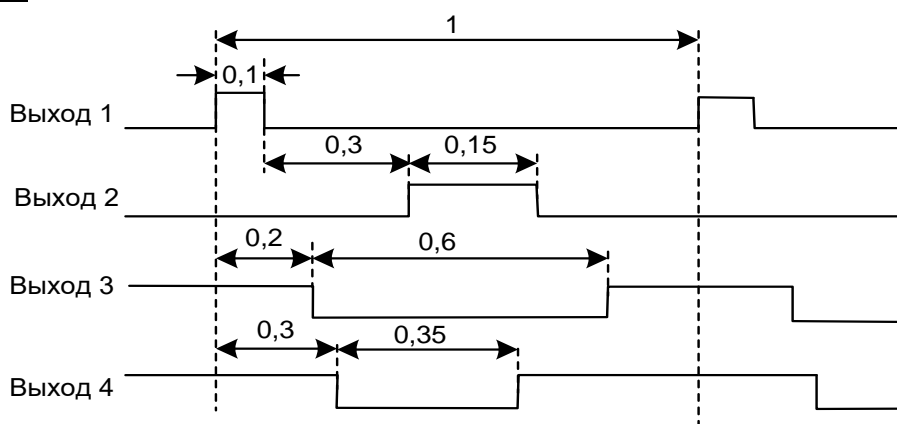
Таблица 2

Вариант	Т _р	Номера импульсов, проходящих на выход	Начальный адрес	θ
2.1	30	0,1,3,7,15,23,27,29,30	30h	4
2.2	30	0,1,3,6,10,15,20,24,27,29,30	40h	8
2.3	31	0,1,3,7,15,31	60h	4
2.4	21	0,6,11,15,18,20,21,	80h	8
2.5	25	0,1,3,7,12,18,25	30h	4

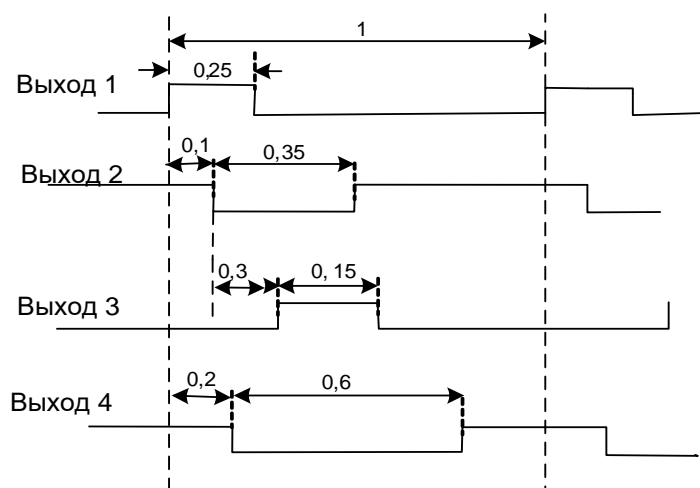
3. Четырехканальный генератор сигналов с заданными временными диаграммами

Разработать принципиальную электрическую схему генератора импульсов, формирующего на выходах 1, 2, 3, 4 сигналы, изображенные на временных диаграммах (рис. 8). Временные интервалы на диаграммах заданы в микросекундах. Возможные варианты реализации: сочетание двоичного счетчика с ППЗУ, схема на основе распределителя тактов и сборок по ИЛИ; схема с получением искомым импульсов из сигналов счетчика Джонсона, схема на основе кольцевого соединения одновибраторов и др. Критерий выбора наилучшего варианта — минимум аппаратных затрат.

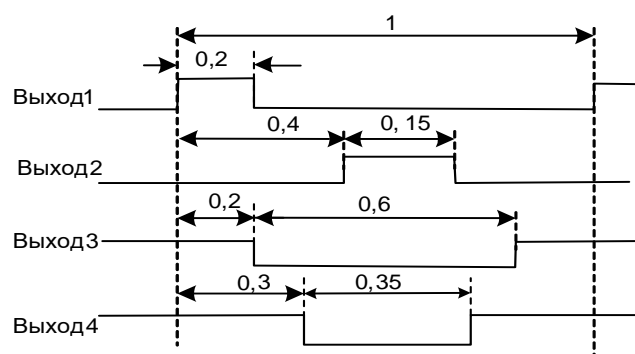
Вариант 3.1



Вариант 3.2



Вариант 3.3



Вариант 3.4

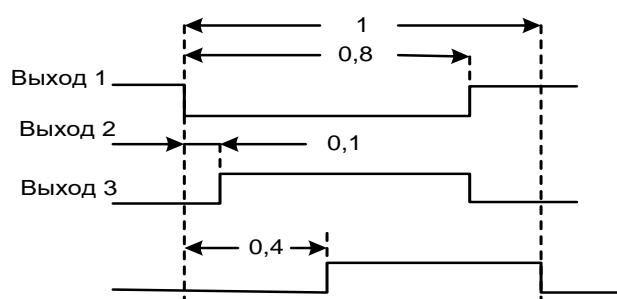


Рис. 8

4. Реверсивный распределитель тактовых сигналов с переменным числом каналов

Разработать распределитель тактов, формирующий на выходах заданные последовательности импульсов. Входные сигналы частоты $f = 1/T$ распределитель получает от генератора ГТИ (генератор разрабатывается).

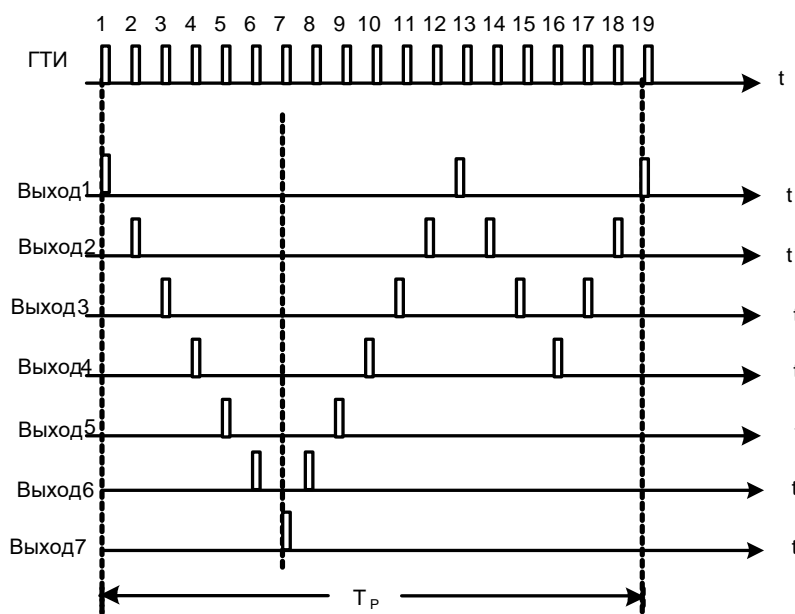


Рис. 9

Выходная последовательность периодична с периодом T_P . Скважность импульсов равна четырем. Сигналы пуска и останова распределителя поступают из управляющего устройства (процессора), количество периодов работы распределителя от пуска до останова фиксируется (их максимальное число 64). Распределитель рассматривается как ВУ процессорной системы, его адреса расположены в третьей четверти адресного пространства емкостью 256 адресов.

Таблица 3

Вариант	T_P	Номера импульсов, проходящих на выходы распределителя									
		1	2	3	4	5	6	7	8	9	10
4.1	19	1, 13	2, 12, 14, 18	3, 11, 15, 17	4, 10, 16	5, 9	6, 8	7	—	—	—
4.2	19	1, 9, 13, 19	2, 8, 10, 12, 14, 17, 18	3, 7, 11, 15, 17	4, 6, 16	5	—	—	—	—	—
4.3	24	1, 19	2, 18, 20, 24	3, 17, 21, 23	4, 16, 22	5, 15	6, 14	7, 13	8, 12	9, 11	10
4.4	24	1, 19	2, 18, 17,	3, 17,	4, 16,	5, 15	6, 14	7, 13	8, 12	9, 11	10

			20, 24	21, 23	22						
--	--	--	-----------	-----------	----	--	--	--	--	--	--

Пример временных диаграмм сигналов показан на рис.9. Варианты заданий приведены в табл.3.

Возможные варианты реализации: на основе двоичного счетчика с дешифратором и сборками по ИЛИ, на основе реверсивного регистра, на основе сочетания двоичного счетчика и программируемой памяти, сочетания счетчика и ПМЛ и др.

5. Программируемый преобразователь кода в ШИМ-сигнал

Разработать принципиальную электрическую схему преобразования параллельного входного кода N в широтно-модулированный импульсный сигнал (рис. 10) с длительностью $t_{и}$ и периодом $T_{ц}$, где $t_{и} = NT_{ц}/N_{MAX}$, $T_{ц} = (N_{MAX}+1)kT$, T – период тактового генератора (генератор разрабатывается), а коэффициент $k = 1, 4, 8$ задается управляющим кодом.

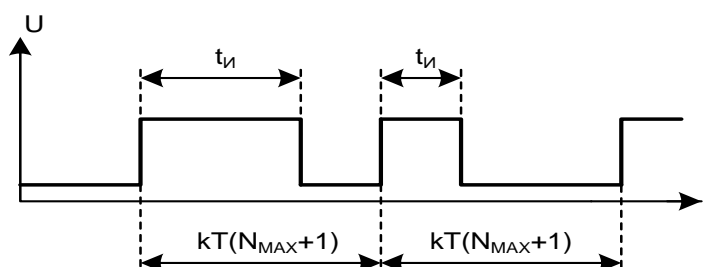


Рис.10

Входной код, управляющий код и сигналы пуска/останова преобразователя поступают из управляющего устройства (процессора). Преобразователь рассматривается как ВУ процессорной системы, имеющее адрес в заданной части адресного пространства АП емкостью 256 адресов. Предложить не менее двух вариантов решения. Варианты задания приведены в табл.4.

Таблица 4

Вариант	Разрядность входного кода	Начало области адресов
5.1	8	45h

5.2	10	75h
5.3	12	85h
5.4	16	C5h
5.5	20	35h

6. Счетчик с программированием приращений содержимого

Счетчик имеет 10 разрядов и является двоично-кодированным. Содержимое счетчика изменяется на ΔN с приходом каждого входного импульса. Возможны два режима работы счетчика: с ΔN_1 и ΔN_2 . Код режима, сигналы сброса, пуска и останова счетчика поступают от управляющего устройства (процессора). Код, полученный в счетчике к моменту останова, фиксируется. Счетчик рассматривается как внешнее устройство процессорной системы, его адреса расположены в заданной области адресного пространства системы, имеющего емкость 256 адресов.

Возможны следующие варианты реализации: формальный синтез (с использованием триггеров и логических элементов заданного типа), схема с применением комбинационного сумматора и регистра в младших разрядах, вариант с изменением содержимого счетчика на $0,5(\Delta N_1 + \Delta N_2)$ и последующим прибавлением или вычитанием единицы, схема с двоичным счетчиком и ППЗУ и др.

Варианты задания приведены в табл.5.

Таблица 5

Вариант	ΔN_1	ΔN_2	Тип триггера	Тип логического элемента	Область АП
6.1	+3	+5	JK	И-НЕ	40h
6.2	-3	-5	D	И-НЕ	80h
6.3	+7	+9	JK	ИЛИ-НЕ	C0h
6.4	-7	-9	D	ИЛИ-НЕ	36h
6.5	+15	+17	JK	И-НЕ	85h
6.5	-15	-17	D	И-НЕ	20h

7. Преобразователь параллельного кода в последовательный для асинхронного протокола передач

Разработать узел преобразующий параллельный код в последовательный с добавлением к нему старт-бита, стоп-бита и бита паритета.

Варианты задания приведены в табл.6.

Таблица 6

Вариант	Разрядность входного кода, бит	Скорости передачи, Кбит/с	Область АП
7.1	8	2,4; 4,8; 9,6	40h
7.2	16	4,8; 9,6; 19,2	80h
7.3	12	1,2; 2,4; 4,8	85h
7.4	8	9,6; 19,2; 38,4	60h

Предусмотреть возможность передачи последовательного кода со скоростями, указанными в вариантах задания. Входной параллельный код, код выбора скорости передачи и сигнал пуска передаются из управляющего устройства (процессора). Критерий выбора варианта реализации схемы — минимум аппаратных затрат.

8. Блок репрограммируемой памяти с контролем по Хеммингу

Разработать блок репрограммируемой памяти, в котором используется модифицированный код Хемминга для контроля и коррекции ошибок в рабочем режиме (режимы стирания/записи не рассматриваются). Блок памяти должен хранить s слов с разрядностью n (имеется в виду число информационных разрядов) и строится. Разработать принципиальную электрическую схему блока репрограмми на микросхемах с организацией $k \times p$. Адреса памяти занимают указанную в таблице область адресного пространства (АП) системы (емкость АП равна 64К). Кодировующее устройство блока преобразует n -разрядные входные слова в m -разрядные модифицированные коды Хемминга, которые хранятся в блоке памяти. При считывании слов декодирующее устройство блока формирует n -разрядные выходные слова с коррекцией одиночной ошибки и обнаружением двойной. Работа блока памяти должна быть согласована с временными диаграммами циклов чтения и записи процессора. Длительность цикла чтения процессора t_{cy} . Признак двойной ошибки фиксируется. Варианты задания приведены в табл. 7.

Таблица 7

Вариант	s	n	k	p	Область АП	t _{cy} , нс
8.1	16К	16	8К	8	Вторая четверть	600
8.2	32К	8	8К	4	Первая половина	400
8.3	8К	12	4К	4	Вторая половина	300
8.4	24К	8	8К	4	Третья четверть	200

Приложение

Пример выполнения некоторых разделов задания

Пусть требуется разработать формирователь импульсов с тремя выходами, на которых воспроизводятся функции времени $Z1$, $Z2$ и $Z3$, показанные на рис. П.1. Режим работы формирователя старт-стопный (после каждого сигнала пуска формирователь генерирует циклы работы вплоть до сигнала останова).

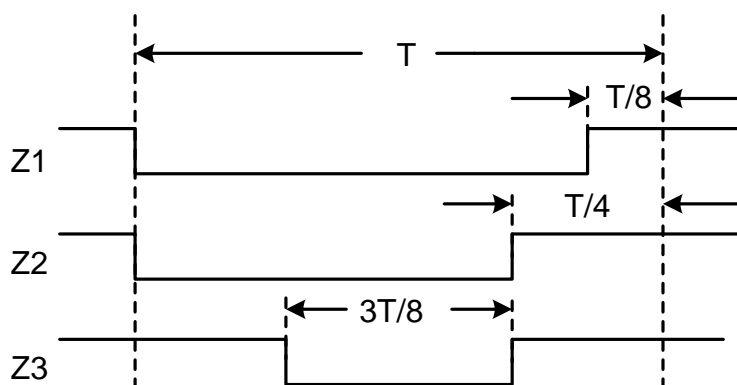
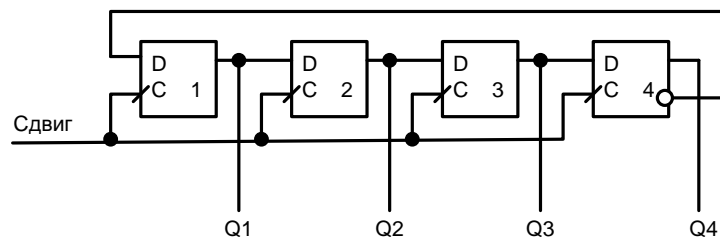


Рис. П.1

Поставленная задача может быть решена многими методами. Не будем их обсуждать, остановимся лишь на одном перспективном направлении. В этом направлении для формирования требуемых импульсов нужно фиксировать интервалы, составляющие наибольший общий делитель искомых интервалов (в нашем случае это интервал $T/8$). Из этих “квантов” можно собирать нужные выходные сигналы, объединяя их по ИЛИ. Временные отметки для границ интервалов можно получать разными способами.

Например, можно для этого использовать цепочку одновибраторов, длительности выходных сигналов которых дают нужные точки во времени. Однако это решение не обеспечит точного формирования временных интервалов вследствие ограниченной точности и стабильности длительностей выходных импульсов одновибраторов. Можно сгенерировать линейно-растущий код (генератор плюс счетчик) и с помощью трех компараторов получить “засечки” трех моментов времени. Однако такое решение по видимому окажется слишком сложным. Можно разбить период T на восемь частей (генератор плюс счетчик плюс дешифратор) и собирать импульсы из отдельных “квантов” схемой ИЛИ. Такое решение будет более простым, но так как выходные импульсы будут состояться из нескольких коротких, на

стыках последних могут возникать опасные помехи (просечки). Если вместо коротких импульсов для сборки выходных сигналов использовать более длительные, то наряду с простотой схемного решения получим исключение просечек. Более длительные импульсы можно получить на выходах счетчика Джонсона (рис. П.2, а). Такая идея представляется удачной. Четырехразрядный счетчик Джонсона имеет восемь состояний, сдвиги между импульсами на соседних выходах составляют $T/2n$, где n — разрядность счетчика Джонсона. При $n = 4$ сдвиги составят $1/8$ периода, что и позволит составить нужные выходные функции $Z1, Z2$ и $Z3$.



(а)

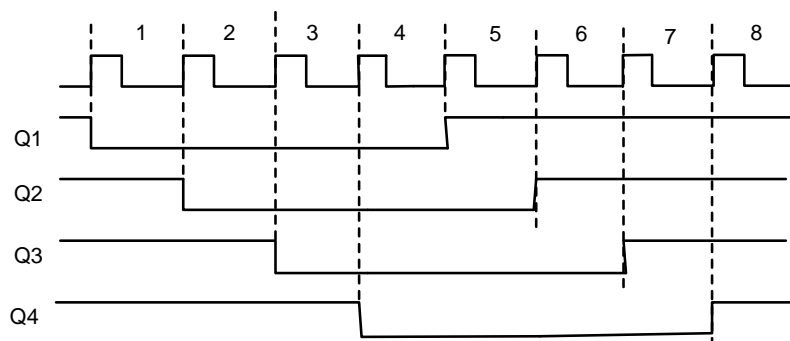


Рис. П...2

(б)

Естественно, что для компоновки искомых функций, нужно выбирать диаграммы, фронты которых совпадают с фронтами импульсов $Z1, Z2$ и $Z3$. Нетрудно видеть, что $Z1 = Q1Q3$, $Z2 = Q1Q2$ и $Z3 = Q2 \vee Q3$.

Блок-схема реализации функций и управления этим процессом от команд программы приведена на рис. П.3. Формирователю присвоен адрес 129 из адресного пространства внешних устройств емкостью 256 адресов. Появление этого адреса открывает путь для прохождения в соответствующие точ-

ки схемы импульсов \overline{IOW} и \overline{IOR} , так что приведение схемы в исходное состояние (сброс) и ее пуск осуществляются командами OUT port и IN port соответственно. Таким образом, для управления формирователем, рассматриваемым как внешнее устройство микропроцессорной системы, принята классическая схема — адресный дешифратор плюс логика передачи управляющих сигналов в требуемые точки схемы.

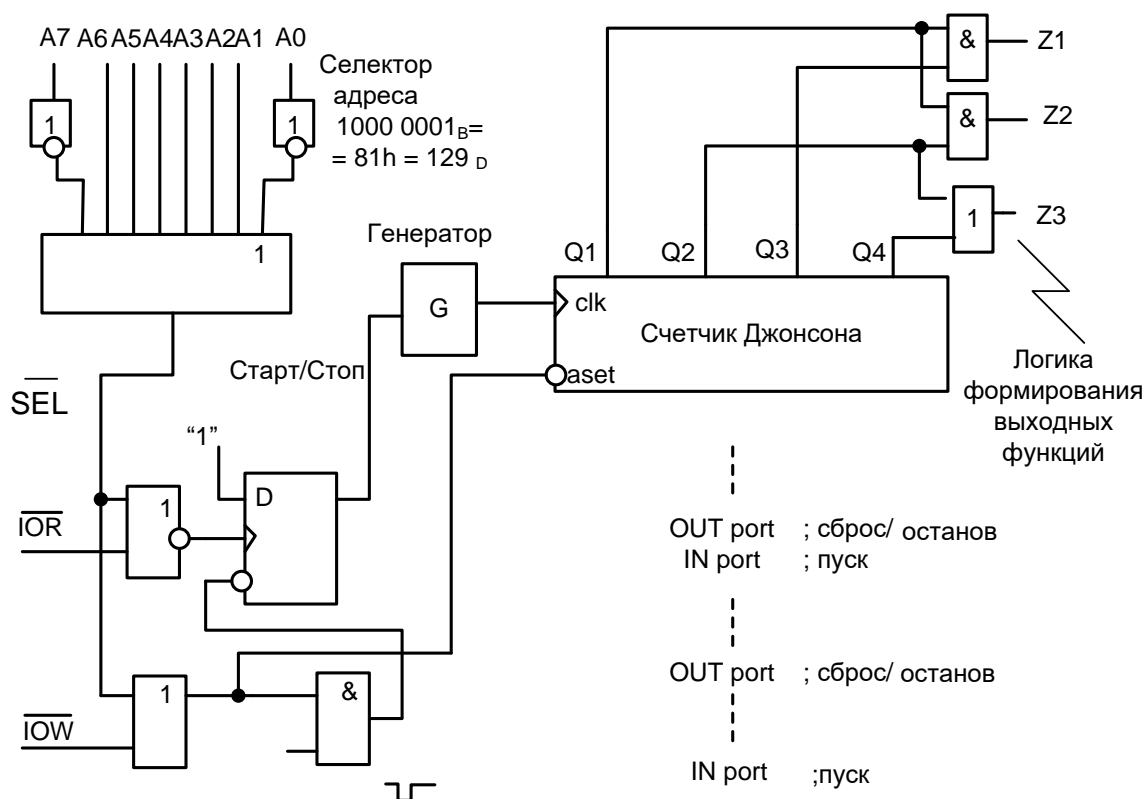


Рис. П.3

Принципиальная электрическая схема формирователя, построенная средствами графического ввода пакета Quartus II, приведена на рис. П.4. Использована библиотека примитивов. Генератор выполнен по кольцевой схеме [1] с цепью задержки из восьми инверторов в обратной связи элемента И-НЕ. При нулевом уровне входного управляющего напряжения выходное напряжение генератора имеет высокий уровень. При переходе управляющего напряжения на высокий уровень первый же нулевой импульс генератора имеет стандартную длительность.

Компиляция проекта, реализованного в микросхеме EPM3032ALC44-4 семейства MAX 3000A, дала следующие результаты: число затраченных макроячеек 9 из общего числа 32 (28%), число использованных контактов 17 из общего числа 34 (50%).

В результате моделирования работы формирователя с учетом временных задержек, свойственных примененной микросхеме семейства MAX 3000A, получены временные диаграммы, показанные на рис. П.5. Как видно из диаграммы требуемое функционирование формирователя достигнуто.

Пока адрес отличается от 10000001 формирователь не реагирует на старт-стопные воздействия. При появлении адреса 10000001 устройство запускается командой \overline{IOR} и останавливается командой \overline{IOW} . На участках работы формируются импульсы Z1, Z2 и Z3 заданной формы. Частота работы генератора около 50 МГц.

Временная диаграмма

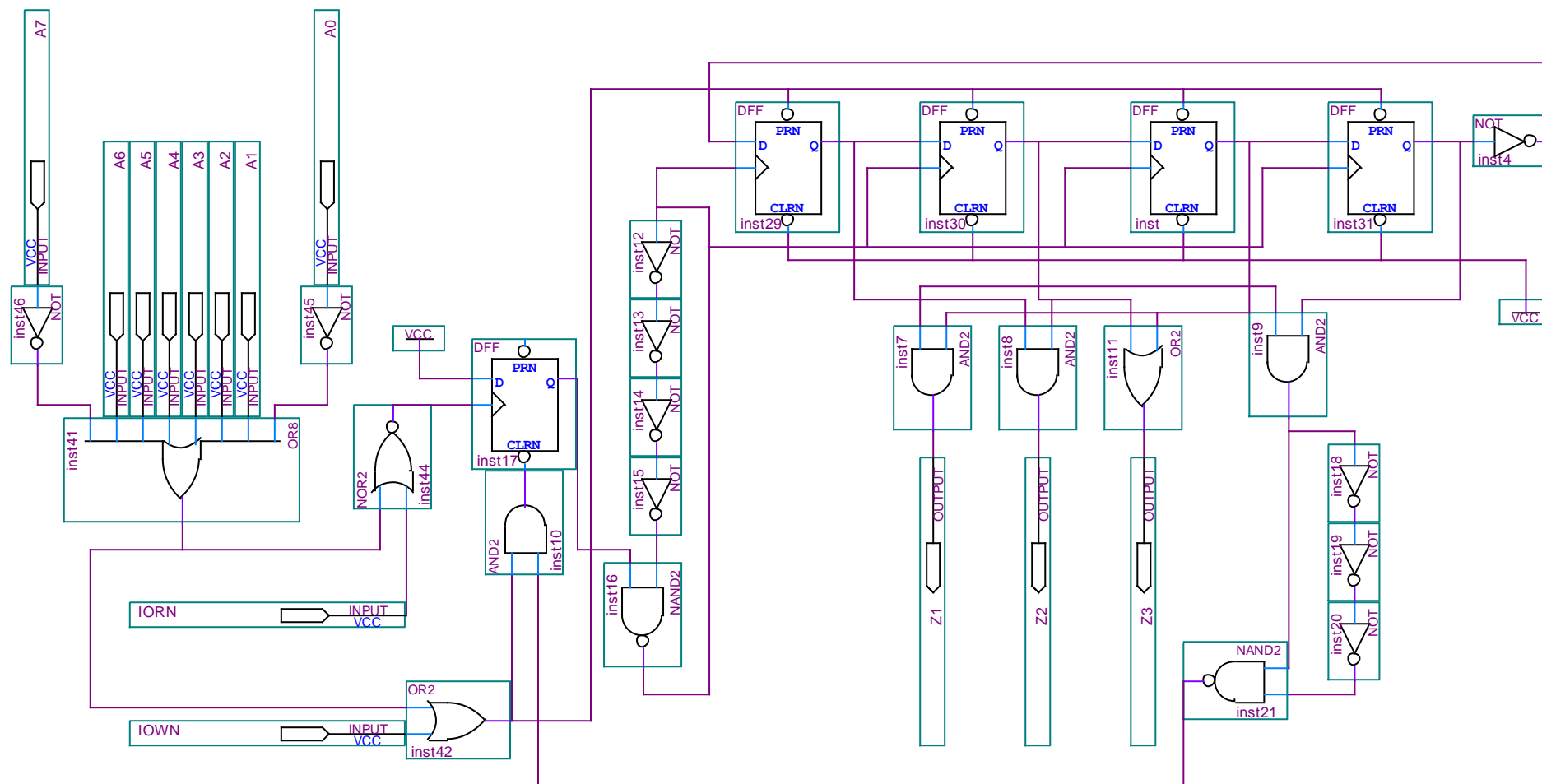


Рис. П...4

Список литературы

1. Угрюмов Е.П. Цифровая схемотехника: учеб. пособие для вузов.— БХВ-Петербург, 1-е изд. 2000 г., 2-е изд. 2006 г., 3-е изд. 2010 г.— 797 с.
2. Комолов Д.А., Мьяльк Р.А., Зобенко А.А., Филиппов А.С. Системы автоматизированного проектирования фирмы Altera MAX+plus II и Quartus II. Краткое описание и самоучитель.— М.: ИП РадиоСофт, 2002 — 352 с.
3. Артамонов А.Б., Смирнов А.М. Интерфейсные БИС микропроцессорных комплектов: учеб. пособие./ЛЭТИ.— Л., 1990.
4. www.altera.com/literature/catalogs/lpm.pdf (Справочник по элементам LPM).
5. Логические ИС КР1533, КР1554: Справочник: В 2-х частях / И. И. Петровский, А. В. Прибыльский, А. А. Троян, В. С. Чувелев. — М.: БИНОМ, 1993. — 496 с.

Оглавление

1. Состав и порядок выполнения курсового проекта	3
2. Рекомендации по выполнению основных этапов проектирования узла	5
2.1. Задание на проектирование узла.	5
2.2. Сравнительный анализ возможных вариантов реализации узла	5
2.3. О критериях качества разрабатываемых вариантов	6
2.4. Применение программируемых ЗУ	7
2.5 Сопряжение проектируемого узла с устройством управления (интерфейсом процессорной системы	9
2.6. Этап технической реализации проектируемого устройства	10
2.7 Разработка принципиальной электрической схемы интерфейсной платы МП системы	11
2.8. Выбор и разработка генератора тактовых импульсов	13
2.5. О некоторых правилах оформления схем	15
3. Работа в САПР Quartus II	
3.1. Последовательность работы в САПР Quartus II	16
3.2. Работа в схемном редакторе	17
3.3. Работа в редакторе конечных автоматов	17
3.4. Процесс моделирования работы узлов средствами пакета Quartus II	18
3.5 Работа в редакторе временных диаграмм	19
3.6. Оценка временных характеристик разработки	20
3. Задания	20