

第四讲

一、译码器

联系：编写 3-8 二进制译码器和 4-16 译码器

注意：

- 1、在 verilog 中 “+” 表示数字加，而在数字电路中 “+” 表示或。
- 2、在 verilog 中 “x” 表示不定值，而在数字电路中 “x” 表示不管取何值。
- 3、编程时脑中要有数字电路的结构。

语法

第九章 Verilog HDL模型的不同抽象级别

9.2 Verilog HDL 的行为描述建模

Verilog HDL 行为描述建模不仅可用于产生仿真测试信号对已设计的模块进行检测，也常常用于复杂数字逻辑系统的顶层设计

9.3 用户定义的原语

利用 UDP 来定义有自己特色的用于仿真的基本逻辑元件模块并建立相应的原语库。

注意：

- 1) UDP 只能有一个输出端，而且必定是端口说明列表的第一项。
- 2) UDP 可以有多个输入端，最多允许有 10 个输入端。
- 3) UDP 所有端口变量必须是标量，也就是必须是 1 位的。
- 4) 在 UDP 的真值表项中，只允许出现 0、1、X 三种逻辑值，高阻值状态 Z 是不允许出现的。
- 5) 只有输出端才可以被定义为寄存器类型变量。
- 6) initial 语句用于为时序电路内部寄存器赋初值，只允许赋 0、1、X 三种逻辑值，缺省值为 X。

第十章 如何编写和验证简单的纯组合逻辑模块

10.1 加法器

1、用 Verilog HDL 来描述加法器是相当容易的，只需要把运算表达式写出就可以了，见下例。

```
module add_4( X, Y, sum, C);  
input [3 : 0] X, Y;  
output [3: 0] sum;  
output C;  
  
assign {C, Sum } = X + Y;  
  
endmodule
```

2、根据用户对运算速度的要求插入流水线结构

3、延时控制

10.2 乘法器

10.3 比较器

10.4 多路器

10.5 总线和总线操作

10.6 流水线 (pipeline)

练习：使用数码管显示 4 位计数器。

总结：

1、 小数点为 Pin 914

2、 刷新速率可为每秒 200 次。

3、 时钟