Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное

учреждение высшего образования

«Воронежский государственный лесотехнический университет

имени Г.Ф. Морозова»

Базовая кафедра технического и программного обеспечения вычислительных и информационных систем

(название кафедры)

**Пояснительная записка**

Основы языка VHDL

(тема)

09.03.02 Информационные системы и технологии

(код и наименование направления подготовки)

По дисциплине «Программирование на языке VHDL и проектирование микроэлектронных устройств»

|  |  |  |
| --- | --- | --- |
| Студент группы ИС2-191-ОБ  (номер группы)  Руководитель, доцент, к.ф-м.н  (ученая степень, ученое звание) | \_\_\_\_\_\_\_\_\_\_\_  (подпись)  \_\_\_\_\_\_\_\_\_\_  (подпись) | Бунеев И. А.\_\_\_  (инициалы и фамилия)  А.А. Караваев  (инициалы и фамилия) |

Воронеж 2022

ОГЛАВЛЕНИЕ

[ВВЕДЕНИЕ 3](#_Toc124938502)

[1 Язык описания аппаратуры интегральных схем VHDL 4](#_Toc124938503)

[1.1 Лексические элементы VHDL 5](#_Toc124938504)

[1.2 Методология проектирования на языке VHDL 8](#_Toc124938505)

[ПРАКТИЧЕСКАЯ ЧАСТЬ 11](#_Toc124938506)

[ЗАКЛЮЧЕНИЕ 15](#_Toc124938507)

[СПИСОК ЛИТЕРАТУРЫ 16](#_Toc124938508)

# ВВЕДЕНИЕ

Быстрый рост степени интеграции и функциональной сложности современных электронных устройств приводит к необходимости совершенствования и развития методов проектирования больших и сверхбольших интегральных схем (БИС и СБИС). Эта задача наиболее актуальна для специализированных интегральных схем. Это объясняется тем, что они, как правило, характеризуются нерегулярной архитектурой, обуславливающей сложность проектов, в то время как рынок электронных изделий требует разработки в короткие сроки чрезвычайно широкого ассортимента подобных устройств. Наряду с требованиями ко времени разработки необходимо обеспечить бездефектность и высокое качество проектов.

Поскольку современные СБИС содержат миллионы полупроводниковых структур на кристалле, широко использовавшийся ранее метод поэлементного проектирования архитектуры вычислительных систем (ВС) по восходящей методологии не в состоянии обеспечить бездефектное проектирование сложных ВС в приемлемые сроки. Только на этапе графического или текстового ввода схемы потребуются недопустимо большие затраты времени. Кроме того, данная методология не обеспечивает эффективную адаптацию архитектуры устройств на конкретную задачу, а также возможность описания программных средств и учет этого описания при проектировании. Ошибки, выявленные на верхних уровнях представления, приводят к необходимости повторного выполнения этапов маршрута проектирования, начиная с уровня, на котором были допущены просчеты.

# 1 Язык описания аппаратуры интегральных схем VHDL

Язык VHDL (VHSIC - Very High Speed Integrated Circuits – Hardware Description Language) разрабатывался как язык описания аппаратуры для высокоскоростных интегральных схем. Первоначальное назначение языка заключалось в обеспечении обмена проектами между различными соисполнителями работ по созданию сверхскоростных интегральных схем. Однако позже с учетом предложений и рекомендаций известных специалистов в области вычислительных систем язык был усовершенствован и стандартизован Институтом инженеров по электротехнике и радиоэлектронике (IEEE), в результате чего в 1987 году был утвержден стандарт IEEE Standard 1076 VHDL.

Язык VHDL обеспечивает высокоуровневую абстракцию описания аппаратных средств благодаря наличию как множества предопределенных типов данных, так и возможности создавать пользовательские иерархически организованные типы данных на основе базовых, заложенных в языке.

VHDL является формальной записью, предназначенной для описания функции и логической организации цифровой системы. Функция системы определяется как преобразование значений на входах в значения на выходах. Причем время в этом преобразовании задается явно. Организация системы задается перечнем связанных компонентов.

Язык предназначен для моделирования преимущественно на вентильном уровне, уровне регистровых передач и корпусов микросхем, а также успешно используется и при синтезе устройств. VHDL - не единственный язык проектирования радиоэлектронной аппаратуры ( в аббревиатура HDL в переводе с английского означает «язык проектирования устройств» - Hardware Design Language). Так, для проектирования интегральных схем широко применяется Verilog, находят применение языки Cupl, Palasm, Ample и др., но для сквозного проектирования на всех требуемых для интегральных микросхем уровнях (выше схемотехнического) предназначены только VHDL и Verilog. Преимущества VHDL по сравнению с Verilog заключаются в следующем:

• промышленный стандарт IEEE;

• не зависит от конкретной технологии или производителя;

• код может использоваться многократно

## 1.1 Лексические элементы VHDL

Текст описания на языке VHDL состоит из одного или более файлов проекта. Файл проекта представляет собой последовательность лексических элементов, каждый из которых составлен из строго определенного набора символов. Текст каждого модуля проекта является последовательностью отдельных лексических элементов. Каждый лексический элемент может быть ограничителем, идентификатором (который может являться служебным словом), абстрактным, символьным, строковым или битово-строковым литералом, либо комментарием.

VHDL поддерживает три различных стиля для описания аппаратных архитектур. Первый из них — структурное описание (structural description), в котором архитектура представляется в виде иерархии связанных компонентов.

Второй — потоковое описание (data-flow description), в котором архитектура представляется в виде множества параллельных регистровых операций, каждая из которых управляется вентильными сигналами.

Третий – поведенческое описание (behavioral description), в котором преобразование описывается последовательными программными предложениями, которые похожи на имеющиеся в любом современном языке программирования высокого уровня.

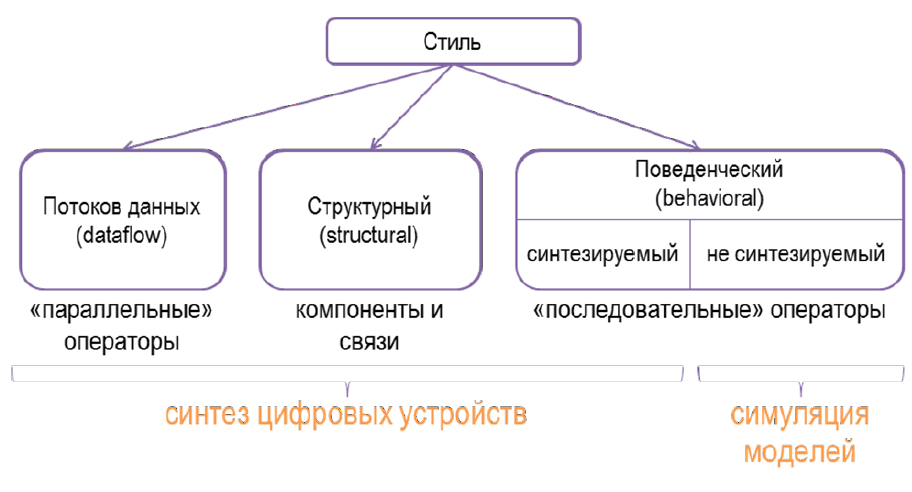


Рисунок 1 – Основные стили языка

Основными в языке VHDL являются следующие понятия.

• ОБЪЕКТ ПРОЕКТА (entity) представляет собой описание компоненты проекта, имеющей чётко заданные входы и выходы и выполняющей чётко определённую функцию. Объект проекта может представлять всю проектируемую систему, некоторую подсистему, устройство, узел, стойку, плату, кристалл, макроячейку, логический элемент и т. п. В описании объекта проекта можно использовать компоненты, которые, в свою очередь, могут быть описаны как самостоятельные объекты проекта более низкого уровня.

• ОБЪЯВЛЕНИЕ ОБЪЕКТА ПРОЕКТА (entity declaration) описывает интерфейс и определяет только входы и выходы объекта проекта.

• Для описания поведения объекта или его структуры служит АРХИТЕКТУРНОЕ ТЕЛО (architecture body). Чтобы задать, какие объекты проекта использованы для создания полного проекта, используется ОБЪЯВЛЕНИЕ КОНФИГУРАЦИИ (configuration declaration).

• В языке VHDL предусмотрен механизм пакетов для часто используемых описаний, констант, типов, сигналов. Эти описания помещаются в ОБЪЯВЛЕНИИ ПАКЕТА (package declaration).

• Если пользователь использует нестандартные операции или функции, их интерфейсы описываются в объявлении пакета, а тела содержатся в ТЕЛЕ ПАКЕТА (package body).

Таким образом, при описании цифровых схем на языке VHDL, возможно использование пяти различных типов описаний: объявление объекта проекта, архитектурное тело, объявление конфигурации, объявление пакета и тело пакета. Каждое из описаний является самостоятельной конструкцией языка VHDL, может быть независимо проанализирована анализатором и поэтому получило название “МОДУЛЬ ПРОЕКТА” (design unit).

Средства VHDL для отображения структур цифровых систем базируются на представлении о том, что описываемый объект entity представляет собой структуру из компонент component соединяемых друг с другом линиями связи. (Компонент часто называют «черным ящиком» - он является аналогом объявления интерфейса объекта, реализованного в библиотеке) Каждая компонента, в свою очередь, является объектом и может состоять из компонент низшего уровня (иерархия объектов). Взаимодействуют объекты путем передачи сигналов signal по линиям связи. Линии связи подключаются к входным и выходным портам компонент. В VHDL сигналы отождествляются с линиями связи. писание структуры объекта строится как описание связей конкретных компонент, каждая из которых имеет имя, тип и карты портов. Карта портов port map определяет соответствие портов компонент поступающим на них сигналам, можно интерпретировать карту портов как разъем, на который приходят сигналы и в который вставляется объект-компонента.

Архитектура любого проекта, написанного на языке VHDL, включает в себя описание одного или нескольких параллельных процессов. Описание процесса состоит из последовательности операторов, отображающих действия по переработке информации. Все операторы внутри процесса выполняются последовательно. Процесс может находиться в одном из двух состояний- либо пассивном, когда процесс ожидает прихода сигналов запуска или наступления соответствующего момента времени, либо активном- когда процесс исполняется. Процессы взаимодействуют путем обмена сигналами.

Кроме описанных выше средств языка, в программах нередко используются массивы. Массив представляет собой упорядоченную структуру однотипных данных. Массив имеет диапазон индексов, который может быть возрастающим либо убывающим. На любой элемент массива можно сослаться, используя его индекс.

Объявление ограниченного типа "массив" имеет вид:

TYPE имя\_типа IS

ARRAY (диапазон\_индексов [ , диапазон\_индексов] )

OF тип\_элемента;

диапазон\_индексов может определяться двумя способами:

1) явным заданием границ диапазона

m TO n

n DOWNTO m

где m, n – целочисленные константы, m <= n;

2) с использованием идентификатора ограниченного подтипа. В этом случае значения границ подтипа являются значениями границ индекса массива.

Все описанные элементы являются примером использования поведенческого стиля описания.

## 1.2 Методология проектирования на языке VHDL

Широкое применение язык VHDL нашел при проектировании устройств на так называемых программируемых интегральных микросхемах (сокращенно ПЛИС).

На данный момент на рынке ПЛИС представлены микросхемы FPGA- и CPLD-структур. FPGA ( Field Programmable Gate Array) – это микросхема, конфигурируемая пользователем. Она состоит из ячеек, часть из которых отвечает за реализацию элементарных функций (Configurable Logic Blocks CLB) а часть – за внутренние соединения (Programmable Switch Matrices PSM). Схема FPGA представлена на рисунке 2.



Рисунок 2 – FPGA структура

ПЛИС на FPGA-архитектуре основаны на ОЗУ и могут быть перепрограммированы бесконечное число раз. Но из-за этого у таких ПЛИС появляется существенный недостаток: при отключении питания программа теряется и для предотвращения ее потери нужен специальный конфигуратор. У ПЛИС с CPLD-структурой отсутствует проблема потери программы при отключении питания, т.к. они основаны на флэш-памяти. Поэтому программа может храниться постоянно, к тому же она может быть защищена битом секретности. Но вместе с тем у таких ПЛИС появляется другой недостаток: микросхема может быть перепрограммируема ограниченное число раз.

Процесс проектирования устройства является многоступенчатым (рисунок 3).



Рисунок 3 – Процесс проектирования устройства

Разработку интегральных схем можно выполнять, опираясь на различные уровни абстрактного представления проекта. Наиболее высокий уровень абстракции можно представить алгоритмическими выражениями, описывающими логику работы устройства. Кроме того, стандарт IEEУ Standart 1076 VHDL позволяет компиляторам, реализующим VHDL, допускать встраивание описаний различных подпрограмм на языках программирования высокого уровня (Си, Паскаль и т.д.) в качестве реализации функций и процедур. В этом стиле удобно разрабатывать описание устройств высокого уровня сложности (ЭВМ, процессоры, контроллеры различных устройств, сложные последовательные схемы и т.д.). При поведенческом описании компоненты могут вводиться на любых иерархических уровнях, что позволяет пользоваться библиотечными наборами готовых логических и архитектурных решений. Немаловажное значение для качества проектирования имеет возможность оптимизации исходного поведенческого описания. Так, например, введение неоправданно большого количества дополнительных, промежуточных сигналов и переменных способно привести к совершенно неработоспособному проекту.

# ПРАКТИЧЕСКАЯ ЧАСТЬ

Задание: для заданной нерегулярной логической схемы составить структурное VHDL-описание, выполнить моделирование на разных наборах значений входных переменных, построить систему логических функций, реализуемую схемой как на рисунке 4.



Рисунок 4 – Система логических функций

Практическая часть была выполнена в среде разработки Active-HDL.

В таблице 1 представлены логические элементы, которые были использованы в ходе построения системы логических функций.

|  |  |  |
| --- | --- | --- |
| Имя элемента | Функция элемента | Задержка (ns) |
| N |  | 1 |
| NA2 |  | 2 |
| EX2 |  | 5 |
| NAO2 |  | 3 |
| NAO3 |  | 5 |
| NA3 |  | 3 |
| A3 |  | 3 |
| NAOA2 |  | 4 |
| NA2 |  | 2 |
| VCC |  | 1 |

Таблица 1 – используемые логические элементы

Все логические элементы в формате vhd представлены на рисунке 5.

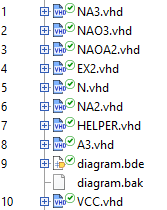


Рисунок 5 – список логических элементов

После разработки всех логических элементов мы строим схему логических функций согласно заданию (Рисунок 6).

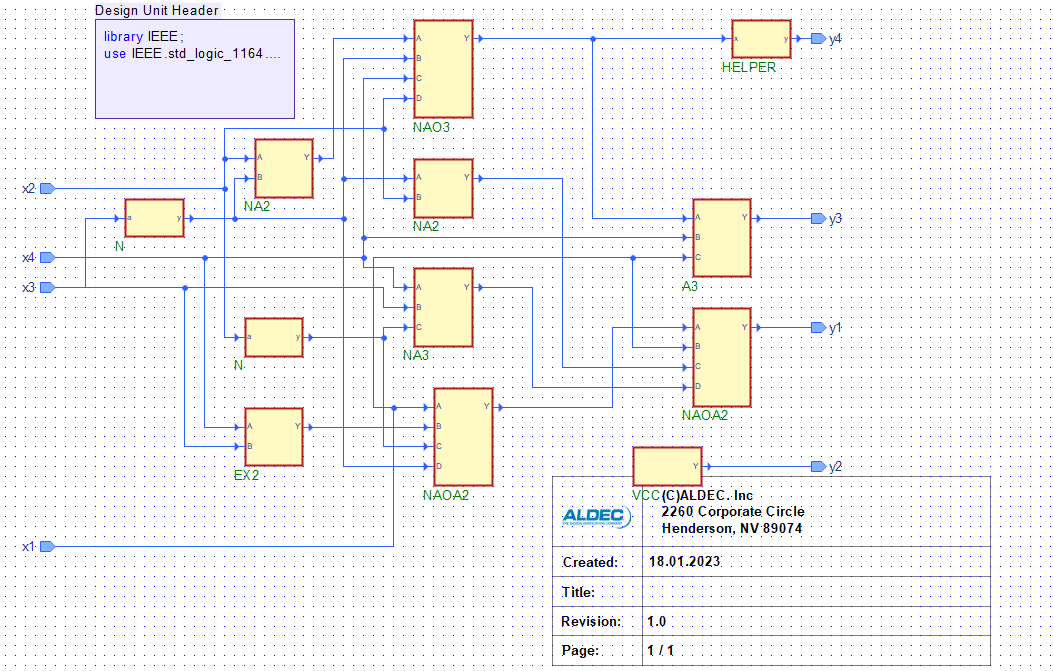


Рисунок 6 – схема логических функций

Далее запускаем симуляцию на вкладке Waveform. Указываем значения входных параметров x1, x2, x3, x4. Для значения x1 – начальное значение 0, изменения каждые 100нс, для x2 – начальное значение 1, изменение каждые 100нс, для значения х3 – начальное значение 0, изменение каждые 50нс, для значения х4 – начальное значение 1, изменение каждые 50нс.

На рисунке 7 представлен результат симуляции работы схемы логических функций.

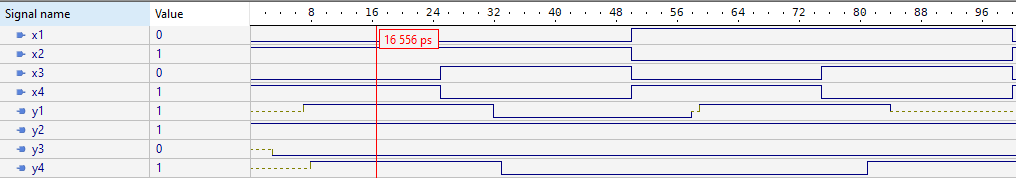


Рисунок 7 – результат симуляции

На рисунке 8 представлена модель работы и изменения сигналов внутри схемы логических функций

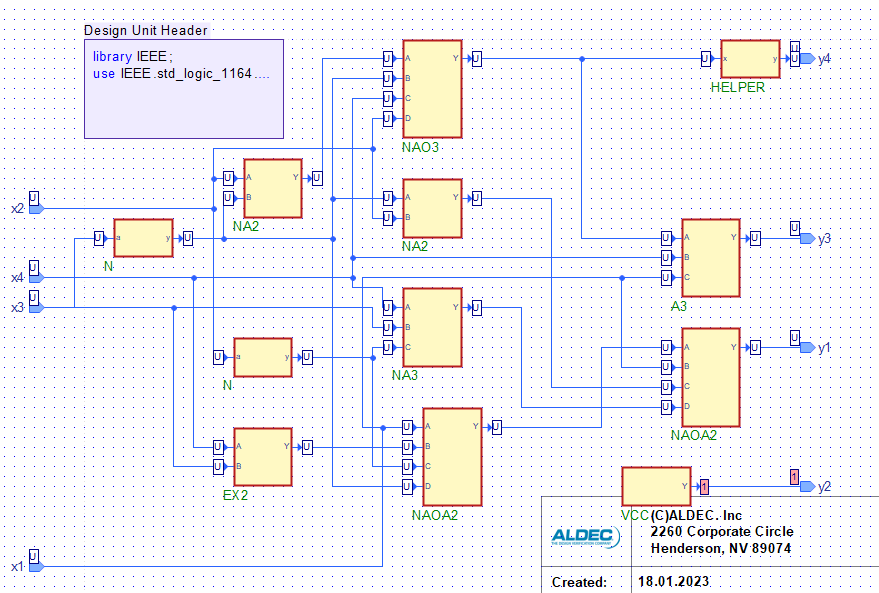


Рисунок 8 – результат моделирования изменения сигналов

На рисунке 8 и 9 представлены примеры логических элементов на языке VHDL

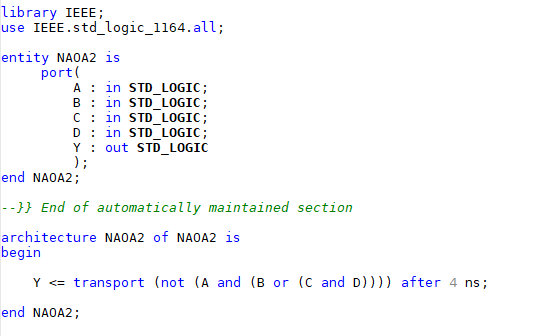


Рисунок 8 – элемент NAOA2

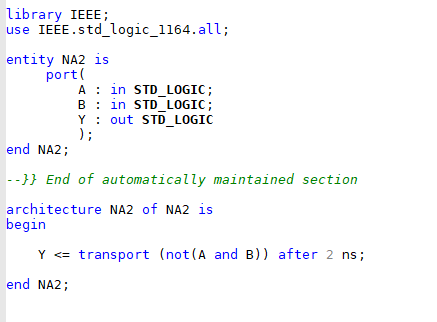


Рисунок 9 – элемент NA2

# ЗАКЛЮЧЕНИЕ

Язык VHDL предназначен для решения комплекса задач в ходе проектирования. Это и описания структуры системы, декомпозиции системы на подсистемы, спецификации связей и взаимодействия подсистем. Так же моделирования системы и ее работы на основе четкой спецификации структуры системы. И синтез схемотехнической реализации системы, автоматической генерации детальной структуры на основе строгой спецификации системы на языке VHDL – спецификации на более абстрактном уровне.

Язык VHDL обеспечивает высокоуровневую абстракцию описания аппаратных средств благодаря наличию как множества предопределенных типов данных, так и возможности создавать пользовательские иерархически организованные типы данных на основе базовых, заложенных в языке.

VHDL является формальной записью, предназначенной для описания функции и логической организации цифровой системы. Функция системы определяется как преобразование значений на входах в значения на выходах. Причем время в этом преобразовании задается явно. Организация системы задается перечнем связанных компонентов. Язык предназначен для моделирования преимущественно на вентильном уровне, уровне регистровых передач и корпусов микросхем, он успешно используется и при синтезе устройств.

# СПИСОК ЛИТЕРАТУРЫ

1. Максфилд К., Проектирование на ПЛИС. Архитектура, средства и методы. Курс молодого бойца (The Design Warrior's Guide to FPGA's ). М.: Додэка XXI, 2015, - 408.

2. Перельройзен . Е.А. . Проектируем на VHDL. М.: СОЛОН-Пресс, 2014, - 80 с.

3. Поляков А. К. Языки VHDL и VERILOG в проектировании цифровой аппаратуры. М.: СОЛОН-Пресс,2003.

4. Сергиенко А.М. VHDL для проектирования вычислительных устройств. Киев: ЧП. «Корнейчук», ООО «ТИД «ДС», 2003.

5. Суворова Е.А., Шейнин Ю.Е. Проектирование цифровых систем на VHDL. СПб.: БХВ-Петербург,2003.

6. Эйкерли Дж. Ф. Проектирование цифровых устройств. В 2-хт. Т. 1 — 2. М.: Постмаркет, 2002.