

# 实现 VScode 编译 Verilog 并查看波形图

## 一、需要的配置软件

iverilog

官方网址:

iverilog: [iverilog 下载地址](#)

## 二、使用步骤

### 1. 安装 iverilog

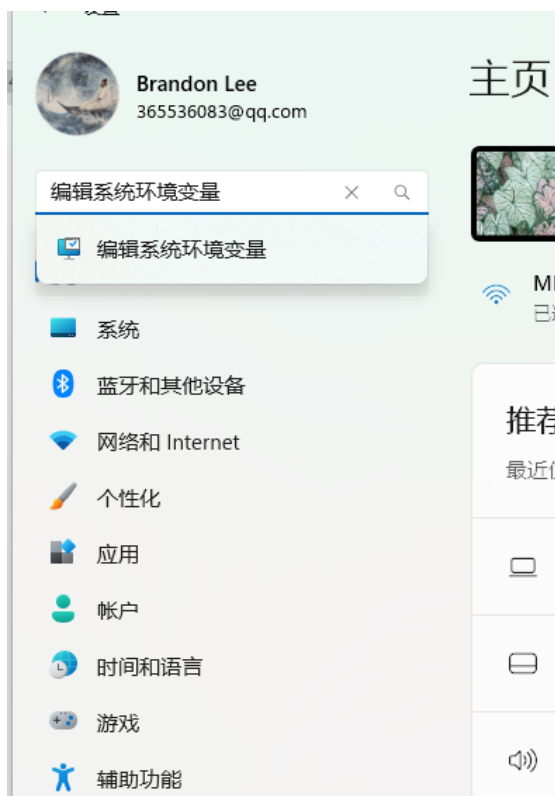
选择一个比较干净的位置安装 iverilog (路径最好不要包含中文, 防止奇奇怪怪的问题产生)。

点击.exe 文件, 进行安装。注意: 安装过程中最好将所有选项都勾选上, 防止后面出错。

### 2. 配置环境变量

将安装好的 iverilog 及其自带的 gtkwave 添加到系统环境变量中。

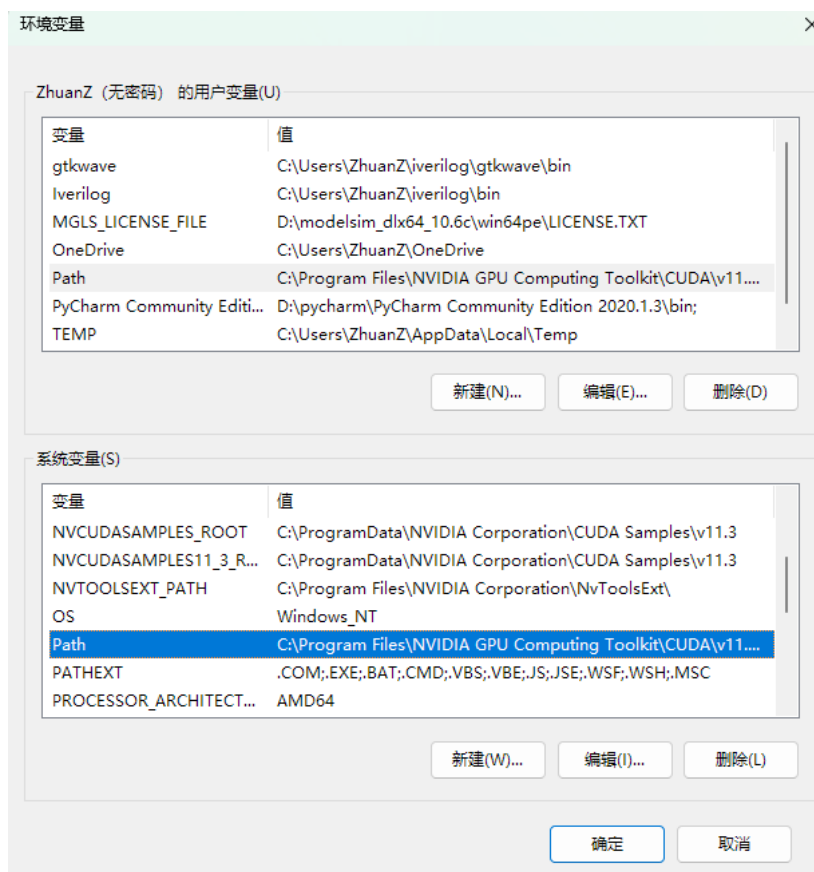
#### (1) 在设置中搜索编辑系统环境变量



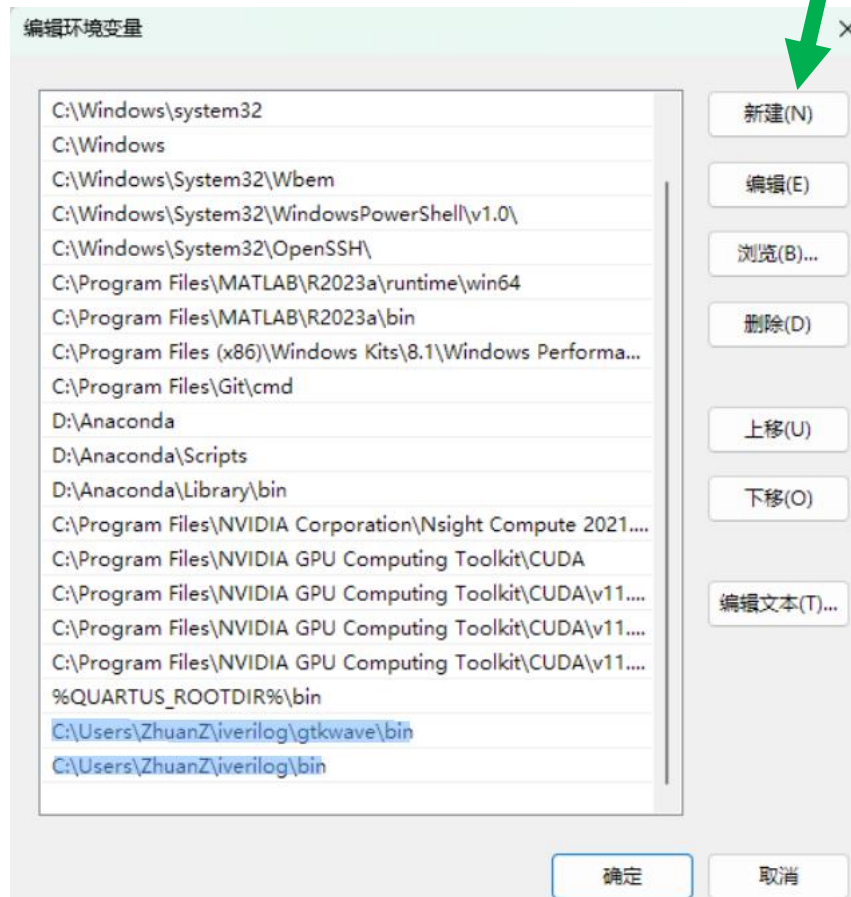
#### (2) 点击环境变量



(3) 双击系统变量当中的 path



- (4) 将文件夹 iverilog\bin 的绝对路径和 iverilog\gtkwave\bin 的绝对路径分别添加至系统环境变量 path 当中（也就是刚刚点 path 打开的这个界面，点击新建添加即可）



确认成功之后在运行框中分别输入 `iverilog -v` 和 `gtkwave -v`，如果有正常输出即为设置成功,类似于下图。

```
C:\Windows\system32\cmd X + -
C:\Users\ZhuanZ>iverilog -v
Icarus Verilog version 12.0 (devel) (s20150603-965-g55e06db6)

Copyright 1998-2020 Stephen Williams

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

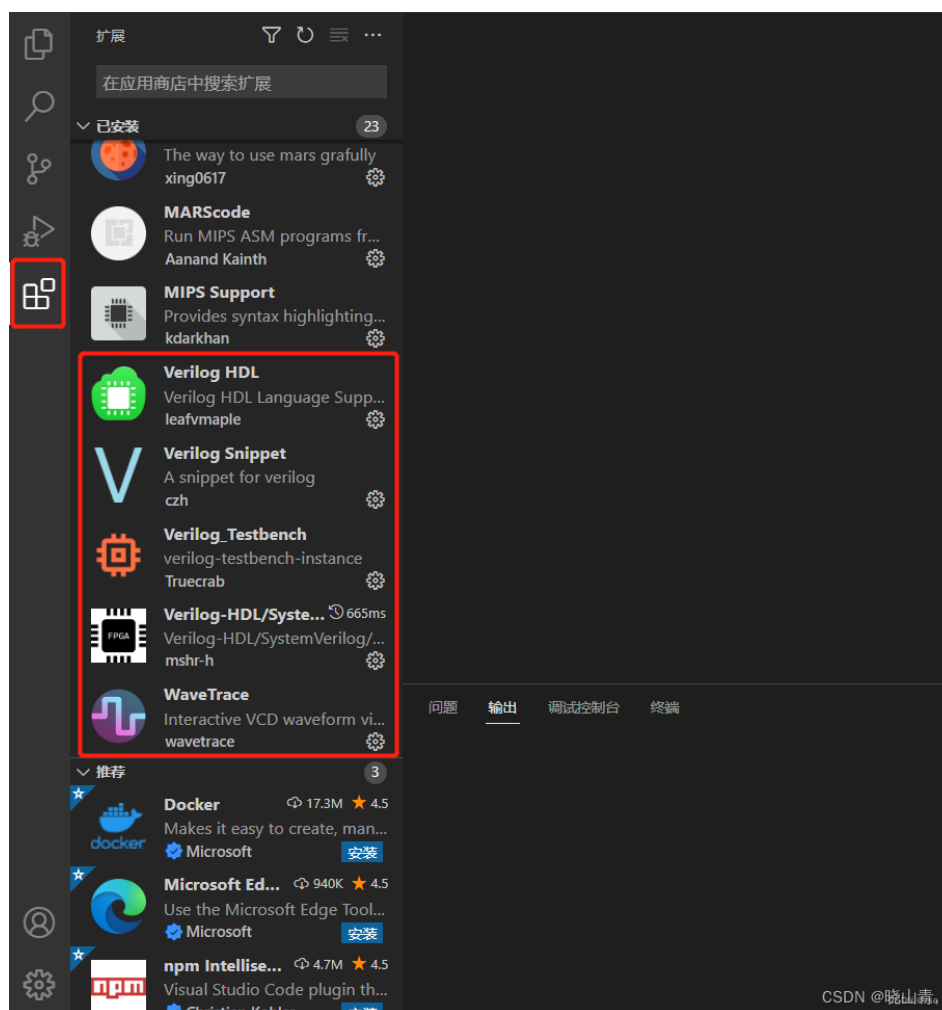
You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

iverilog: no source files.

Usage: iverilog [-EiSuvV] [-B base] [-c cmdfile] [-f cmdfile]
[-g1995|-g2001|-g2005|-g2005-sv|-g2009|-g2012] [--g<feature>]
[-D macro[=defn]] [-I includedir] [-L moduledir]
[-M [mode=]depfile] [-m module]
[-N file] [-o filename] [-p flag=value]
[-s topmodule] [-t target] [-T min|typ|max]
[-W class] [-y dir] [-Y suf] [-l file] source_file(s)
```

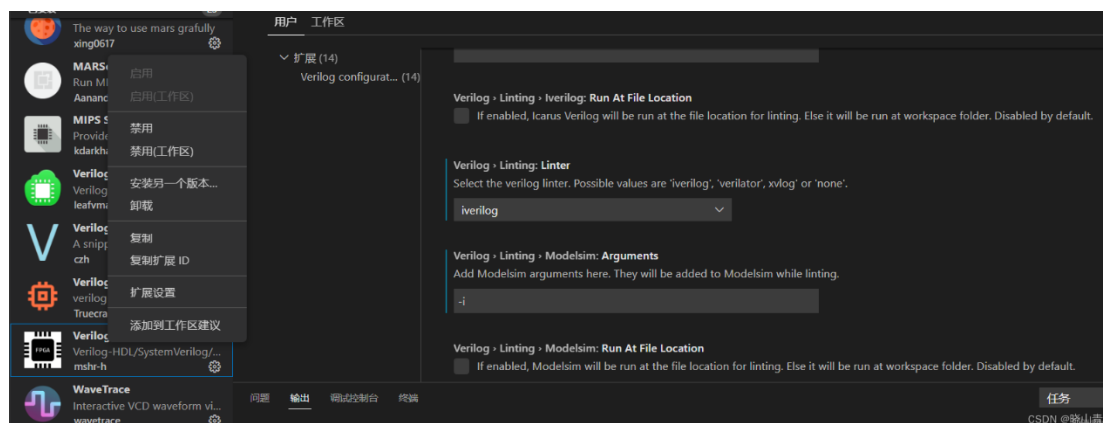
### 三.安装 VScode 插件

在插件商店里找到一下几个红框内插件并安装（插件图标为左侧四个小方块）



## 四.对插件进行配置

- 1.右键选择 Verilog-HDL/SystemVerilog/Bluespec SystemVerilog（即上图中黑白配色的插件，第四个）
- 2.点击扩展设置



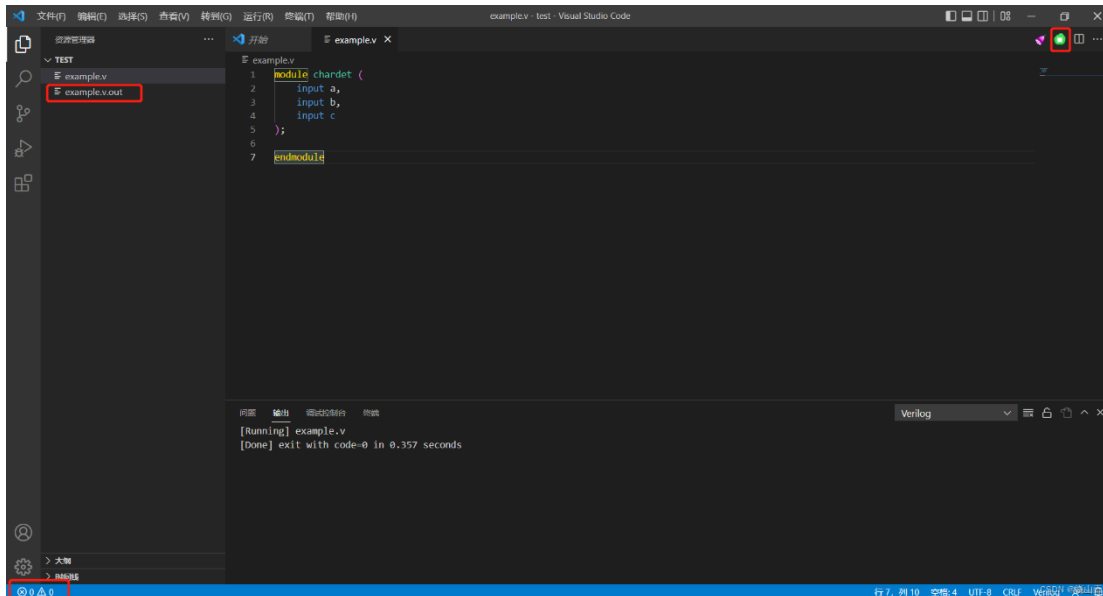
**Verilog > Linting: Linter: 选择 iverilog**

## Verilog › Linting › Modelsim: Arguments: 填入-i

重启之后，你就会发现你的 Vscode 支持 verilog 代码补全，代码高亮，报错，以及编译的功能。

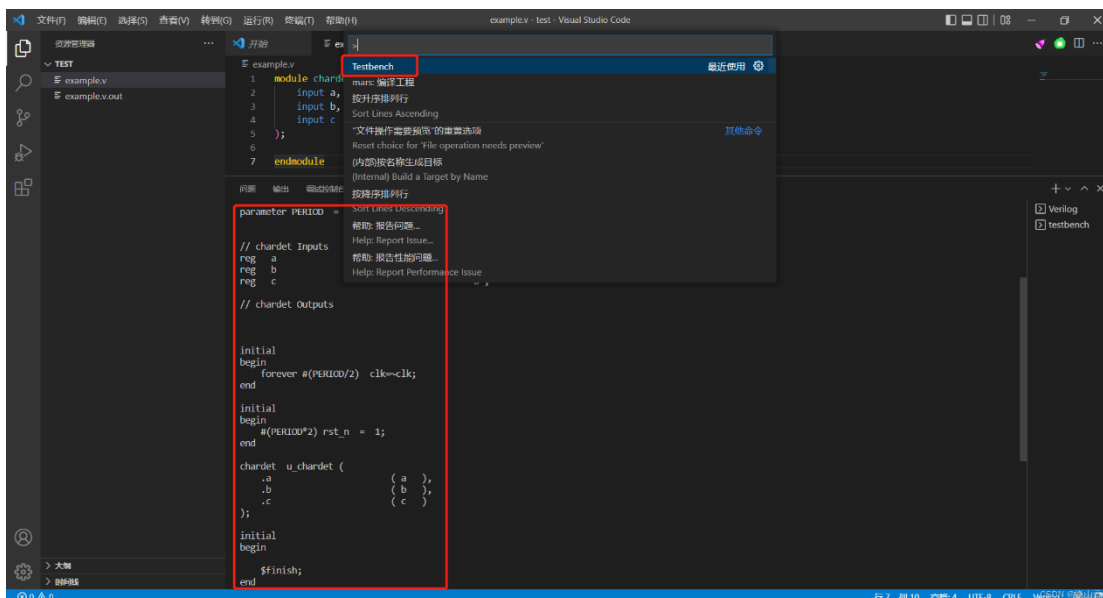
## 五.插件的使用

1.verilog 文件后缀为.v，点击右上角的绿色的编译按钮，若无错误提醒，即可编译成功，并在左侧的文件管理窗口看见.v.out 的文件



## 2.生成 Testbench 模板

在 VScode 使用快捷键 Ctrl+Shift+P 打开功能键，或者点一下顶部搜索栏，输入一个>,然后输入 testbench。运行，即可在终端中显示生成的模板文件,复制内容生成 tb 文件即可，但要注意的是文件后缀仍然是.v



## 3.对自动生成的 testbench 进行改编

根据要测试的文件的内容，改写 testbench。同时在 testbench 文件开头添加一下

## 代码以引用待测文件

```
1 | `include "[文件名.v]"
```

同时，在 initial 块中加入如下代码以生成波形文件：

```
1 | $dumpfile("波形文件名.vcd");
2 | $dumpvars;
```

其中\ \$dumpfile 指定波形文件名，\ \$dumpvars 表示所有定义的 wire 和 reg 都参与仿真。

再自行设置测试数据(在 dumpvars 和 finish 的中间添加状态代码)后点击运行，即可发现 tb 文件的.v.out 文件，同时还有 vcd 文件。

## 4.观察仿真波形

打开 vcd 文件，点击添加信号，即可查看波形,类似于下图。(下图为作者自己设计的矩阵 scatter-gather 乘法)

