testbench\_gen\_sw文件夹为软件目录，软件目录下已经放置有一个示例工程grey2bin。

该工程是一个将格雷码转化为二进制码的程序。

testbench\_generate\_rev.py为主程序，运行该程序

Enter top file name: 此处输入要验证工程顶层文件的文件名

Enter top file type (v\sv) : 此处输入要验证顶层文件的类型

Enter project director name : 可以输入相对路径或绝对路径 由于grey2bin工程在工程路径下的grey2bin文件夹下，只要输入./grey2bin 或grey2bin即可

Enable Frame Generater : 是否生成测试测试数据帧，此处接收True或False参数

Enable Frame Generate Length : 如果上一选项选择True此处需要添加生成长度

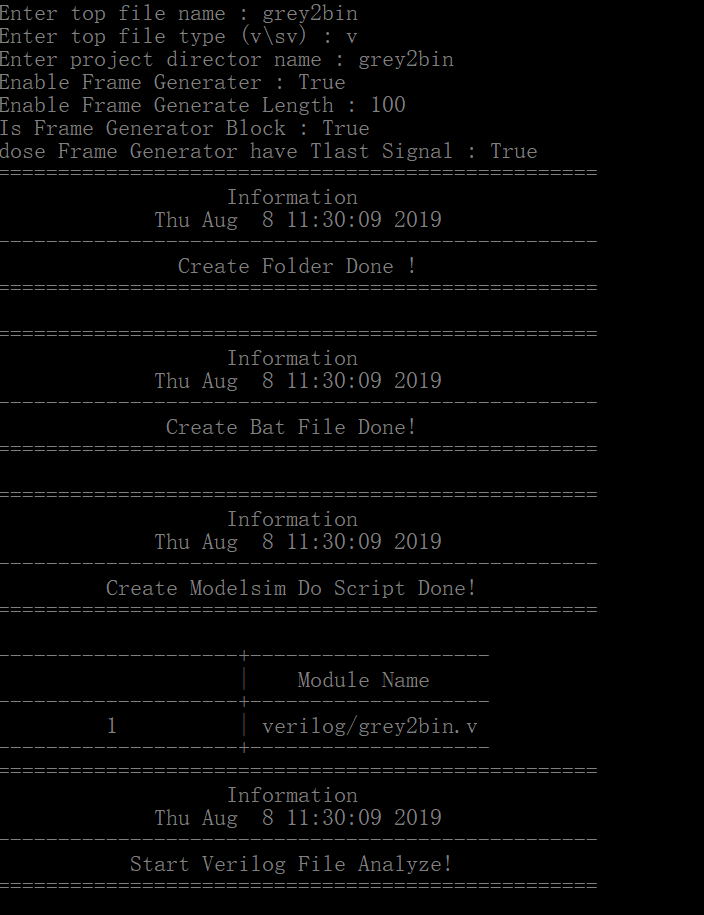
Is Frame Generator Block : 测试数据帧是否为阻塞，该选项设置为True或False，设置为True情况下只有在s\_axis\_tready被后端模块拉高后才输出数据，否则停留在当前数据等待tready信号变高

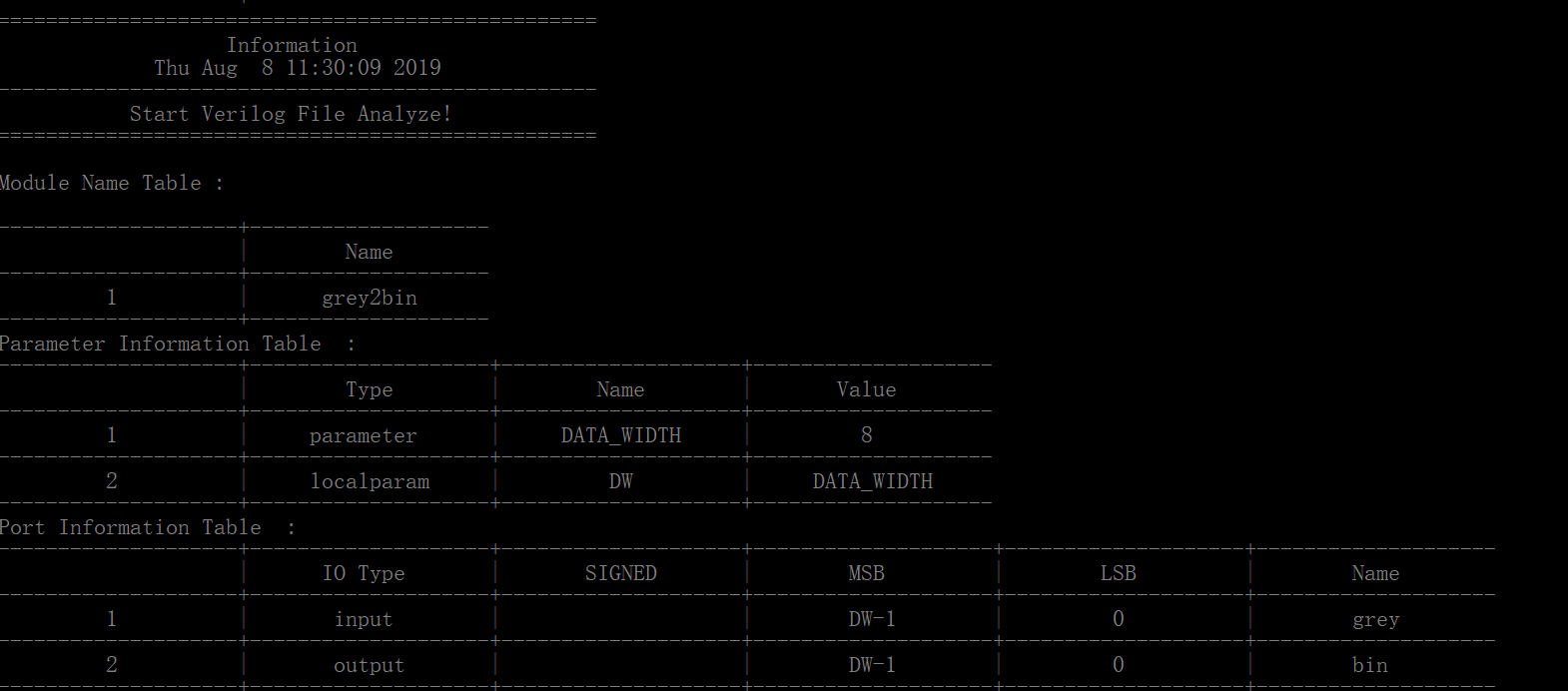
dose Frame Generator have Tlast Signal : 该流模式信号是否需要一个tlast终止信号，该信号会在当前帧的最后一个数据置位

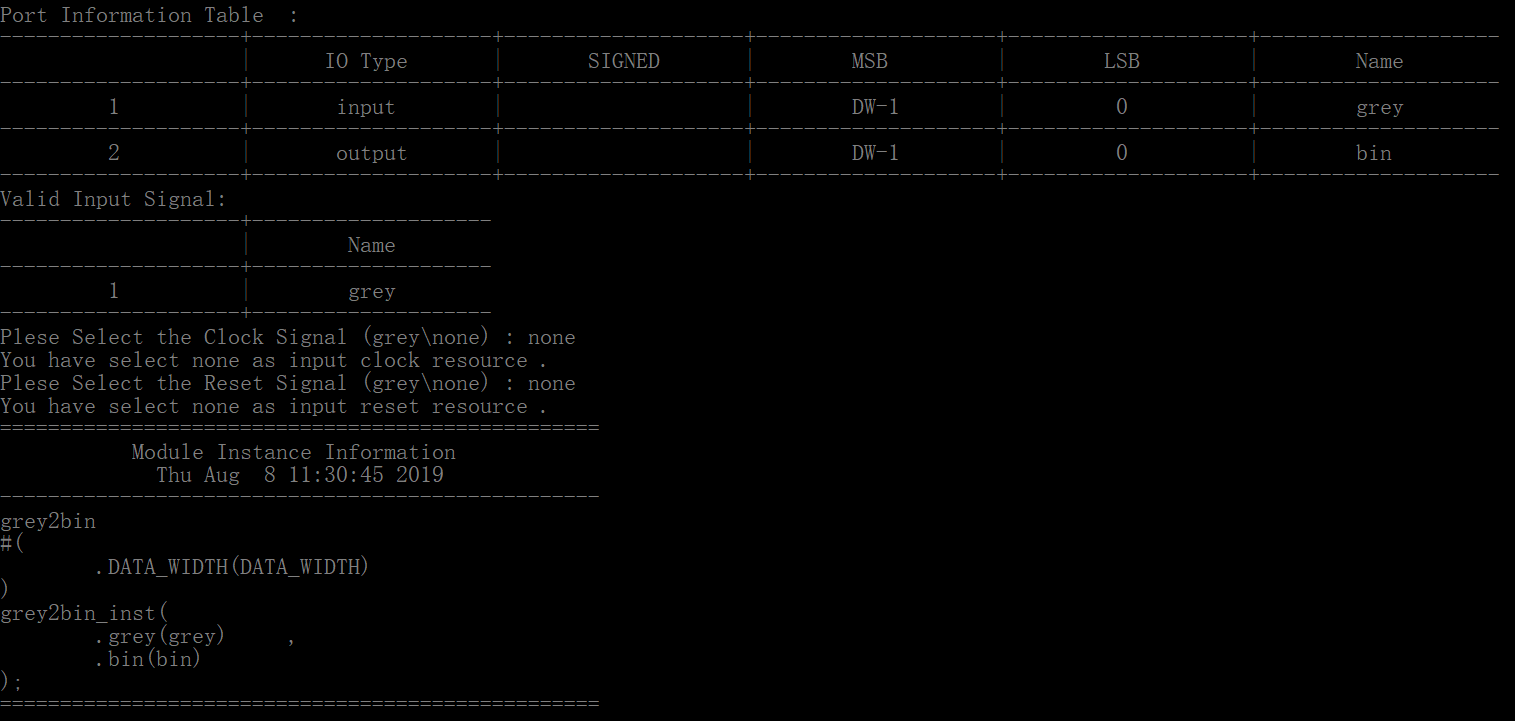
程序会自动搜索所有输入端口作为时钟或复位信号的备选端口，如果没有复位时钟端口直接设置为none：

|  |
| --- |
| Plese Select the Clock Signal (grey\none) : none  You have select none as input clock resource  Plese Select the Reset Signal (grey\none) : none  You have select none as input reset resource |

命令行输出如下：



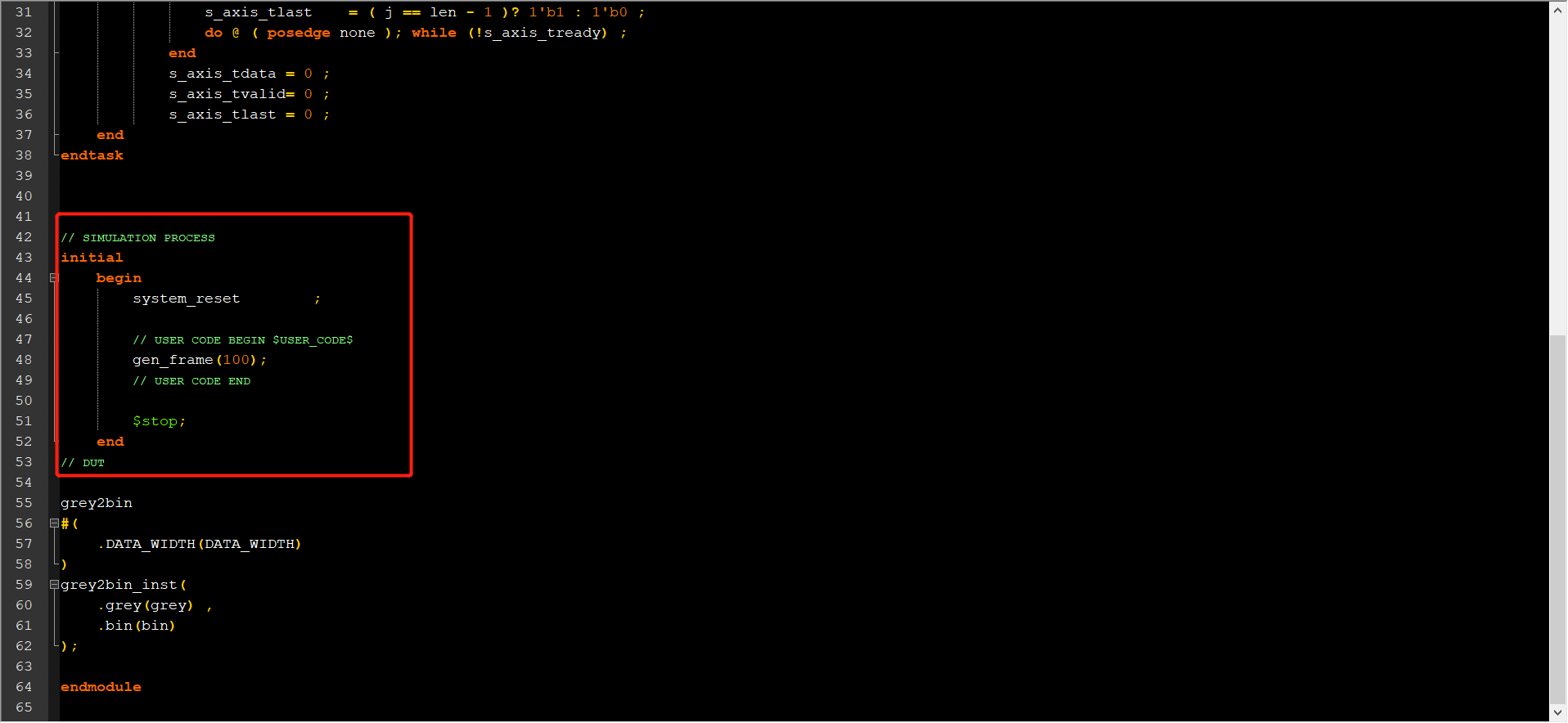




生成的自动化工程如下所示,其中data是仿真中可能用到的数据，proj是modelsim的工程路径，script时自动仿真脚本，verilog则是源文件目录，modelsim\_auto.bat则是自动仿真批处理文件，在将modelsim 下bin文件夹（或bin64）添加环境变量后双击即可进行自动仿真。



生成好的工程完成了基础的信号建立，DUT例化，如果要进行进一步仿真还要在代码对应位置添加用户自己的逻辑



找到自动生成的sv文件，在

// USER CODE BEGIN $USER\_CODE$

gen\_frame(100);

// USER CODE END

间添加其他的的用户逻辑