Verilog规范

**代码设计规范**

1. 命名规则：区分大小写

所有的信号(signal)、变量(variable)及模块(module)的名字都用小写，常量(如参数 parameter)用大写字母。

1. 信号命名：信号的命名由缩写单词用下划线(‘\_’)连接而成，尽量使用缩写，名字不要太长。
2. 避免使用关键字。
3. 时钟信号使用前缀clk\_\*,使用有含义的缩写构成时钟信号，对于同一个时钟信号，在不同模块中保持名字一致。

复位信号时钟rst\_\*作为前缀，同上。

1. 低电平有效的信号添加尾缀\*\_n。
2. 名称中包含信号方向信息 输入\*\_i 输出 \*\_o
3. 用于仿真的文件名字与被测试模块名字对应，添加后缀\*\_tb.
4. 模块中用到的所有信号必须在信号声明部分进行声明，若做声明，将假定为为一位宽的wire变量。
5. 模块之间信号命名：表明数据方向，数据发出方在前，接收方在后，命名视信号的主要路径而定

示例：cpu\_mmu\_wr\_req CPU模块到mmu模块的写请求。

**注释及文本规范**

1 文件头：每个设计文件开头包含如下注释：公司名称 作者 创建时间 模块名称 实现工程 创建及修改记录

2 注释：使用//进行行尾注释 与代码之间添加一个空格 已分号结束

使用/\* \*/进行注释 /\*和\*/各占一行并且顶头。

注释与代码必须一致，修改程序后一定修改相应注释

注释应简介式点名程序的突出特征

3 端口的定义：每个功能模块按照输入 输出 双向的顺序定义，各个功能模块之间用空行最为间隔。

4 模块区域划分： .文件头

.时延单位和时延精度

.端口声明

.参数声明

.信号声明

.逻辑功能

各个模块之间用空行或者注释作为间隔

5 总线顺序：定义董MSB到LSB data[4:0]

6 空格使用：不同变量 变量与符号 变量与括号之间保留一个空格

关键词与其他任何字符串之间保留一个空格

逻辑运算符 算数运算符 比较运算符的两侧各留一个空格

单操作数运算符例外 直接位于操作数前

7 语句对齐：同一个层次的语句左端对齐

Always、Initial等语句块的begin跟在本行末尾，相应的end与之对齐，避免begin单独一行造成行数过多。

语句后的分号进行对齐

模块例化括号进行对齐

8 数字使用：书写数值是每隔4个bit用下划线(‘\_’) 隔开，8’h6666\_7777

在设计中不要直接使用数字，建议采用参数定义代替直接的数字

`define Data\_width 8

Wire [Data\_width-1:00\_] data\_i,

9 行长度： 不要太长，提高代码可读性及层次感

10 代码拷本 拷本代码后注意修改相应注释 避免产生错误指导 导致理解错误

**状态机设计**

1 状态定义 状态定义用parameter的方式，不使用`define的方式定义，因为编译时`define自动替换所有的宏，而parameter仅仅定义模块内部参数。

2 状态机健壮性 必须包括所有状态都处理，不能出现无法处理的状态导致失控，具备初始化(reset)状态和默认(default)状态。注意综合工具优化掉从非法状态跳回合法状态的电路，可以使用安全状态机选项禁止优化。

3 状态机风格：状态机推荐使用三段式，第一段产生sequential寄存器 第二段 产生下一状态的组合逻辑 第三段 时序逻辑输出