

状态机练习

1、题目：

利用按键控制计数器。有两个按键，一个控制开始计数与否，一个为计数按钮。

2、总结

- (1) 先画出状态图，对写好状态机很有帮助。
- (2) 三段式状态机的第一、三段为时序逻辑，应写为非阻塞赋值，而第二段为组合逻辑，应写为阻塞赋值。
- (3) 三段式状态机的第二段中，敏感列表要写全。
- (4) 默认取值要写好。

第十三章 设计可综合的状态机的指导原则

13.1 用Verilog HDL语言设计可综合的状态机的指导原则

- 1、在设计采用 FPGA 实现的状态机时往往采用独热码状态机
- 2、建议采用 `case`, `casex` 或 `casez` 语句来建立状态机的模型
- 3、不能将缺省状态设置为某一确定的状态 S1
- 4、状态机应该有一个异步或同步复位端，
- 5、用 Verilog HDL 描述的状态机应明确地由唯一时钟触发。

如果设计要求必须有不同的时钟触发的状态机，可以采用以下办法：编写另一个模块，在那个模块中使用另外一个时钟。然后用实例引用的方法在另外一个模块中把它们联接起来。为了使设计比较简单调试比较容易，应该尽量使这两个状态机的时钟有一定的关系。例如甲模块的时钟是乙模块时钟同步计数器的输出。把异步触发的电路转换为同步时钟触发的电路并不困难，我们将在 15 章例[3]中见到这样一个例子。

- 6、注意千万不要使用综合工具来设计异步状态机。

13.2 典型的状态机实例

13.3 综合的一般原则

- 1) 综合之前一定要进行仿真,这是因为仿真可以暴露逻辑错误,所以建议大家这样做。如果不做仿真,没有发现的逻辑错误会进入综合器,使综合的结果产生同样的逻辑错误。
- 2) 每一次布局布线之后都要进行仿真,在器件编程或流片之前要做最后的仿真。
- 3) 用 Verilog HDL 描述的异步状态机是不能综合的,因此应该避免用综合器来设计,若一定要设计异步状态机,则可用电路图输入的方法来设计。
- 4) 若要为电平敏感的锁存器建模,使用连续赋值语句是最简单的方法。

13.4 语言指导原则

1、always 块

- 1) 每个 always 块只能有一个事件控制"@(event-expression)",而且要紧跟在 always 关键字后面。
- 2) always 块可以表示时序逻辑或者组合逻辑,也可以用 always 块既表示电平敏感的透明锁存器又同时表示组合逻辑。但是不推荐使用这种描述方法,因为这容易产生错误和多余的电平敏感的透明锁存器。
- 3) 带有 posedge 或 negedge 关键字的事件表达式表示沿触发的时序逻辑,没有 posedge 或 negedge 关键字的表示组合逻辑或电平敏感的锁存器,或者两种都表示。在表示时序和组合逻辑的事件控制表达式中如有多个沿和多个电平,其间必须用关键字“or”连接。
- 4) 每个表示时序 always 块只能由一个时钟跳变沿触发,置位或复位最好也由该时钟跳变沿触发。
- 5) 每个在 always 块中赋值的信号都必需定义成 reg 型或整型。整型变量缺省为 32bit,使用 Verilog 操作符可对其进行二进制求补的算术运算。综合器还支持整型量的范围说明,这样就允许产生不是 32 位的整型量。句法结构: integer[<msb>:<lsb>]<identifier>。
- 6) always 块中应该避免组合反馈回路。即在三段式状态机的第二段中,赋值表达式右端的信号,必须都放到 always 敏感电平列表中。

2、赋值

- 1) 对一个寄存器型(reg)和整型(integer)变量给定位的赋值只允许在一个 always 块内进行,如在另一 always 块也对其赋值,这是非法的。

- 2) 把某一信号值赋为 'bx，综合器就把它解释成无关状态，因而综合器为其生成的硬件电路最简洁。

13.5 可综合风格的Verilog HDL模块实例

13.5.1 组合逻辑电路设计实例

[例 6]. 8-3 编码器的设计实例

编码器设计方案：----- 代码非常简洁

```
module encoder2 ( none_on, out2, out1, out0, h, g, f, e, d, c,
b, a);
    input h, g, f, e, d, c, b, a;
    output none_on, out2, out1, out0;
    wire [3:0] outvec;

    assign outvec= h? 4'b0111 : g? 4'b0110 : f? 4'b0101:
e? 4'b0100 : d? 4'b0011 :c? 4'b0010 : b? 4'b0001:
a? 4'b0000 : 4'b1000;

    assign {none_on,out2,out1,out0} = outvec;
endmodule
```

[例 8]. 奇偶校验位生成器设计实例

```
module parity( even_numbits,odd_numbits,input_bus);
    output even_numbits, odd_numbits;
    input [7:0] input_bus;
    assign odd_numbits = ^input_bus;
    assign even_numbits = ~odd_numbits;
endmodule
```

13.5.2 时序逻辑电路设计实例

13.6 状态机的置位与复位

13.6.1 状态机的异步置位与复位

13.6.2 状态机的同步置位与复位

总结 1:

- 1、凡是 always 敏感列表中出现沿触发 (posedge 和 negedge), 内部使用非阻塞赋值 (即<=)。
- 2、凡是 always 敏感列表中出现电平触发, 内部使用非阻塞赋值 (即<=)。
- 3、always 内部不能混合使用阻塞赋值与非阻塞赋值。

总结 2:

- 1、状态机: 状态多, 则状态转移简单, 状态少, 则状态转移复杂。
- 2、