

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования

**«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ» (ТУСУР)**

Кафедра радиотехнических систем (РТС)

А.А. Савин

**РАЗРАБОТКА И ИССЛЕДОВАНИЕ ЦИФРОВЫХ УСТРОЙСТВ
НА БАЗЕ ПРОГРАММИРУЕМЫХ ЛОГИЧЕСКИХ ИНТЕГРАЛЬНЫХ
СХЕМ ПРИ ПОМОЩИ САПР «MAX+plus II»**

Учебно-методическое пособие для выполнения лабораторных работ

Томск – 2012

Савин А.А. Разработка и исследование цифровых устройств на базе программируемых логических интегральных схем при помощи САПР «MAX+plus II». Учебно-методическое пособие для выполнения лабораторных работ. – Томск: ТУСУР, 2012.

Учебно-методическое пособие предназначено для выполнения лабораторных работ по дисциплине «Интегральные схемы и их применение». Лабораторные работы ориентированы на проведение синтеза и исследования цифровых устройств средствами системы автоматизированного проектирования «MAX+plusII». Пособие будет полезным при дипломном проектировании.

Рекомендовано кафедрой радиотехнических систем ТУСУР для студентов специальностей 210300.62 «Радиотехника», 210304.65 «Радиоэлектронные системы», 210403.65 «Защищенные системы связи» и 090106.65 «Информационная безопасность телекоммуникационных систем».

Содержание

Введение.....	4
1. Краткое описание САПР «МАХ+plus II».....	5
1.1 Приложения для ввода проектов (редакторы проектов).....	5
1.2 Приложения «МАХ+plus II Compiler».....	6
1.3 Приложения для верификации проектов.....	7
1.4 Приложение «МАХ+plus II Programmer».....	7
1.5 Сервисные приложения.....	7
1.6 Рабочие каталоги системы.....	8
2. Работа в редакторе «Graphic Editor».....	9
2.1 Ввод графических элементов.....	12
2.2 Проведение связей.....	14
2.3 Шины.....	15
3. Компиляция.....	16
4. Назначение типа ПЛИС и выводов ПЛИС.....	17
5. Моделирование и анализ работы схемы.....	19
6. Задания на выполнение лабораторных работ.....	23
6.1 Лабораторная работа № 1.....	23
6.2 Лабораторная работа № 2.....	23
6.3 Лабораторная работа № 3.....	24
6.4 Лабораторная работа № 4.....	27
7. Требования к оформлению отчета.....	28
Рекомендуемая литература.....	29

Введение

Целью цикла лабораторных работ является освоение методов синтеза и анализа цифровых устройств, получение навыков работы в системе автоматизированного проектирования (САПР) «MAX+plus II». Данный программный продукт находится в свободном доступе и его можно скачать с официального сайта производителя – фирмы Altera [1].

Цифровые устройства различной сложности являются неотъемлемой частью практически любой радиотехнической системы. Для закрепления теоретического материала, полученного в рамках курса «Интегральные схемы и их применение», предлагается выполнить ряд заданий [2 – 6]. Специфика заданий заключается в разработке некоторого цифрового логического устройства, симуляции его работы и тестировании путем подачи на входы всевозможных комбинаций сигналов. Трудоемкость работ возрастает по мере их выполнения.

1. Краткое описание САПР «MAX+plusII»

САПР «MAX+plus II» представляет собой интегрированную среду для разработки цифровых устройств на базе программируемых логических интегральных схем (ПЛИС) фирмы Altera и обеспечивает выполнение всех этапов, необходимых для выпуска готовых изделий:

- создание проектов устройств;
- синтез структур и трассировку внутренних связей ПЛИС;
- подготовку данных для программирования или конфигурирования ПЛИС (компиляцию);
- верификацию проектов (функциональное моделирование и временной анализ);
- программирование или конфигурирование ПЛИС.

Наиболее полное фирменное описание системы «MAX+plus II» можно найти на сайте фирмы Altera www.altera.com [1].

1.1 Приложения для ввода проектов (редакторы проектов)

Для ввода проектов можно использовать:

- «Graphic Editor» – графический редактор, предназначенный для ввода проекта в виде схемы соединений символов элементов, извлекаемых из стандартных библиотек пакета либо из библиотеки пользователя.
- «Text Editor» – текстовый редактор, предназначенный для создания и редактирования текстовых файлов, содержащих описание логики проекта на языке описания устройств AHDL (Altera Hardware Description Language) или на близких к нему языках типа VHDL, «Verilog».
- «Symbol Editor» – символьный редактор, позволяющий редактировать существующие символы и создавать новые. Любой откомпилированный проект может быть свёрнут в символ, помещён в библиотеку символов и использован как элемент в любом другом проекте.

– «Floorplan Editor» – редактор связей (поуровневый планировщик), который на плане расположения основных логических элементов позволяет вручную распределять выводы ПЛИС (закреплять выводы за конкретными входными и выходными сигналами) и перераспределять некоторые внутренние ресурсы ПЛИС.

1.2 Приложения «MAX+plus II Compiler»

Это приложения, входящие в пакет компилятора и предназначенные для синтеза структуры, трассировки связей, проверки корректности проекта и локализации ошибок, формирования файлов программирования или конфигурирования ПЛИС:

– «Netlist Extractor» – приложение, обеспечивающее извлечение списка соединений из исходного файла представления проекта, созданного при вводе проекта.

– «Database Builder» – приложение, предназначенное для построения базы данных проекта.

– «Logic Synthesizer» – приложение, обеспечивающее проверку корректности проекта по формальным правилам и синтез оптимальной структуры проекта.

– «Partitioner» – приложение, обеспечивающее разбиение проекта на части в тех случаях, когда ресурсов одного кристалла (микросхемы) недостаточно для реализации проекта.

– «Fitter» – трассировщик внутренних связей, обеспечивающий реализацию синтезированной структуры.

– «SNF Extractor» – приложение, обеспечивающее извлечение параметров проекта, необходимых для функционального моделирования и временного анализа.

1.3 Приложения для верификации проектов

Для верификации проектов используют:

- «Simulator» – приложение, которое совместно с редактором временных диаграмм предназначено для функционального моделирования проекта с целью проверки правильности логики его функционирования.
- «Timing Analyzer» – приложение, обеспечивающее расчет временных задержек от каждого входа до каждого логически связанного с ним выхода.

1.4 Приложение «MAX+plus II Programmer»

Для программирования или конфигурирования ПЛИС используется приложение «MAX+plus II Programmer». Программирование и перепрограммирование микросхем, имеющих встроенную систему программирования (ISP), может осуществляться непосредственно в составе конечного изделия через специальный кабель, подключаемый либо к LPT-порту (Byte Blaster), либо к COM-порту (Bit Blaster) компьютера и технологического 10-контактного соединителя интерфейса JTAG, устанавливаемого на плате изделия.

1.5 Сервисные приложения

В состав САПР, кроме того, входят три сервисных приложения:

- «Design Doctor» – приложение, предназначенное для проверки корректности проекта с использованием эмпирических правил.
- «Message Processor» – процессор сообщений, обеспечивающий обработку, вывод на отображение и локализацию (указание места в проекте, к которому оно относится) сообщений трёх типов: сообщений об ошибках – «Error», предупреждений – «Warning» и информационных сообщений – «Info». Причину вывода того или иного сообщения можно выяснить через опцию

«Help on Message» процессора сообщений. При наличии сообщений об ошибках компиляция проекта невозможна до их полного устранения. При наличии предупреждений компиляция успешно завершается, однако наличие предупреждения свидетельствует об обнаружении проблемы, которая может привести к неверной работе устройства. Поэтому все предупреждения должны быть тщательно проанализированы, до выяснения причин их появления и последующего устранения этих причин (или игнорирования предупреждения, что иногда возможно). Информационные сообщения нужно только принимать к сведению.

– «Hierarchy Display» – приложение, обеспечивающее обзор иерархической структуры проекта, который может состоять из множества составленных в различных редакторах и свёрнутых в символы проектов более низких уровней, причём число уровней не ограничивается.

1.6 Рабочие каталоги системы

Во время инсталляции пакета создаются два каталога: каталог \MAXPLUS2, который содержит все приложения и библиотеки пакета, и каталог \MAX2WORK, который содержит подкаталог \CHIPTRIP со всеми файлами учебного проекта, рассмотренного в руководстве («MAX+plus II Tutorial»), и ряд подкаталогов, используемых электронным справочником («MAX+plus II Help»). В этом же каталоге \MAX2WORK следует размещать и рабочие каталоги создаваемых проектов устройств. Необходимость создания отдельных каталогов для каждого разрабатываемого проекта обусловлена тем, что в процессе разработки проекта системой MAX+plus II создаётся и поддерживается множество файлов, относящихся к текущему проекту. Прежде всего это файл проекта («Project File»), название которого определяет название проекта в целом. Этот файл содержит основную логику и иерархию проекта, обрабатываемую компилятором. Кроме того, создаётся ряд вспомогательных файлов, связанных с проектом, но не являющихся частью логики проекта.

Большая часть вспомогательных файлов создаётся и автоматически помещается в каталог проекта в процессе ввода и компиляции проекта. Это прежде всего файлы назначений и конфигурации (.ACF), файлы отчётов (.RPT), файлы данных для функционального моделирования и временного анализа (.SNF), файлы данных для программирования (.POF) и ряд других. Названия этих файлов всегда совпадают с названием проекта. Некоторые вспомогательные файлы создаются пользователем: например, для выполнения функционального моделирования создаётся файл (.SCF), содержащий описание начальных и текущих состояний входных сигналов (входов) и перечень выходов, для которых должны быть определены выходные сигналы. **Поэтому перед началом работы над новым проектом следует создать рабочий каталог проекта**, при этом имя каталога можно выбирать произвольно (т.е. имя каталога может не совпадать с именем файла проекта).

2. Работа в редакторе «Graphic Editor»

После запуска пакета (файл max2win.exe) открывается главное окно (рис. 1). В самой верхней строке содержится имя последнего проекта, с которым велась работа. Две следующие строки являются обычными для windows-приложений: строка основного меню и панель инструментов, в левой части которой расположены стандартные функции Windows («New», «Open», «Save», «Print», «Cut», «Copy», «Paste», «Undo»), а в правой части – специфические инструменты пакета, с помощью которых осуществляется запуск его основных приложений. Назначение любого инструмента можно узнать, совместив с ним указатель мыши, при этом в строке подсказки (самая нижняя строка окна) появляется текст с пояснением. Также для запуска приложений можно использовать также пункты меню «MAX+plus II».

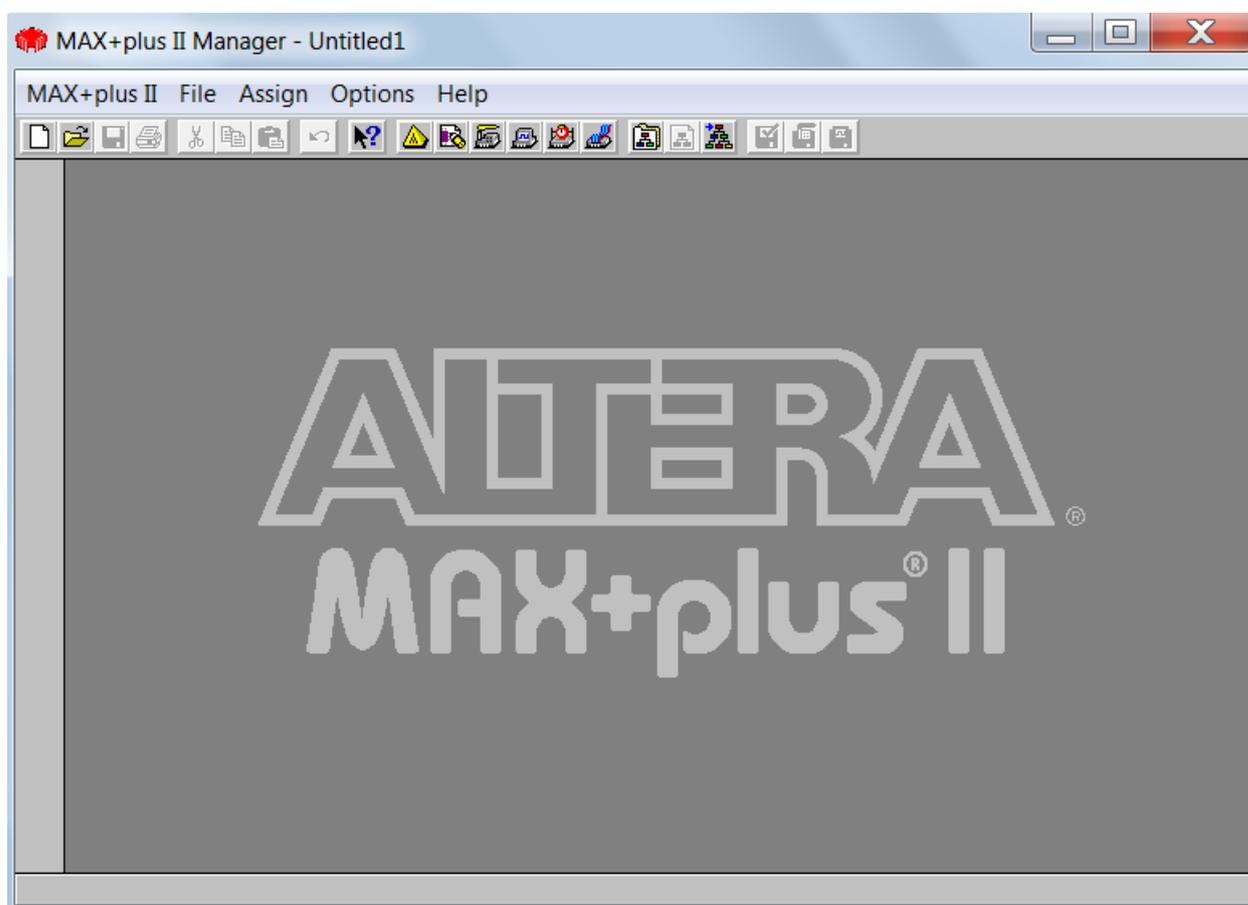


Рис.1 Главное окно САПР MaxPlus II

Порядок использования основных приложений системы «MAX+plus II» рассмотрим на примере разработки простого инвертора. Для начала в директории \Max2Work необходимо создать рабочую папку для хранения файлов будущего проекта. Далее создадим проект схемы, для этого нужно выполнить команду «New» из меню «File», или нажать на иконку . В открывшемся диалоговом окне «New» выбираем пункт «Graphic Editor file» и нажимаем кнопку «ОК», при этом автоматически открывается окно редактора «Graphic Editor» предназначенного для графического ввода схем (рис. 2).

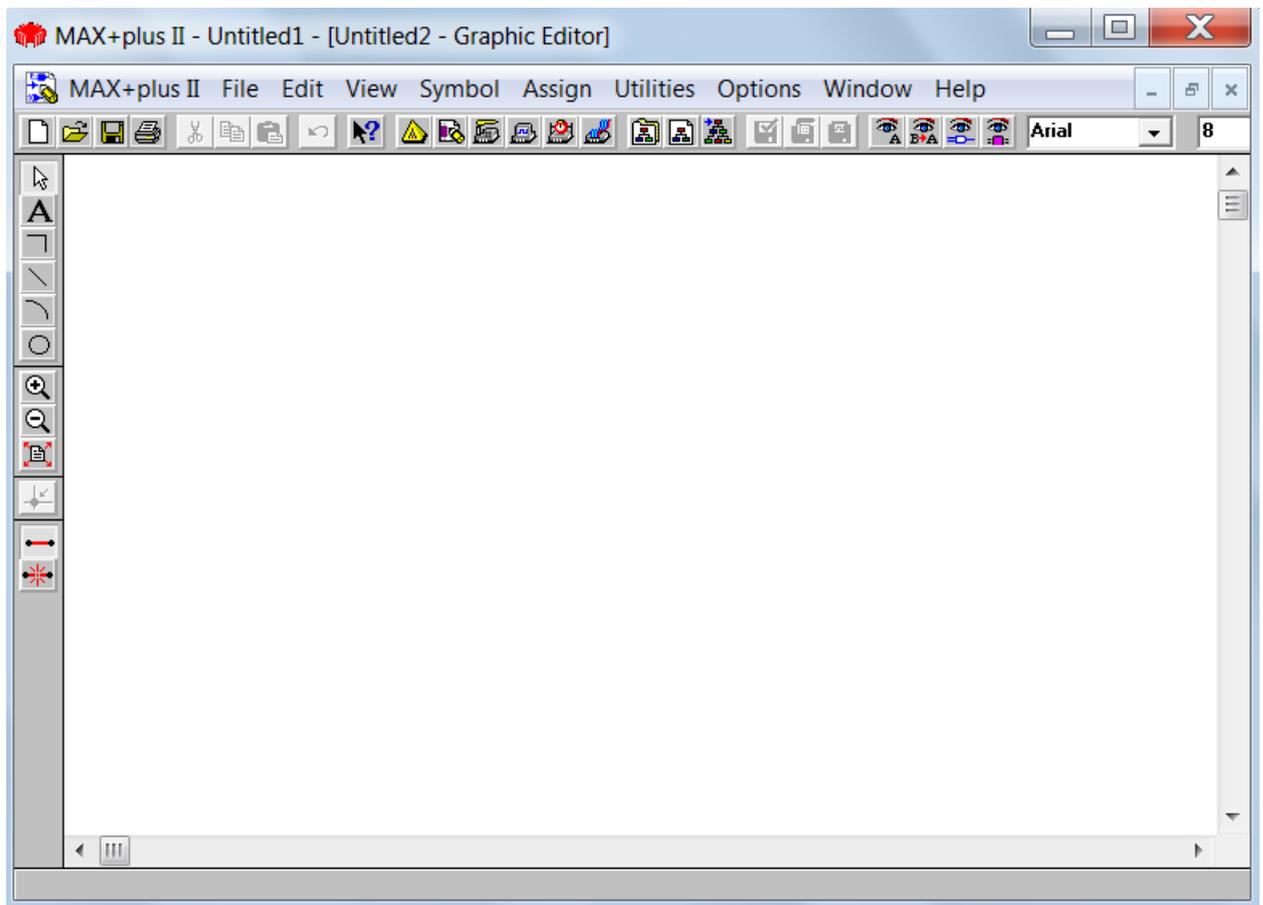


Рис. 2 Окно приложения Graphic Editor

Новый файл проекта необходимо сохранить в ранее созданной рабочей папке, используя меню «File → Save». Имя файла проекта следует обязательно привязать к имени проекта. Это делается при выборе пункта «Set Project to Current File» в меню «File → Project». Прежде чем приступить к созданию схемы необходимо рассмотреть возможности графического редактора. Слева в окне располагается линейка с инструментами. Часть инструментов редактора предназначена для создания поясняющих надписей, рисунков и таблиц, которые не являются частью проекта и не обрабатываются компилятором.

 – При активизации этого инструмента становится возможным выделение, передвижение, копирование или удаление объектов. Горячая клавиша вызова данного инструмента – «ESC».

 – Инструмент позволяющий создавать новые надписи или редактировать существующие. Шрифт и его размер задаются при помощи меню «Options → Fonts» и «Options → Text Size». Горячая клавиша – «F2».

 – Рисование отрезков горизонтально/вертикально и под любым углом. Кроме того, этот инструмент используется для задания связей между графическими элементами. Горячие клавиши «F3, F4».

 – Рисование дуг и окружностей. Горячие клавиши «F5, F6».

  – инструменты масштабирования, соответственно увеличение и уменьшение. Горячие клавиши «Ctrl+Space» и «Ctrl+Shift+Space».

 – Этот инструмент также относится к группе масштабирования и позволяет уменьшить масштаб до значения, при котором поле ввода схемы уменьшается до размеров окна. Горячая клавиша «Ctrl+W».

 – Этот инструмент необходим для создания или удаления точек соединения связей графических элементов. Активизируется при выделении одной из линий связи.

  – RubberBanding ON/OFF. Данная функция определяет положение линий связи при передвижении графический компонентов. Если данная функция включена, то при перетаскивании компонента линии связи не обрываются, а передвигаются вместе с компонентом. В противном случае при смене местоположения графического компонента связи его с другими элементами обрываются.

Часто в графических редакторах необходимым является наличие сетки. В приложении «Graphic Editor» сетка включается/отключается нажатием комбинации клавиш «Ctrl+Shift+G» или при помощи меню «Options → Show Guidelines». Настройка шага сетки осуществляется пунктом меню «Options → Guideline Spacing».

2.1 Ввод графических элементов

Для размещения элементов в окне редактора используется диалоговое окно «Enter Symbol», (рис. 3) которое можно открыть через меню «Symbol → Enter Symbol». Однако удобнее вызывать это окно двойным щелчком левой клавиши мыши по свободному пространству открытого окна редактора. После

чего выбранный элемент будет размещён именно в этом месте (позиционирование элемента производится по верхнему левому углу условного обозначения элемента).

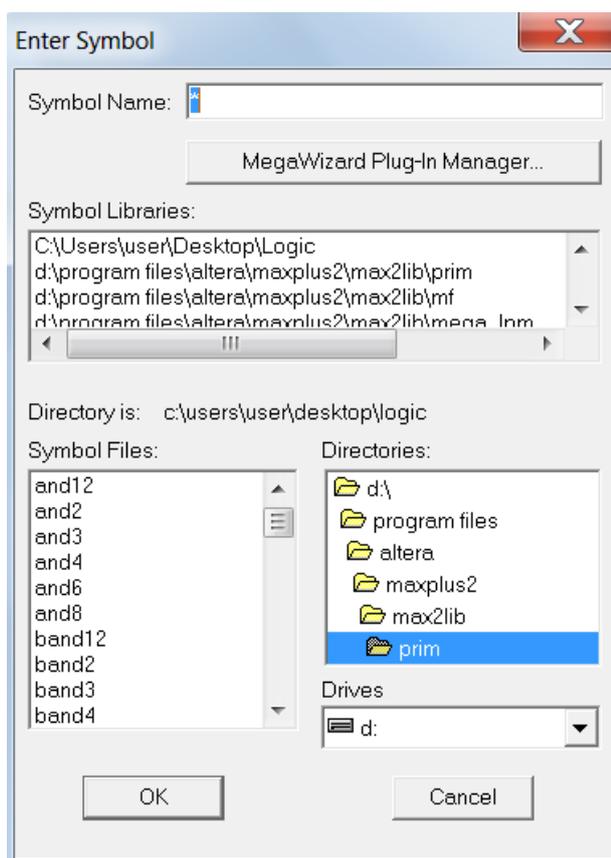


Рис. 3 Окно ввода графических элементов

Выбрать и установить необходимый элемент можно двумя способами:

1. Набрать имя элемента (примитива, мега- или макрофункции) в окне «Symbol Name» диалогового окна «Enter Symbol» и нажать «OK».
2. Выбрать необходимую библиотеку в окне «Symbol Libraries» диалогового окна «Enter Symbol» и двойным щелчком левой клавиши мыши открыть её. Затем аналогичным образом выбрать необходимый элемент в окне «Symbol File».

2.2 Проведение связей

Для проведения сигнальной линии необходимо активизировать инструменты рисования прямых отрезков; совместить указатель курсора с выводом элемента, при этом курсор автоматически превращается в инструмент рисования ортогональных линий (перекрестие). После этого проводится необходимая линия (при нажатой и удерживаемой левой кнопке мыши). На рис. 3 показан результат ввода трех элементов, входного вывода (примитив input) , инвертора (примитив not) и выходного вывода (примитив output) и их связей.

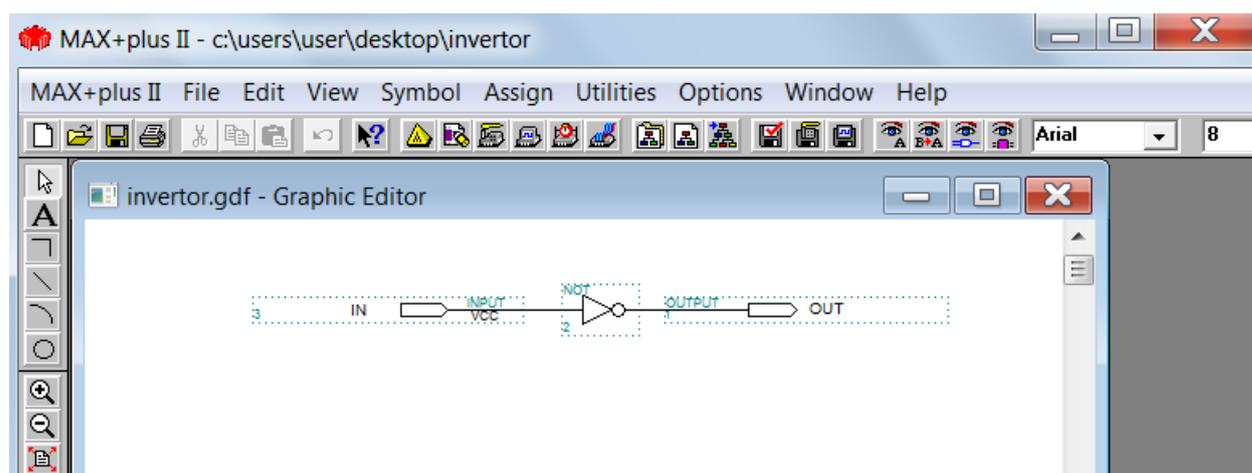


Рис. 4 Пример ввода графических элементов

Так как в схеме может быть несколько входов и выходов, они должны иметь разные имена. Редактирование имен примитивов осуществляется при помощи инструмента редактирования текста. Для соединения элементов обязательно проводить сплошные соединительные линии между выводами. Достаточно выводам, которые должны быть соединены между собой, присвоить одинаковые имена. Поименованный вывод элемента называется узлом («Node»). Для присвоения узлу имени необходимо сначала провести короткую сигнальную линию. Так как по окончании рисования линии она остаётся выделенной (выделенные элементы обозначаются красным цветом или красной рамкой), то имя узла можно вводить сразу. Если линия не выделена (имеет чёрный цвет), то необходимо выделить её щелчком левой кнопки мыши

в том месте, где должно начинаться имя узла, и набрать на клавиатуре требуемое имя. Схемы, приведенные на рис. 5, эквивалентны.

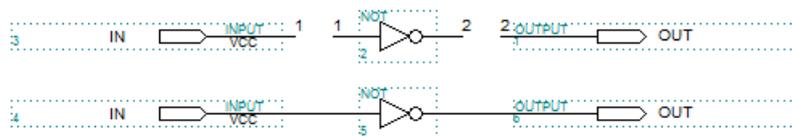


Рис. 5 Способы проведения связей между компонентами

Установленные элементы и группы элементов (любую выделенную область окна) можно удалять, перемещать, поворачивать, копировать и размножать в пределах окна редактора, а также переносить в другие окна через буфер обмена. Все эти операции можно выполнять с использованием пунктов меню «Edit», предварительно выделив требуемую область или элемент. Выделение отдельных элементов, включая и соединительные линии, производится щелчком левой кнопки мыши на изображении символа элемента. Выделение области возможно при нажатой и удерживаемой левой кнопке мыши. Выделенная область обозначается рамкой чёрного цвета, а все выделенные элементы – красным цветом. Для удаления, копирования, вырезания, поворота и некоторых других операций с элементами можно использовать выпадающее контекстное меню, которое вызывается щелчком правой кнопки мыши по выбранному элементу. Для перемещения выделенных областей в пределах окна используется способ «Drag&Drop», а для размножения выделенных областей и элементов – «Drag&Drop» при нажатой и удерживаемой клавише «Ctrl».

2.3 Шины

Редко когда схема состоит из двух или трех компонентов, имеющих небольшое количество входов и выходов, обычно их больше десятка. В этом случае проведение большого числа линий связей ухудшает читаемость схемы и загромождает пространство в окне редактора, что может привести к ошибкам и

потере времени при их выявлении. Решением этой проблемы является использование шин («Bus»). Для проведения шин необходимо в меню «Options → Line Style» выбрать жирную сплошную линию, активизировать инструменты рисования прямых отрезков, затем нажав левую клавишу мыши провести шину. На рис. 6 приведен пример использования шины.

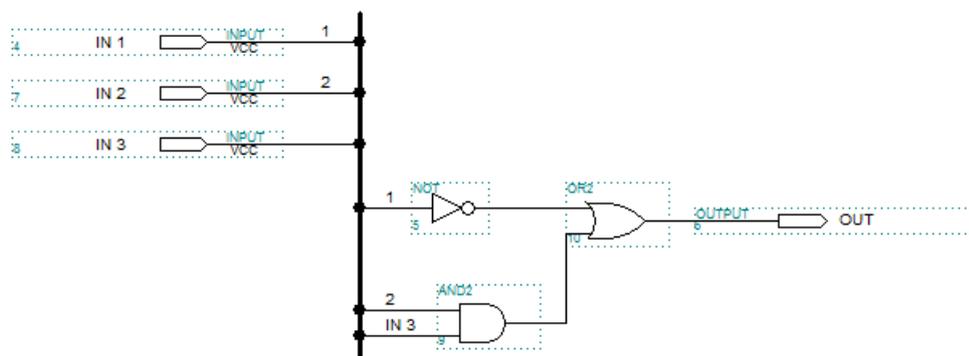


Рис. 6 Пример использования шины

Сформулируем некоторые правила использования шин:

- линия связи, заходящая на шину от входного вывода (примитив «input») именуется также как входной вывод.
- линия связи, соединяющая шину и выходной вывод (примитив «output») должна именоваться также как выходной вывод.
- для каждой линии связи, выходящей из шины, должна существовать связь с тем же именем входящая в шину.

3. Компиляция

После окончания ввода схемы можно перейти к компиляции проекта, однако перед компиляцией нелишне выполнить проверку корректности введённого проекта. Проверка осуществляется через пункт меню «File → Project → Save&Check» или нажатием комбинации клавиш «Ctrl+K». При обнаружении ошибок компилятор выдаст сообщение об ошибке в окне с заголовком «Messages – Compiler», при этом если мышью дважды щелкнуть на сообщении программа выделит красной рамкой место ошибки в окне

редактора. Если ошибок не обнаружено можно проводить полную компиляцию, результатом которой является файл прошивки микросхемы. Для полной компиляции необходимо выбрать пункт меню «File → Project → Save&Compile» или нажать клавиши «Ctrl+K». После успешной компиляции можно создать символ, что бы использовать схему в проектах более высокого уровня. Создание символа проекта осуществляется через меню «File → Create Default Symbol» – этот пункт становится доступным только после закрытия окна компилятора. Созданный символ будет помещён в каталог проекта и будет иметь такое же как проект. Использование созданных символов, так же как и элементов других библиотек, производится через диалоговое окно «Enter Symbol». На рис. 7 показан пример использования созданных символов. Символ создавался на базе схемы приведенной на рис. 4.

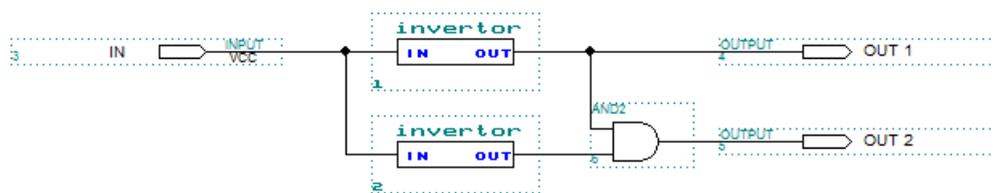


Рис. 7 Пример использования символов

MaxPlusII позволяет редактировать УГО символа. Для этого необходимо вызвать приложение «Symbol Editor» используя меню «File → Edit Symbol».

4. Назначение типа ПЛИС и выводов ПЛИС

Тип ПЛИС, необходимый для реализации проекта, может быть выбран автоматически или назначен вручную. При создании файла проекта по умолчанию установлен режим автоматического выбора минимальной по объёму ПЛИС, в которой может быть реализован данный проект. При необходимости тип ПЛИС может быть назначен вручную с помощью пункта «Assign → Device». После назначения устройства проект необходимо перекомпилировать. Выводы ПЛИС также первоначально назначаются

автоматически. После завершения работы над проектом необходимо закрепить или переназначить выводы ПЛИС, для того чтобы при возможной последующей доработке (отладке) проекта в составе всего изделия компилятор не мог изменить их назначение. Эта операция выполняется с помощью редактора «Floorplan Editor», который запускается либо через меню «MAX+plus II», либо через панель инструментов нажатием на иконку . На рис. 8 приведено окно редактора «Floorplan Editor».

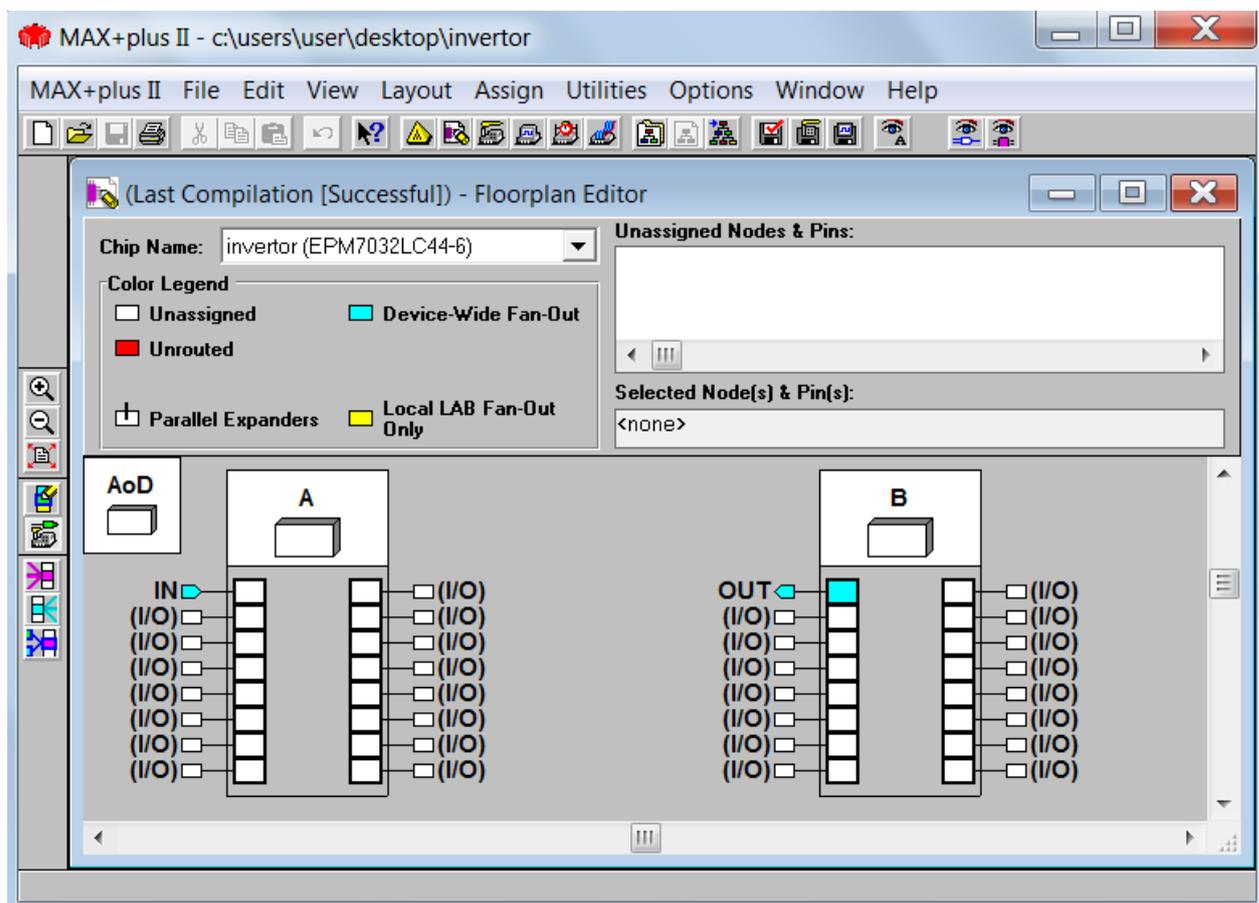


Рис. 8 Окно редактора «Floorplan Editor».

С левой стороны окна имеется панель инструментов, среди которых имеется две кнопки: одна из них устанавливает режим отображения результатов компиляции (...last compilation) – . Вторая – режим отображения текущих назначений выводов (...current assignments) – . Напомним, что назначение кнопок панелей инструментов поясняется в строке подсказки в нижней части окна. Для назначения или закрепления выводов необходимо мышью перетащить имена выводов, отображаемых в окне «Unassigned

Nodes&Pins», на соответствующие номера выводов ПЛИС, а затем перекомпилировать проект. После этого любые изменения, вносимые во внутреннюю структуру проекта, т.е. не связанные с добавлением или удалением выводов, не будут изменять назначение выводов.

Внимание ПЛИС семейства «MAX 7000» и «MAX3000» имеют встроенную систему программирования ISP, которая использует четыре вывода: #TDI, #TDO, #TMS и #TSK. Эти выводы **не должны использоваться для других целей.**

5. Моделирование и анализ работы схемы

Для проверки работоспособности спроектированной схемы необходимо провести функциональное моделирование. Для этого необходимо создать исходный файл с расширением .SCF, задать тестовые (эталонные) состояния входов, выбрать проверяемые выходы и запустить приложение «Simulator». Для создания исходного файла необходимо открыть новый файл и в диалоговом окне «New» отметить пункт «Waveform Editor file», установить в соседнем поле окна расширение .SCF и нажать «ОК», а затем сохранить его через меню «File» основного меню. После этого необходимо связать файл с проектом. Для этого необходимо войти в меню «Node» и выбрать пункт «Enter Nodes from SNF». После нажатия кнопки «List» в левой панели диалогового окна «Enter Nodes from SNF» (рис. 9) появляется список доступных узлов («Available Nodes&Groups») который необходимо перенести на правую панель («Selected Nodes&Groups»), для чего необходимо нажать кнопку со стрелкой, расположенную между панелями. После нажатия кнопки «ОК» в окне редактора появляется готовый шаблон (заготовка) для задания тестовых состояний входов.

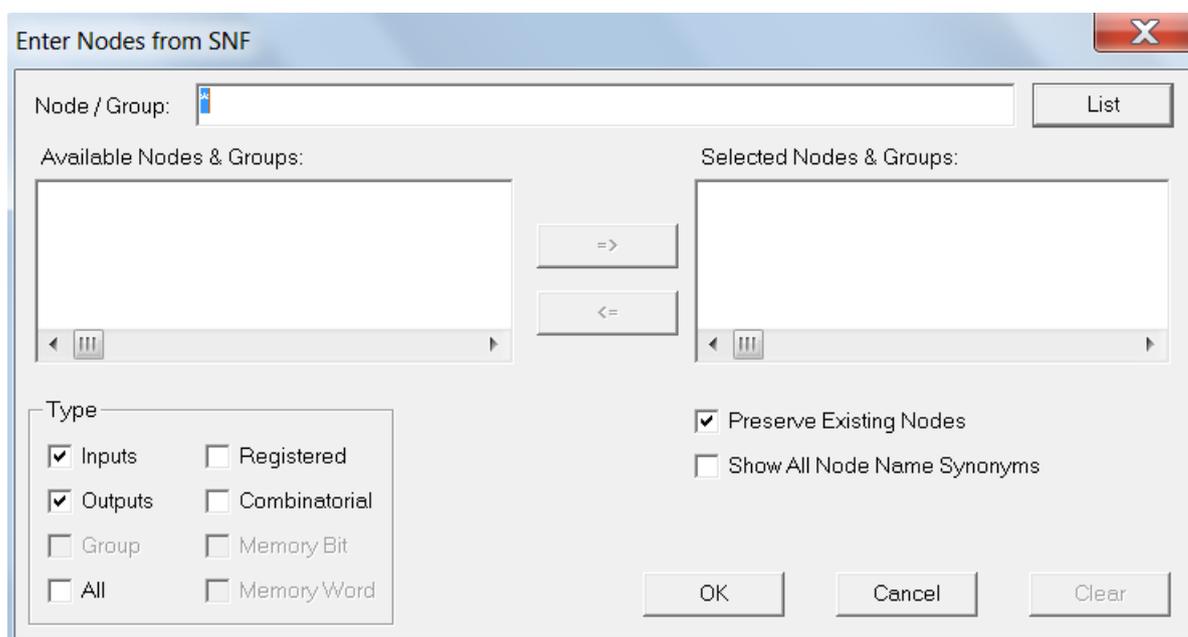


Рис. 9 Диалоговое окно «Enter Nodes from SNF»

Перед началом ввода тестовых состояний необходимо задать длительность интервала моделирования через меню «File → End Time...» и установить шаг сетки («Option → GridSize...»). При выборе временных интервалов следует учитывать, что моделирование ведется с учетом реальных задержек внутри выбранной микросхемы ПЛИС, поэтому не стоит, например выставлять шаг сетки 1 нс. Для микросхем у которых минимальная задержка от входа до выхода составляет 10 – 15 нс, так как это может привести к неверным результатам. Иногда бывает удобно рассматривать сигналы не как отдельные, а объединенными в группы. Для объединения нескольких сигналов в группу необходимо: при помощи мыши выделить нужные сигналы, нажать правую кнопку мыши и в появившемся меню выбрать пункт «Enter Group». Затем в появившемся окне задать имя группы (или принять предложенное программой) и систему счисления в которой будет отображаться значение группы (BIN – двоичная, OCT – восьмеричная, DEC – десятичная, HEX – шестнадцатеричная). Для разгруппирования необходимо выделить группу, нажать правую клавишу мыши и в появившемся меню выбрать «Ungroup».

Для задания тестовых состояний входов необходимо при помощи мыши выделить необходимый участок временного интервала и установить его в одно из требуемых состояний при помощи инструментов:

 – установка на входе низкого уровня;

 – установка на входе высокого уровня;

 – состояние входа неопределенно;

 – сигнал на входе находится в третьем (высокоимпедансном) состоянии;

 – инвертирование входного сигнала;

 – настраиваемый тактовый генератор. Активизация этого инструмента задает на входе переменный сигнал со скважностью 2 (меандр) и периодом $T=2 \cdot (n \cdot \tau)$, где n любое целое положительное число, не превышающее 32000, а τ шаг временной сетки, а также начальное состояние сигнала (0 или 1).

 – Счетный вход. Используется в основном для задания состояний групповых сигналов. При нажатии на эту иконку появляется окно в котором задается закон изменения состояния на входе группы сигналов, например для опции «Binary» закон изменения равен $K(t)=K(t-1)+k$, где $K(t)$ текущее состояние в момент времени t , значение на входе группы на предыдущем шаге $K(t-1)$, k – целое положительное число от 0 до 32000. Кроме того можно задавать период изменения по закону описанному пунктом выше, и начальное значение на входе группы («Starting Value»).

 – Задание постоянного значения на входе группы сигналов, постоянного в течение выбранного периода. Значение задается в принятой для группы системе счисления (BIN, OCT, DEC, HEX).

После того как все значения тестовых сигналов на входах схемы были установлены, необходимо приступить к моделированию. Для необходимо сначала сохранить установки принятые в «Waveform Editor», и из меню «MaxPlus II» вызвать приложение «Simulator». Далее в появившемся нажать кнопку «Start». Либо нажать на иконку .

После проведения моделирования необходимо проанализировать состояние выходов схемы и убедиться, что оно соответствует расчетному. В противном случае необходимо искать логические ошибки в схеме, исправить их и повторить процесс компиляции и моделирования. Из приведенной на рис. 10 временной диаграммы видно, что схема (рис. 4) работает верно – сигнал на выходе является инверсией входного, что и ожидалось.

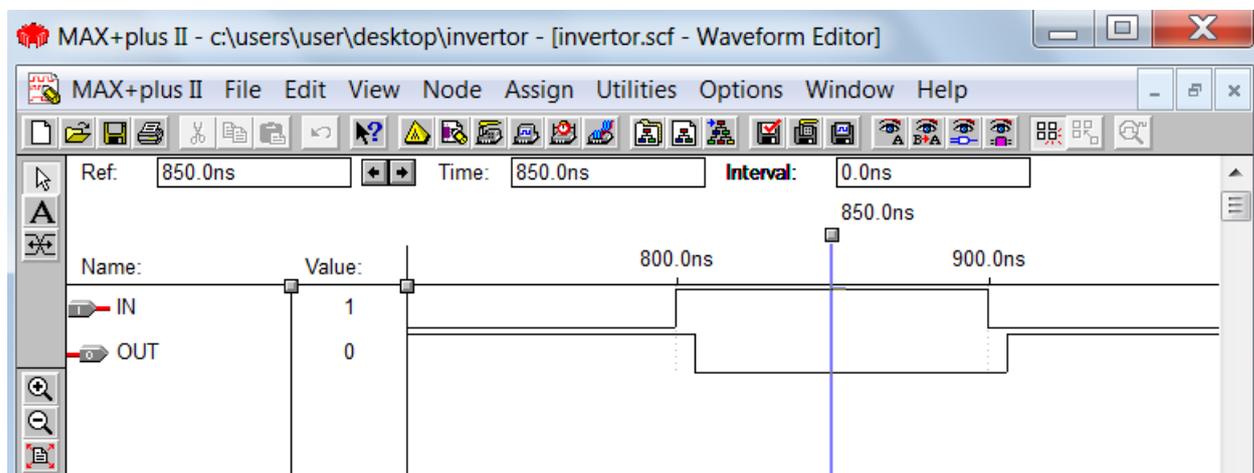


Рис. 10 Результаты моделирования схемы

6. Задания на выполнение лабораторных работ

6.1 Лабораторная работа № 1.

Цель лабораторной работы – изучение пакета «MAX+plus II». Для ознакомления с данным программным пакетом следует использовать [1]. В процессе выполнения лабораторной работы необходимо собрать произвольную схему трех входных аргументов и одним выходным сигналом. Выполнить ее моделирование и сравнить результат с расчетом, выполненным вручную.

6.2 Лабораторная работа № 2. Логические схемы

Цель лабораторной работы – выполнить синтез и исследование цифрового устройства в СДНФ и СКНФ, а также после минимизации. Собрать все три схемы и получить тождественно равные результаты по выходу. Варианты заданий приведены в табл. 1 (N – десятичный эквивалент входа).

Таблица 1

Варианты заданий

N	Входные сигналы				Вариант функции $y = f(x_1, x_2, x_3, x_4)$									
	x_1	x_2	x_3	x_4	1	2	3	4	5	6	7	8	9	10
0	0	0	0	0	0	0	1	1	0	0	1	1	0	1
1	0	0	0	1	0	1	0	1	0	0	0	1	0	1
2	0	0	1	0	1	0	0	1	0	0	0	1	0	1
3	0	0	1	1	0	1	1	0	1	1	1	1	0	1
4	0	1	0	0	1	0	1	1	1	0	0	1	0	1
5	0	1	0	1	0	1	0	1	0	0	1	0	1	1
6	0	1	1	0	1	1	0	1	0	0	0	0	1	1
7	0	1	1	1	1	1	1	0	1	1	1	0	1	0
8	1	0	0	0	1	0	0	1	1	0	1	1	0	0
9	1	0	0	1	0	0	1	1	0	0	1	1	0	0
10	1	0	1	0	0	1	0	1	0	0	0	1	0	0
11	1	0	1	1	1	1	1	0	1	1	0	1	0	0
12	1	1	0	0	1	0	0	1	1	0	1	1	0	1
13	1	1	0	1	0	0	1	1	1	0	1	0	1	1
14	1	1	1	0	0	1	1	1	1	0	0	0	1	1
15	1	1	1	1	1	0	1	0	1	1	0	0	1	1

N	Входные сигналы				Вариант функции $y = f(x_1, x_2, x_3, x_4)$									
	x_1	x_2	x_3	x_4	11	12	13	14	15	16	17	18	19	20
0	0	0	0	0	0	0	0	1	0	1	1	0	1	0
1	0	0	0	1	0	1	0	1	1	1	0	0	0	0
2	0	0	1	0	1	0	0	0	1	0	1	1	0	0
3	0	0	1	1	0	1	0	1	1	0	1	1	0	1
4	0	1	0	0	1	0	0	1	1	0	1	0	0	0
5	0	1	0	1	0	1	1	0	0	1	0	0	0	1
6	0	1	1	0	1	1	0	0	0	1	0	1	0	0
7	0	1	1	1	1	1	1	1	1	0	0	0	1	0
8	1	0	0	0	1	0	0	0	1	0	1	1	1	0
9	1	0	0	1	0	1	0	1	1	1	1	1	1	0
10	1	0	1	0	0	1	0	0	0	0	0	0	1	1
11	1	0	1	1	1	0	1	1	0	1	0	0	0	0
12	1	1	0	0	1	0	1	1	0	0	0	1	1	0
13	1	1	0	1	0	0	1	0	1	1	1	0	1	0
14	1	1	1	0	0	0	1	1	1	0	1	1	1	0
15	1	1	1	1	1	1	1	0	0	0	0	0	0	1

6.3 Лабораторная работа № 3. Комбинационные ЦУ

Цель лабораторной работы – выполнить синтез и исследование цифрового логического устройства в соответствии с исходными данными, приведенными в табл. 2. Кроме этого необходимо исследовать зависимость задержки выходного сигнала по отношению к входным сигналам при изменении назначаемых входных и выходного выводов микросхемы.

Таблица 2

Вариант	Функция $y = f(x_1, x_2, x_3, x_4)$	Базис
<i>Задания стандартной сложности</i>		
1	$\sum_{\text{обязат.}} (1,2,3,12,14) + \sum_{\text{факульт.}} (0,7,13).$ Сигнал $x_5 = 1$ инвертирует вход.	И-НЕ
2	$\sum_{\text{обязат.}} (1,2,12,14) + \sum_{\text{факульт.}} (0,7,10,13).$ Сигнал $x_5 = 1$ сдвигает вход влево.	Любой

3	$\sum(0,1,2,10,11,12,15).$ Сигнал $x_5 = 1$ сдвигает вход вправо.	Любой
4	$\sum(3,4,5,13,14,15).$ Сигнал $x_5 = 0$ сдвигает вход влево.	ИЛИ-НЕ
5	$\sum_{\text{обязат.}}(8,9,10,13,15) + \sum_{\text{факульт.}}(1,11).$ Сигнал $x_5 = 0$ сдвигает вход вправо.	И, ИЛИ, НЕ
6	$\sum_{\text{обязат.}}(1,2,3,4,8,12,14) + \sum_{\text{факульт.}}(0,2,7,10,13).$ Сигнал $x_5 = 1$ инвертирует вход.	И-НЕ
7	$\sum(0,1,10,12,15).$ Сигнал $x_5 = 1$ инвертирует вход.	ИЛИ-НЕ
8	$\sum_{\text{обязат.}}(0,2,7,12,13,15) + \sum_{\text{факульт.}}(1,4,11).$ Сигнал $x_5 = 0$ сдвигает вход влево.	И, ИЛИ, НЕ
9	$\sum(1,2,3,4,5,9,10,13,15).$ Сигнал $x_5 = 1$ сдвигает вход вправо.	И, ИЛИ, НЕ
10	$\sum_{\text{обязат.}}(1,7,12,13,14,15) + \sum_{\text{факульт.}}(8,10,11).$ Сигнал $x_5 = 1$ сдвигает вход влево.	ИЛИ-НЕ
11	$\sum(1,2,7,11,12,14,15).$ Сигнал $x_5 = 0$ сдвигает вход вправо.	И-НЕ
12	$\sum_{\text{обязат.}}(1,3,8,10,12,14) + \sum_{\text{факульт.}}(0,2,7,11,13).$ Сигнал $x_5 = 1$ инвертирует вход.	И-НЕ
13	$\sum(0,1,2,3,8,9,10,12,15).$ Сигнал $x_5 = 1$ инвертирует вход.	ИЛИ-НЕ
14	$\sum_{\text{обязат.}}(0,2,7,12,13,15) + \sum_{\text{факульт.}}(1,4,8,10,11).$	И, ИЛИ, НЕ

	Сигнал $x_5 = 0$ сдвигает вход влево.	
15	$\sum(1,2,3,4,6,9,10,14,15)$. Сигнал $x_5 = 1$ сдвигает вход вправо.	И, ИЛИ, НЕ
16	$\sum_{\text{обязат.}}(1,7,12,13,14) + \sum_{\text{факульт.}}(0,2,8,10,11)$. Сигнал $x_5 = 1$ сдвигает вход влево.	ИЛИ-НЕ
17	$\sum(1,7,11,13,14,15)$. Сигнал $x_5 = 0$ сдвигает вход вправо.	И-НЕ
18	$\sum(6,9,10,14,15)$. Сигнал $x_5 = 1$ инвертирует вход.	Любой
19	$\sum_{\text{обязат.}}(12,13,14) + \sum_{\text{факульт.}}(8,10,11)$. Сигнал $x_5 = 0$ инвертирует вход.	Любой
20	$\sum(1,2,3,4,6,9)$. Сигнал $x_5 = 1$ инвертирует вход.	Любой
21	Дешифратор 4/16. Сигнал $x_5 = 1$ переключает активную 1 на активный 0.	Любой
22	Сумматор двух двухразрядных чисел. Сигнал x_5 – перенос из предыдущего разряда.	Любой
23	Сумматор двух двухразрядных чисел. Сигнал $x_5 = 1$ преобразует второе число в дополнительный код (сумматор работает как вычитатель).	Любой
24	Компаратор двух двухразрядных чисел. Сигнал $x_5 = 1$ инвертирует второй вход.	Любой
25	Дешифратор 4/16 на основе дешифраторов 2/4. Сигнал $x_5 = 1$ переключает активную 1 на активный 0.	Любой
26	Двоично-десятичный сумматор. Параметры устройства и назначение сигнала x_5 определяет исполнитель.	Любой
27	Мультиплексор. Параметры устройства и назначение	Любой

	сигнала x_5 определяет исполнитель.	
28	Сумматор двух двухразрядных чисел. Сигнал x_5 – перенос из предыдущего разряда.	Любой
29	Компаратор двух двухразрядных чисел. Сигнал $x_5 = 1$ инвертирует второй вход.	Любой
30	Дешифратор 3/8. Сигнал $x_5 = 1$ переключает активную 1 на активный 0.	Любой

6.4 Лабораторная работа № 4. Последовательные ЦУ

Цель лабораторной работы – реализовать в САПР «MAX+plus II» цифровое логическое устройство последовательного типа и выполнить моделирование его работы и исследование. Примеры заданий и варианты приведены в табл. 3.

Таблица 3

Вариант	Тип триггера	Тип последовательного ЦУ
1	D-триггер	Делитель на 8
2	D-триггер	Делитель на 16
3	D-триггер	Делитель на 32
4	D-триггер	Делитель на 64
5	D-триггер	Суммирующий счетчик до 8
6	D-триггер	Суммирующий счетчик до 16
7	D-триггер	Суммирующий счетчик до 32
8	D-триггер	Вычитающий счетчик с 8
9	D-триггер	Вычитающий счетчик с 16
10	D-триггер	Вычитающий счетчик с 32
11	JK-триггер	Делитель на 8
12	JK-триггер	Делитель на 16
13	JK-триггер	Делитель на 32
14	JK-триггер	Делитель на 64
15	JK-триггер	Суммирующий счетчик до 8
16	JK-триггер	Суммирующий счетчик до 16
17	JK-триггер	Суммирующий счетчик до 32
18	JK-триггер	Вычитающий счетчик с 8
19	JK-триггер	Вычитающий счетчик с 16
20	JK-триггер	Вычитающий счетчик с 32

7. Требования к оформлению отчета

Отчет о проделанной работе должен содержать:

- титульный лист;
- введение с указанием целей и задач выполнения работы;
- все схемы и коды программ, необходимые для выполнения всех пунктов задания на лабораторную работу;
- эюры сигналов, подтверждающие правильность работы разработанной схемы цифрового устройства;
- заключение с подробными выводами о результатах работы.

При оформлении отчета следует пользоваться стандартом вуза ОС ТУСУР 6.1-97 [7].

Перед защитой работы необходимо представить отчет, программы моделирования и предварительно изучить контрольные вопросы по выполняемой работе.

Рекомендуемая литература

1. Официальный сайт фирмы Altera [Электронный ресурс]. – Режим доступа: www.altera.com, свободный.
2. Безуглов Д.А. Цифровые устройства и микропроцессоры / Д.А. Безуглов, И.В. Калиенко. – Изд. 2-е. – Ростов н/Д: Феникс, 2008. – 468, [1] с: ил. – (Высшее образование).
3. Браммер Ю.А. Цифровые устройства: Учеб. пособие для вузов / Ю.А. Браммер, И.Н. Пащук. – М.: Высш. шк., 2004. – 229 с.: ил.
4. Гивоне Д., Россер Р. Микропроцессоры и микрокомпьютеры: Вводный курс: Пер. с англ. – М.: Мир, 1983. – 464 с., ил.
5. Горюнов А.Г. Ливенцов С.Н. Микропроцессоры: Учеб. пособие. – Томск: Изд. ТПУ, 2005. – 89 с.
6. Клингман Э. Проектирование специализированных микропроцессорных систем: Пер. с англ. – М.: Мир, 1985. – 363 с., ил.
7. А.А. Чернышев ОС ТУСУР 6.1-97 «Работы студенческие учебные и выпускные квалификационные. Общие требования и правила оформления». Томск, 2002.