

# 明德扬科技教育有限公司

## verilog 快速掌握练习 2 答案

官 网: [www.mdy-edu.com](http://www.mdy-edu.com)

淘 宝: [mdy-edu.taobao.com](http://mdy-edu.taobao.com)

QQ 群: 97925396

QQ 咨询: 158063679

## 一、quartus 常见错误

yinyue.v

Quartus 运行之后会出现以下错误,每改完一句都要编译运行一下 问题要一个一个解决

1. Error (10170): Verilog HDL syntax error at yinyue.v(20) near text "wire"; expecting an identifier ("wire" is a reserved keyword)

第 20 行 我们之前说过,在 `always` 里面的信号要用 `reg` 数据类型,而且,在时序逻辑中要用 `<=` 进行赋值,所以修改 `wire` 型为 `reg` 型,把 `always` 里的赋值都改为 `<=`,注意每句结束之后都要有一个 `;`,而不是其他符号,这样这个问题就解决了。

2. Error (10170): Verilog HDL syntax error at yinyue.v(33) near text "<="; expecting "="

第 33 行 这个错误已讲过,非时序要用 `=` 赋值。

3. Error (10170): Verilog HDL syntax error at yinyue.v(39) near text "=="

第 39 行,这个错误比较常见,就是在判断语句中,判断条件赋值应该用赋值符号 `==`。

4. Error (10137): Verilog HDL Procedural Assignment error at yinyue.v(41): object "beep" on left-hand side of assignment must have a variable data type

第 41 行,是说 `beep` 没有数据类型声明,要知道,在 `always` 过程块中被赋值的变量必须是 `reg` (寄存器型),用 `assign` 连续赋值的对象 必须定义成 `wire` (线型),所以出现矛盾,因为代码中,`beep` 既是在 `always` 块中,又用 `assign` 连续赋值,所以应该把 `assign` 去掉,同时在接下来的那个 `always` 中,在 `count <= count + 1'b1;`后面加上一句 `beep<=beep_r;`,别忘了定义 `beep` 的数据类型, `reg beep;`