* uvm\_gen脚本路径：<https://192.168.100.106/svn/jin/verification/common/script/>
* 脚本使用命令示例：

svn co https://192.168.100.106/svn/jin/verification/function

(1)source prj\_setup.sh

(2)gen\_tb.py -tbname rce\_mu -infile mu\_wrapper\_ut.v

* 脚本使用的mu\_wrapper\_ut.v，示范示例在 verification/common/script/infile

mu\_wrapper\_ut.v详解如下：

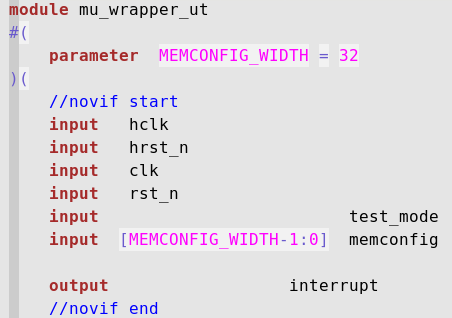
Module 后面的mu\_wrapper\_ut会解析，不能多余（符号 等

ports(input/output pins)，如下图，不能多余符号

主要加关键性的注释 (严格对应格式)

用于python脚本获取不同类型的接口信息，以下是示例：

1. 上图 novif表示该类型的接口作为驱动时，可在tb中的initial块中产生；作为输出时可以floating；



（2）下图verify表示该类型接口为uvc或者vip，需要uvm组件或vip集成来产生激励或monitor下来做自动检查；

对于标准VIP，脚本支持AHB/AXI/APB的集成 – 通过ahb#master#0类似信息来完成自动集成：

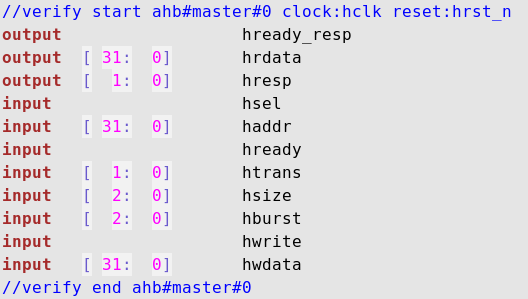
ahb作为VIP bus 名称；

master/slave指示是master还是slave；

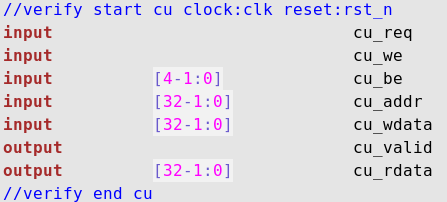
0 num表示master或slave的第几个接口数；

Clock后面的表示vip的接入时钟信号；

Reset后面的表示vip的接入复位信号；



（3）下图verify表示该类型接口为uvc，不是标准VIP，需要用户写agent，monitor，driver等；



* 在运行gen\_tb.py行过程中，会有2个提示信息，

内容是环境中是否需要regmodel，输入“yes” or “no”；

首先会提示在env中是否有regmodel，如果输入“yes”，regmodel会自动集成在env中；然后提示在base\_test中是否有regmodel，如果输入“yes”，regmodel会自动集成在base\_test中；建议均为“yes”或“no”；

在env中默认regmodel和AHB master[0] map连接，如下图所示：



根据设计中具体的adapter，此处需要手动更新上述的内容。

同时 reg目录下，有一个示例的adapter的文件；

如果有其他非common的总线想要通过regmodel来访问寄存器时，参考以下文档：

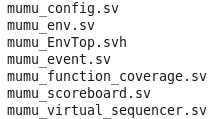


* 上述指令会生成rce\_mu的文件夹，下面的文件结构如下：



注意：如下表述中，为了使环境编译没有问题，\*表示该部分可能需要手动修改，\*\*表示该部分肯定需要手动修改。

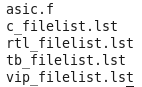
(1)\*\*env结构如下图所示：



其中mumu\_env.sv中可能需要手动修改：



(2)\*\*filelist结构如下图所示：



asic.f:目前的脚本不使用；

c\_filelist.lst: c reference model

rtl\_filelist.lst: RTL file

tb\_filelist.lst: tb file, 不修改

vip\_filelist.lst: vip相关，有些变量可能需要修改

(3)\*reference用于存放cmodel相关文件，格式按上述描述手动存放

(4)\*regmodel用于存放regmodel相关文件，把脚本生成的(rce\_mu)\_reg\_top.sv存放到该目录下

(5)run用于存放脚本，从common/下copy而来。

(6)\*sva用于存放assertion文件，如果需要的话

(7)\*\*tb用于存放tb\_top及时钟产生相关文件；

其中tb\_top需要手动更新：

1.删除DUT例化列表中最后一个信号后面的“,”；

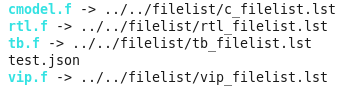
2.对于AHB master VIP接口，DUT的hsel/hready\_resp激励可能需要手动更新，比如hsel tie 1，hready\_resp和VIP的hready信号连接；

3.tb中时钟可能有多个，需要手动添加各个频率的时钟的产生，可通过时钟产生文件产生，也可在tb中直接定义产生时钟；

(8)tcl用于存放波形设置脚本文件，wave\_debug.tcl, wave.tcl，其他增加的文件，可以加在这里；

(9)testcase用于存放case

(10)testplan用于存放json及filelist link等，严格如下图



(11)uvc下为各个接口的uvc及vip的uvc，

vip的uvc可以在common/script/vip中copy而来

另外，建立了一个寄存器扫描的case -- AHB通过regmodel访问寄存器，case中包括regmodel的mirror、write等函数的使用，及regmodel自带seq（uvm\_reg\_hw\_reset\_seq/uvm\_reg\_bit\_bask\_seq）的使用；可以作为参考；case源路径如下图所示：

