VHDL语言基础（一）

单击输入您的封面副标题

VHDL语言基础

简介

VHDL程序基本结构

数据对象、数据类型、运算符和属性

进程

顺序语句

并行语句

集成电路的描述方式

门级描述

- 随着设计规模增大难以管理
- 要求更高抽象层次描述方法

图形和布尔方程式的描述方式

- 费时、易出错，且在方程式中寻找错误很困难
- 原理图的保持比较困难，需要一个文本来描述其设计构思和功能
- 图形的输入环境往往也是专用

硬件描述语言 HDL

- Hardware Description Language

硬件描述语言

用于电子系统硬件描述的语言

- 与高层次的软件设计语言类似
- 不同点
  - 主要目的是用来编写硬件设计文件并建立硬件器件的仿真模型
  - 语意和语法的定义是为了能够描述硬件的行为

功能

- 在希望的抽象层次上，可以对设计进行精确而简练的描述
- 在不同层次上都易于形成用语言模拟和验证的设计描述
- 在自动设计系统中作为设计输入
- 易于设计的修改，易于把相应的修改并入设计文件中

VHDL的优点

描述能力强

- 具有功能强大的语言结构
- 用简洁明确的代码描述复杂的逻辑设计
- 它支持多层次的设计描述
- 支持设计库和可重用设计模块

设计和工艺、器件无关

- 设计不依赖于工艺和器件
- 同一设计，可以用多种不同的方法实现
- 设计人员可以专注于设计

VHDL的优点

可移植、设计易于共享和复用

用VHDL描述的设计可以被多种工具支持

效率高，成本低

语言描述快捷、方便，大大提高了数字系统的设计速度

和可编程逻辑器件结合，可以使产品以很快的速度面市

VHDL代码可以很容易地转为ASIC的设计

VHDL程序的基本结构

VHDL把任意复杂度的电路模块看作一个单元

一个单元又可以分为接口部分和设计描述部分

一个VHDL程序包括五个部分

实体（Entity）

描述设计的外部接口信号和该设计单元的公共信息

结构体（Architecture）

描述该设计单元的行为、数据的流程或结构

库（Library）

存放已编译的实体、结构体、程序包等，用户可以直接引用

程序包（Package）

存放设计可以共享的数据类型、常数和子程序等

配置（Configuration）

VHDL 大小写不敏感

MUX.vhd

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
ENTITY MUX IS
PORT(
        A, B : IN STD_LOGIC;
        SEL : IN STD_LOGIC;
        C : OUT STD_LOGIC);
END MUX;
ARCHITECTURE MUX_EXAMPLE OF MUX IS
BEGIN
        PROCESS(SEL, A, B)
        BEGIN
                IF SEL = '1' THEN
                        C <= A;
                ELSE
                        C <= B;
                END IF;
        END PROCESS;
END MUX_EXAMPLE;
```

库

程序包

实体

结构体

文件名和实体名一致

每行；结尾

关键字end后跟实体名

关键字begin

关键字end后跟构造体名

实体ENTITY

VHDL中的基本单元和最重要的抽象

- 可以代表整个系统
- 可以代表一块电路板
- 可以代表一个芯片
- 可以代表一个单元或一个门电路

组成

- 实体名
- 类属表
- 端口表
- 实体说明部分
- 实体语句

给实体命名，并给实体定义一个接口

- 从某种程度上讲，实体是一个器件的外部视图

实体说明格式

```
ENTITY 实体名 IS
        [GENERIC （类属表）;]
        [PORT （端口表）；]
        [实体说明部分；]
        [BEGIN
                实体语句部分；]
END [ENTITY] [实体名];
```

类属说明

- 为设计实体和其外部环境通信的静态信息提供通道
  - 用来规定端口的大小
  - 实体中的子元件数目
  - 实体的定时特性等
- 是可选项，放在端口说明之前
- 类属说明的一般格式

```
GENERIC ([COSTANT] 名字表：[IN] 子类型标示[:= 静态表达式], ……);
```

例如：

```
GENERIC(N：POSITIVE:= 3);
```

端口说明

- 每一个I/O信号都被称为端口，其功能对应于电路图符号的一个引脚
- 每个端口必须有
  - 一个名字
  - 通信模式
    - 说明数据通过该端口的流动方向
  - 数据类型
    - 说明流过该端口的数据类型
- 端口说明的一般格式

```
PORT（[SIGNAL] 名字：[模式] 子类型标识 [BUS] [：= 静态表达式]，….）；
```

通信模式

IN模式

输入模式，只允许信号流入实体

OUT模式

输出模式，只允许信号流出实体

INOUT模式

双向模式，既可以流入，也可以流出

双向模式可以替代输入、输出和缓冲模式

实际的设计中，只有纯粹的双向信号才使用双向模式

BUFFER模式

缓冲模式 ，和输出模式的端口类似

既可以用于输出，也可以用于反馈

out与buffer的区别

```
entity test1 is
   port(a: in std_logic;
      b,c: out std_logic );
end test1;

architecture a of test1 is
begin
      b <= not(a);
      c <= b;--Error
end a;
```

```
entity test2 is
     port(a: in std_logic;
       b: buffer std_logic;
       c: out std_logic );
end test2;

architecture a of test2 is
begin
        b <= not(a);
        c <= b;
 end a;
```

数据类型

- IEEE1076/93标准规定的数据类型
  - 布尔型（boolean）
  - 位型（bit）
  - 位矢量型（bit_vector）
  - 整数型（integer）

```
ENTITY  MUX  IS
      GENERIC (DELAY : =  5ns) ;
      PORT (
            IN1, IN2, SEL  : IN  BIT;
            OUTPUT :  OUT  BIT) ;
END  ENTITY  MUX ;
```

多选题 1分

设置

VHDL代码包括以下哪几个部分

A 库和程序包

B 实体

C 结构体

D 配置

提交

多选题 1分

设置

以下说法哪些是正确的是：

A VHDL大小写敏感

B 文件名应和实体名一致

C 文件名可以不和实体名一致

D VHDL大小写不敏感

提交

多选题 1分

设置

关于实体，下列说法正确的是：

A 从某种程度上讲，实体是一个器件的外部视图

B 在实体中可以描述电路的行为或结构

C 实体中的语句不描述电路的行为或结构

D 实体可以是空的

提交

多选题 1分

设置

关于信号的通信模式，下列说法正确的是：

A 输入IN只允许信号流入

B 输出OUT只允许信号流出

C BUFFER和OUT都用于输出，因此是一样的

D 所有的端口都可以定义为INOUT

提交

结构体

- 描述实体的内在
  - 描述一个实体的功能
  - 规定实体的数据流程
  - 定义实体中内部单元的连接关系
- 结构体的一般格式

```
ARCHITECTURE  结构体名  OF  实体名  IS
        [说明部分]
BEGIN
        [并行语句]
END  结构体名;
```

对结构体内部所使用的信号、常数、数据类型、元件和子程序等进行声明和定义

用于具体地描述结构体的行为及其连接关系

行为描述

- 对设计中的描述是按算法的路径来描述的
- 优点在于无需关注设计的门级实现，而只需注意正确的函数模型

例：等值比较器的功能，当信号A和B相等时，EQUALS有效

```
LIBRARY  IEEE;
USE  IEEE.STD_LOGIC_1164.ALL;
ENTITY  EQCOMP4  IS
       PORT(
              A, B : IN STD_LOGIC_VECTOR(3  DOWNTO 0);
              EQUALS:  OUT  STD_LOGIC);
END  EQCOMP4;

ARCHITECTURE  BEHAVAL  OF  EQCOMP4  IS
BEGIN
       COMP:  PROCESS(A, B)
       BEGIN
              IF  A = B  THEN
                     EQUALS <= '1';
              ELSE
                     EQUALS <= '0';
              END  IF;
       END  PROCESS  COMP;
END  BEHAVAL;
```

不考虑在电路中到底是怎样实现的。

数据流描述

- 描述数据流的运动路径和运动方向
- 主要使用并行信号赋值语句，既显式表示了该设计单元的行为，也隐式表示了该设计单元的结构

例：等值比较器的功能，当信号A和B相等时，EQUALS有效

```
LIBRARY  IEEE;
USE  IEEE.STD_LOGIC_1164.ALL;
ENTITY       EQCOMP4  IS
      PORT(
            A,  B: IN  STD_LOGIC_VECTOR(3  DOWNTO  0);
            EQUALS:  OUT  STD_LOGIC);
END  EQCOMP4;

ARCHITECTURE  DATAFLOW  OF  EQCOMP4  IS
BEGIN
      EQUALS <= '1' WHEN  (A = B) ELSE  '0';
END  DATAFLOW;
```

结构化描述

- 用VHDL来描述网表组成
- 和图形网表非常相似：元件被例化，且用信号互相连接
- 通常用于层次设计

例：等值比较器的功能，当信号A和B相等时，EQUALS有效

```
LIBRARY  IEEE;
USE  IEEE.STD_LOGIC_1164.ALL;
ENTITY       EQCOMP4  IS
      PORT(
            A,  B: IN  STD_LOGIC_VECTOR(3  DOWNTO  0);
            EQUALS:  OUT  STD_LOGIC);
END  EQCOMP4;

ARCHITECTURE  DATAFLOW  OF  EQCOMP4  IS
BEGIN
      EQUALS <= '1' WHEN  (A = B) ELSE  '0';
END  DATAFLOW;
```

结构化描述

- 用VHDL来描述网表组成
- 和图形网表非常相似：元件被例化，且用信号互相连接
- 通常用于层次设计

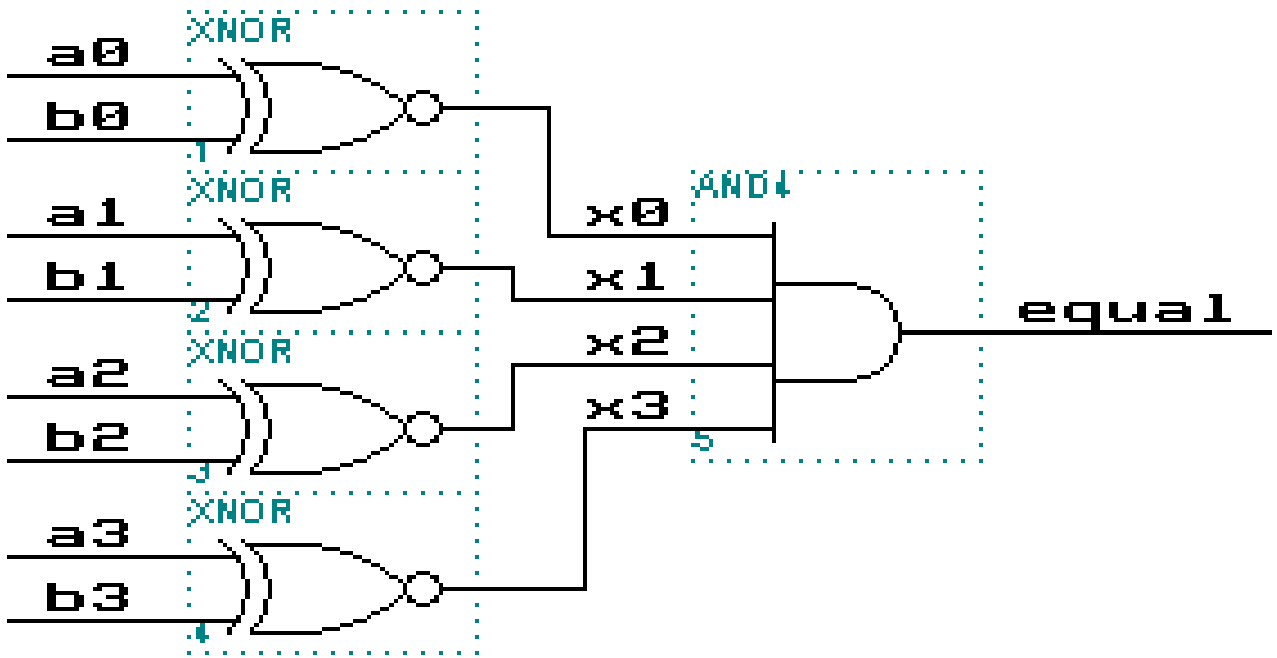
例：等值比较器的功能，当信号A和B相等时，EQUALS有效

```
LIBRARY  IEEE;
USE  IEEE.STD_LOGIC_1164.ALL;
USE  WORK.GATESPKG.ALL;
ENTITY       EQCOMP4  IS
      PORT(
            A,  B:IN  STD_LOGIC_VECTOR(3  DOWNTO  0);
            EQUALS:  OUT  STD_LOGIC);
END  EQCOMP4;
```

```
ARCHITECTURE  STRUCTURE  OF  EQCOMP4  IS
        COMPONENT XNOR2
        PORT(
                A, B : IN STD_LOGIC;
                C : OUT STD_LOGIC);
        END COMPONENT;
        COMPONENT AND4
        PORT(
                A, B, C, D : IN STD_LOGIC;
                E : OUT STD_LOGIC);
        END COMPONENT;
        SIGNAL  X : STD_LOGIC_VECTOR (0 TO 3) ;
BEGIN
        U0: XNOR2  PORT  MAP (A (0) , B (0) , X (0) ) ;
        U1: XNOR2  PORT  MAP (A (1) , B (1) , X (1) ) ;
        U2: XNOR2  PORT  MAP (A (2) , B (2) , X (2) ) ;
        U3: XNOR2  PORT  MAP (A (3) , B (3) , X (3) ) ;
        U4: AND4  PORT  MAP (X (0) , X (1) , X (2) , X (3) ,
                                EQUALS) ;
END  STRUCTURE;
```

•类似于电路的网络表，将各个器件通过语言的形式进行连接，与电路有一一对应的关系。

•一般用于大规模电路的层次化设计时。



单选题 1分

设置

关于结构体，下面说法正确的是：

A 结构体由顺序语句构成，顺序执行

B 结构体由并行语句构成，并行执行

提交