AN45-0007 MH2457+LVGL模拟器开发指南

概览：
目的：
开发环境：
建议开发流程：
最简联调demo：
模拟器效果：
开发板效果：
工程架构：
模拟器环境搭建：
安装VSCode：
确保可以访问GitHub
安装GCC
安装SDL2
打开模拟器工程
模拟器常见问题：
附件：
LVGL相关资料：
参考文章：

概览：

目的：

由于LVGL工程过于庞大，直接在keil5MDK中编写界面代码并调试 耗时较长，开发效率较低。故建议使用电脑模拟器开发LVGL显示界面与界面跳转逻辑，当调试完成后再将demo下载入MH2457中运行并观察效果。

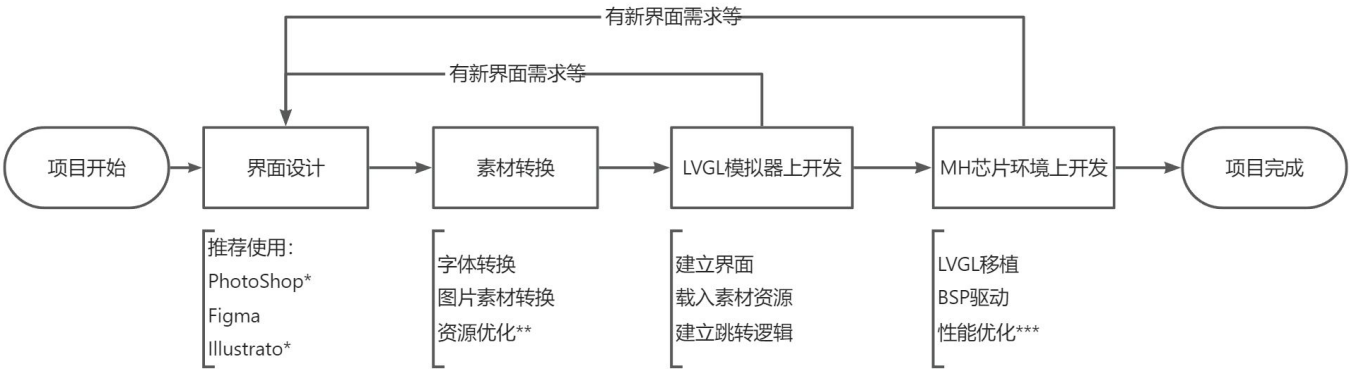
本文用于展示 模拟器和MH2457环境共用一套界面代码，可实现 **一次编程，到处编译** 的效果！

开发环境：

> VSCode环境相较GUI Guider等图形化软件：
>
> 优势是编程方便，适用于编写复杂跳转逻辑，缺点是需要对LVGL控件与常用接口具有一定熟练度。

- 软件：VSCode
- 模拟器编译运行：GCC + CMAKE + SDL2
- MH2457编译运行：KeiI5MDK（本文使用MDK 5.41.00，内置编译器V6.22）
- 硬件：MH2457 + 对应开发板 + 屏幕

建议开发流程：



注意！！！

1. 软件*：PhotoShop、Illustrator等软件需要商业购买，使用**破解版会有法律风险**。
2. 资源优化**：即根据需求合并部分图层、针对图片是否镂空合理使用RGB565或RGB565A8格式等。
3. 性能优化***：资源优化后，将部分资源导入SDRAM运行，同时优化屏幕驱动等，使屏幕刷新更流畅。

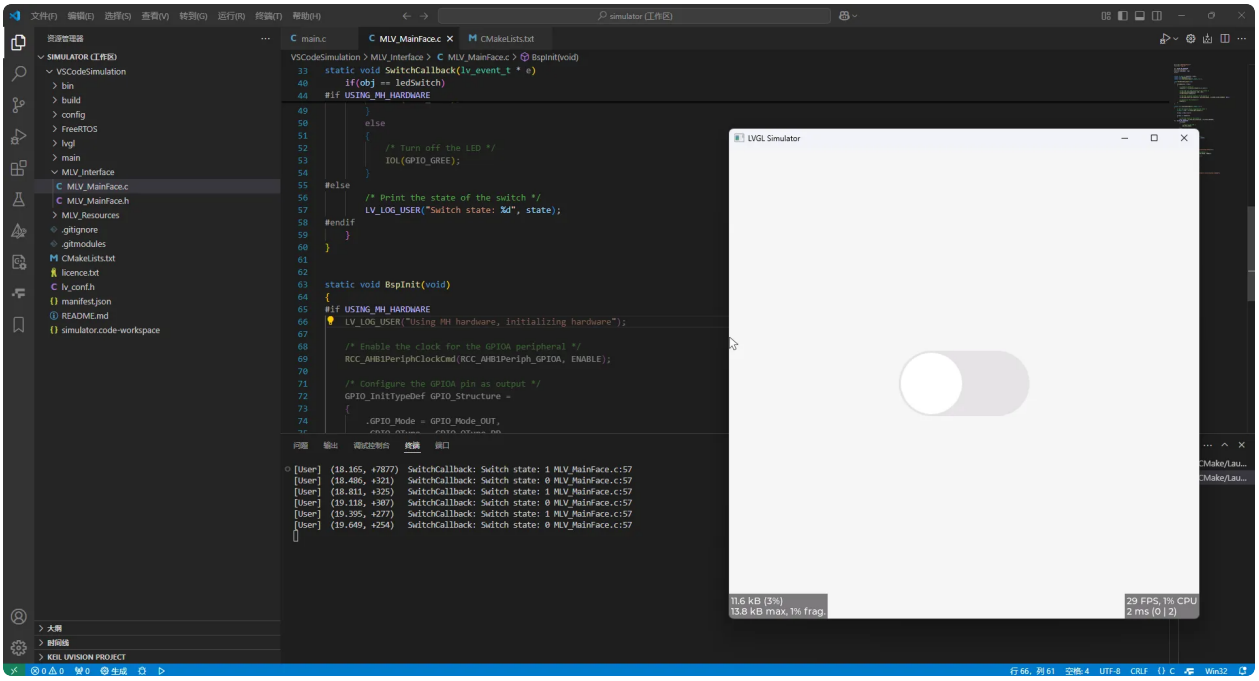
最简联调demo：

模拟器效果：

成功配置开发环境后，点击右下角编译运行，效果如下：

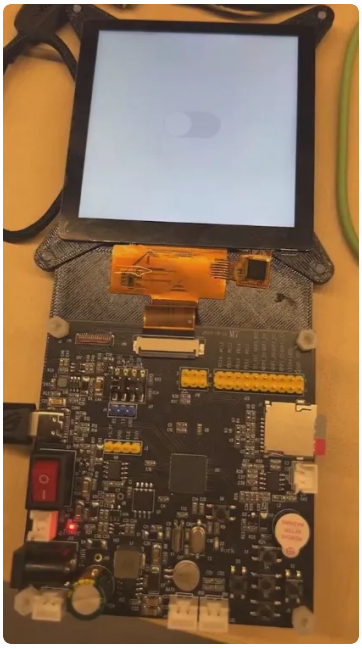
- 右侧白色正方形窗口为模拟器屏幕。
- 可以使用鼠标点击开关，实现开关的 **开启/关闭**。

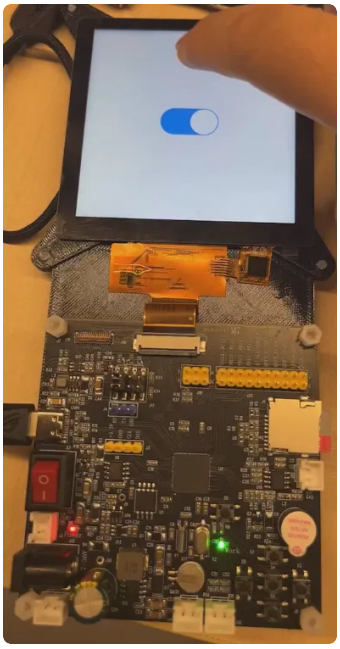
- 每次开关状态改变，都会在VSCode的终端中打印开关状态，可以看到状态是0 1交替的(0关1开)。



开发板效果：

- 屏幕显示与电脑模拟器环境一致。
- 手指点击触屏中心的开关，开关打开，同时开发板下方的绿色LED亮起。





工程架构：

公共文件部分中使用宏定义USING_MH_HARDWARE来区分 模拟器环境 和 硬件环境。



模拟器环境搭建：

安装VSCode：

https://code.visualstudio.com/Download

下载windows版本，一路next就行。

确保可以访问GitHub

推荐使用FastGithub.UI.exe

下载链接：

- 清华云盘：https://cloud.tsinghua.edu.cn/d/df482a15afb64dfeaff8/

下载完成后解压，双击 

运行即可，界面如下：

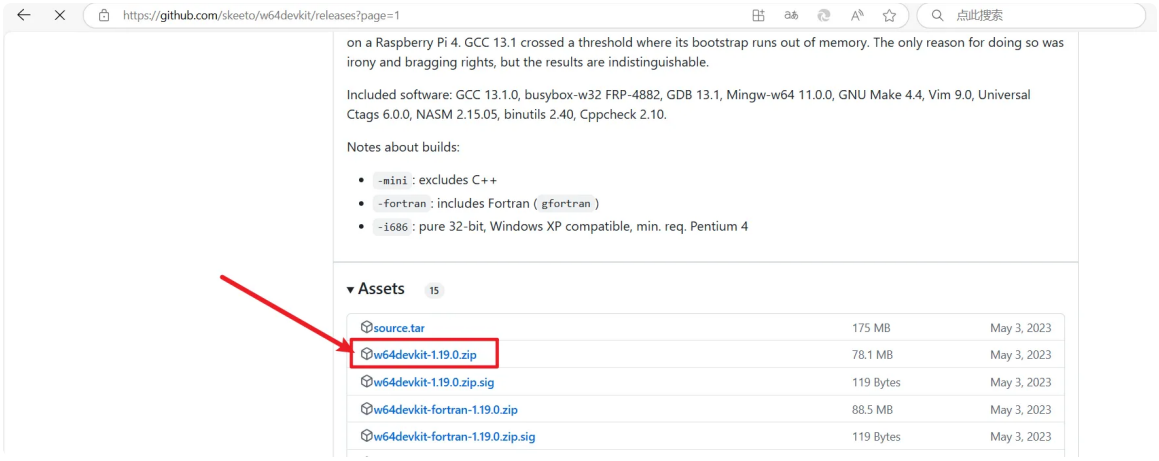


安装GCC

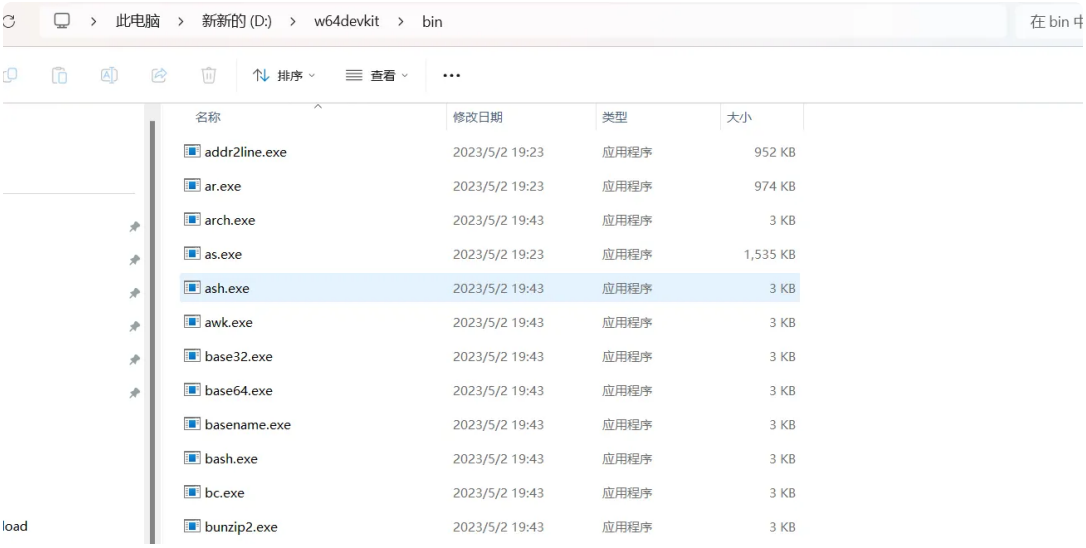
> 推荐使用mingw64安装，如果电脑已经安装GCC，这步可忽略。

打开网站：https://github.com/skeeto/w64devkit/releases?page=1

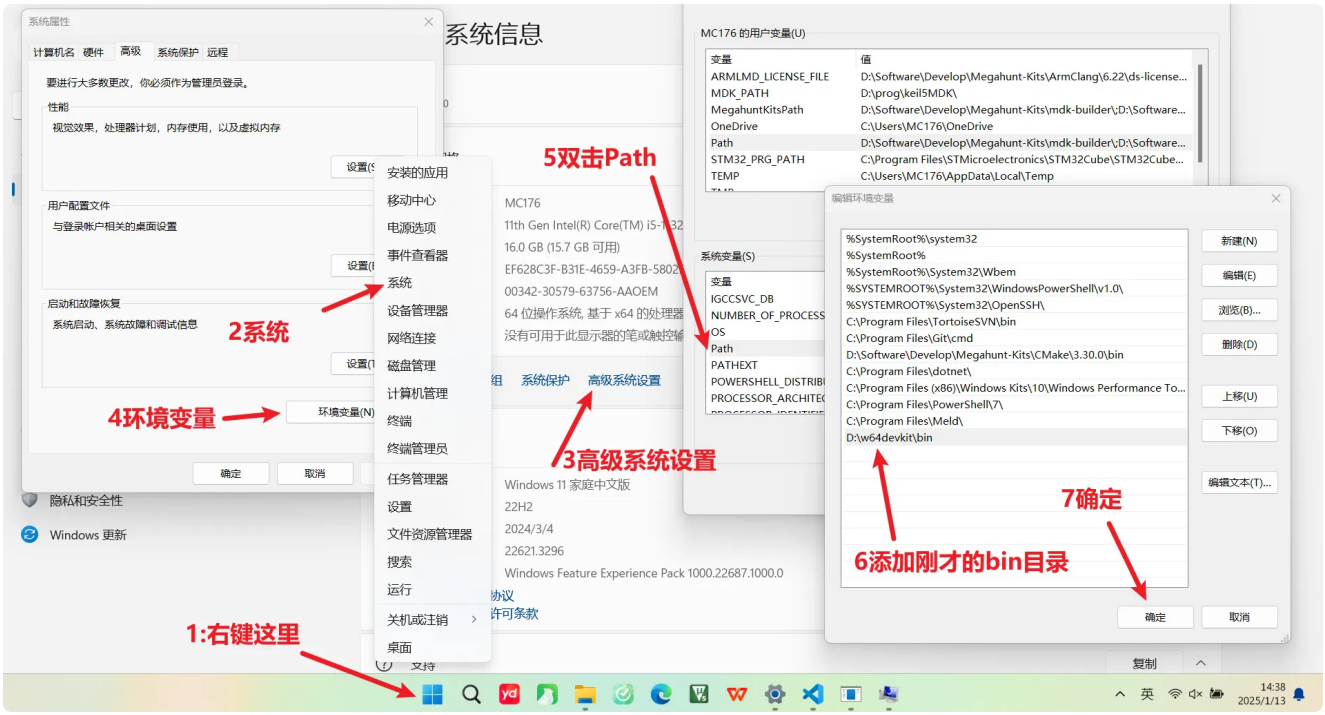
下载如下，w64devkit-1.19.0.zip



解压缩到D盘根目录，打开其中的bin目录，确保与下图类似：



将此bin目录添加到环境变量，并依次点击确定进行保存。



验证环境是否配置正确：

打开任意cmd窗口，输入如下：

```
gcc --version
```

如果返回如下，表示配置正常。

```
C:\WINDOWS\system32\cmd.
Microsoft Windows [版本 10.0.22621.3296]
(c) Microsoft Corporation。保留所有权利。

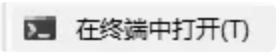
C:\Users\MC176>gcc --version
gcc (GCC) 13.1.0
Copyright (C) 2023 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

如果提示gcc不是内部或外部命令，请在确保上述步骤正确后，**重启电脑**，再次尝试gcc --version命令。

安装SDL2

> 这里推荐使用vcpkg安装，vcpkg为一款C/C++库管理工具
>
> 进行以下步骤的前提条件：
>
> - 电脑是Windows7及以上系统
> - 安装了Git

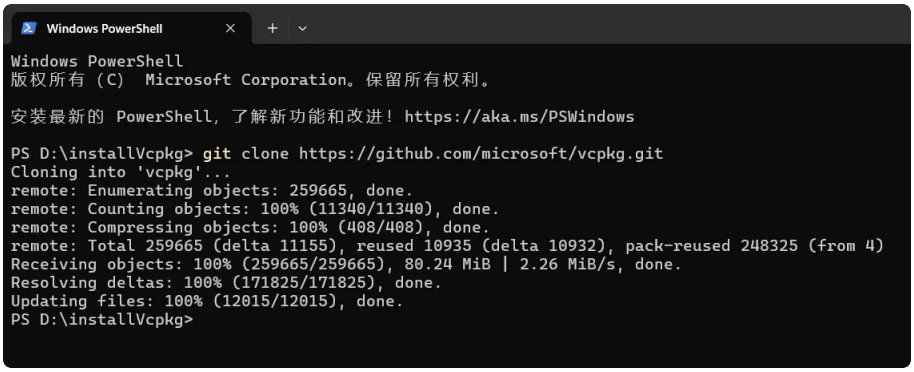
在D盘新建纯英文目录，建议名称installVcpkg，打开此目录

- 打开 PowerShell 或命令提示符（右键该目录任意空白处，选择 

）
- 克隆 `vcpkg` 仓库：

```
git clone https://github.com/microsoft/vcpkg.git
```

克隆完成效果如下：



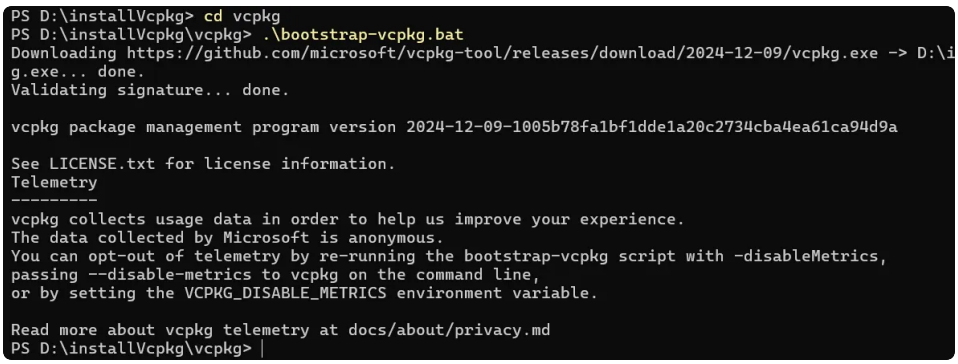
- 进入 `vcpkg` 目录：

```
cd vcpkg
```

- 运行 `bootstrap-vcpkg.bat` 脚本来构建 `vcpkg`：

```
.\bootstrap-vcpkg.bat
```

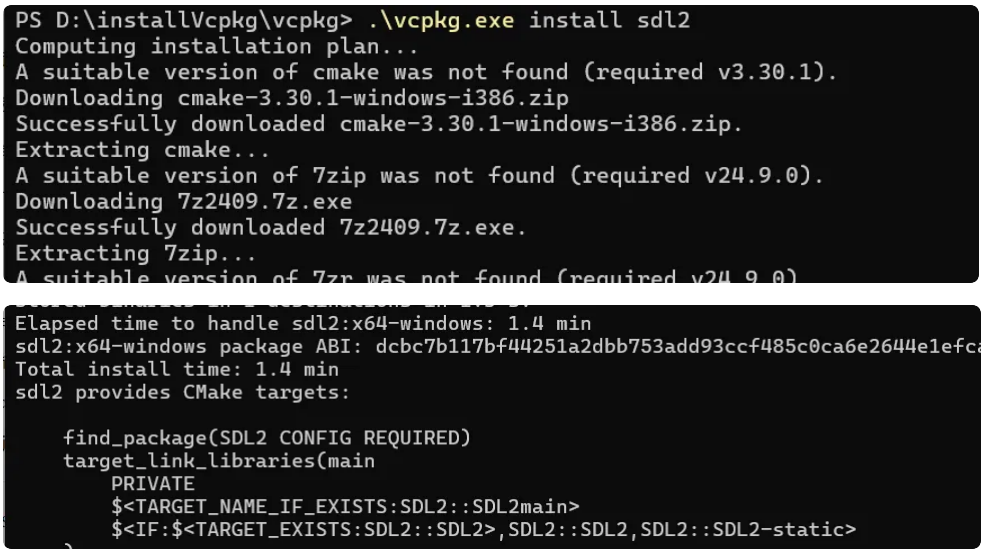
进入目录并运行脚本，如下：



- **安装 SDL2**：
- 运行以下命令来安装 SDL2：

> 这一步需要的时间可能较长，可以看FastGithub.UI工具中的下行流量，只要还在增加，就是正常的。
>
> 在这一步会自动安装并更新所有所需的开发工具（例如CMake、7Zip等）

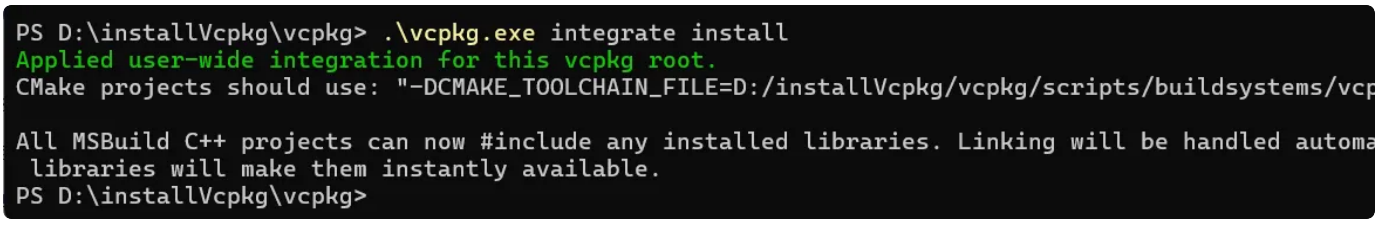
```
.\vcpkg.exe install sdl2
```



- **配置 `vcpkg` 与 Visual Studio（如果你使用的是 VS）：**
- 运行以下命令，将 `vcpkg` 与 Visual Studio 集成：

```
.\vcpkg.exe integrate install
```

这会自动将 SDL2 的头文件和库路径添加到你的 Visual Studio 项目中。



- **将vcpkg集成到系统变量，以便CMake自动识别：**
- 运行以下命令，注意！！如果你的vcpkg安装目录不同，下面的installVcpkg需要替换成你的vcpkg安装目录

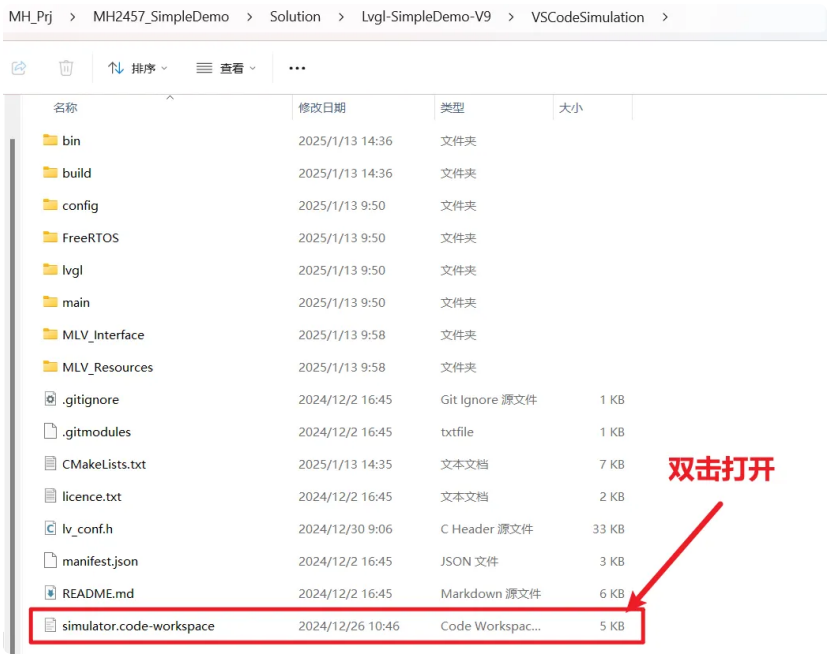
```
setx CMAKE_TOOLCHAIN_FILE "D:/installVcpkg/vcpkg/scripts/buildsystems/vcpkg.cmake" /M
```

```
PS D:\installVcpkg\vcpkg> setx CMAKE_TOOLCHAIN_FILE "D:/installVcpkg/vcpkg/scripts/buildsystems/vcpkg.cmake" /M
成功: 指定的值已得到保存。
```

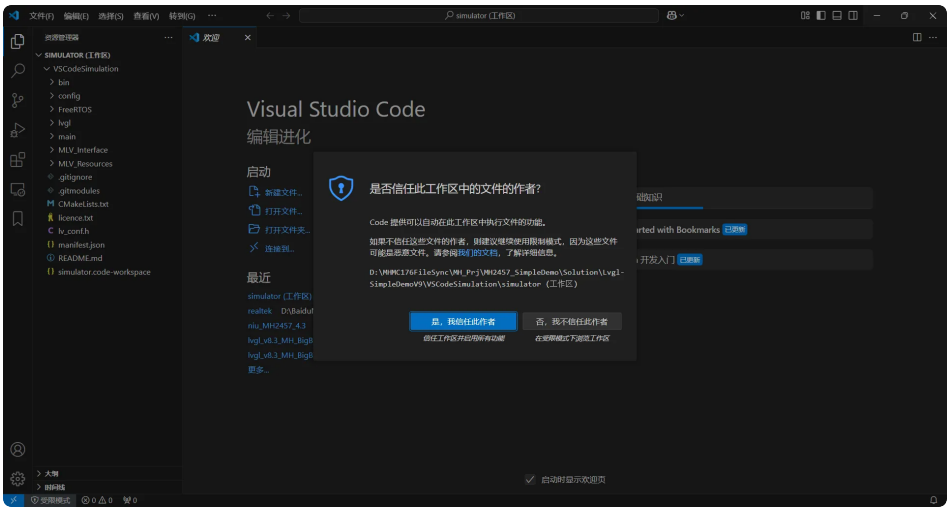
- **使用 SDL2**：
- 在你的 Visual Studio 项目中，`vcpkg` 会自动处理 SDL2 的包含路径和链接库。

打开模拟器工程

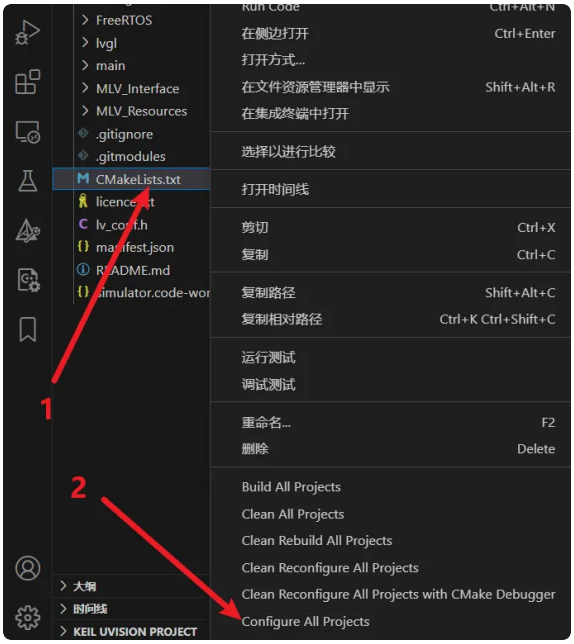
打开工程目录下的VSCodeSimulation文件夹，找到simulator.code-workspace文件，双击打开。



这里点击 我信任此作者

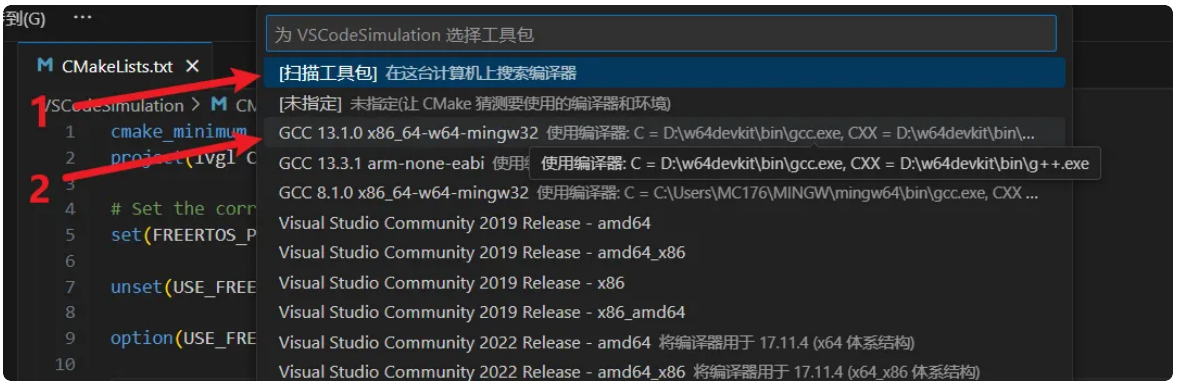


在右侧文件浏览器中，右键CMakeLists.txt，选择Configure All Projects



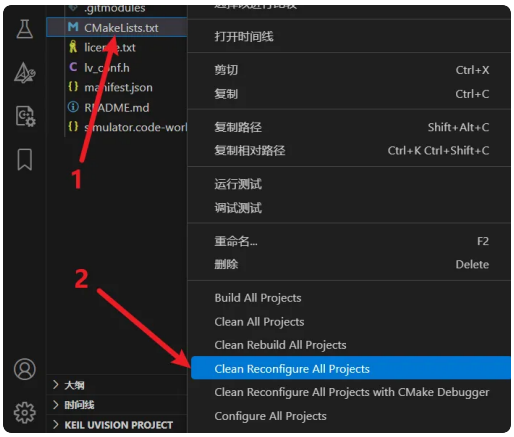
此时会在顶部弹出如下界面，

点击1处，扫码工具包，直到出现2，选择我们刚才安装的GCC（顺便核对一下路径和上文中安装GCC的路径一致）

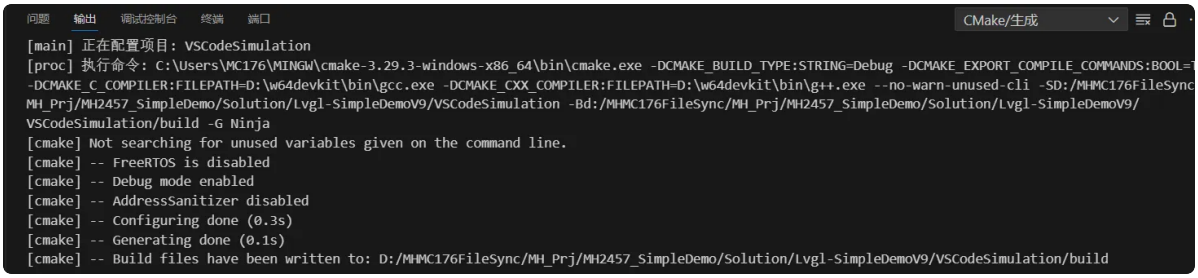


此时就可以开始准备编译了：

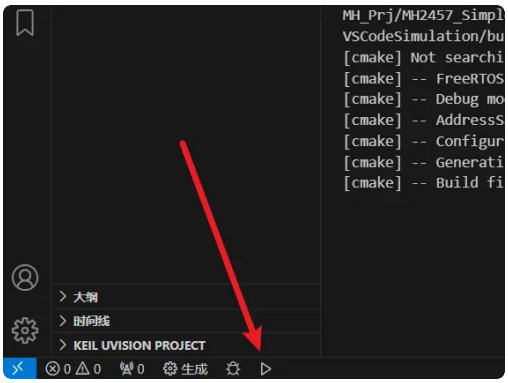
1. 右键CmakeLists.txt
2. 选择Clean Reconfigure All Projects，即清理并重新配置所有工程



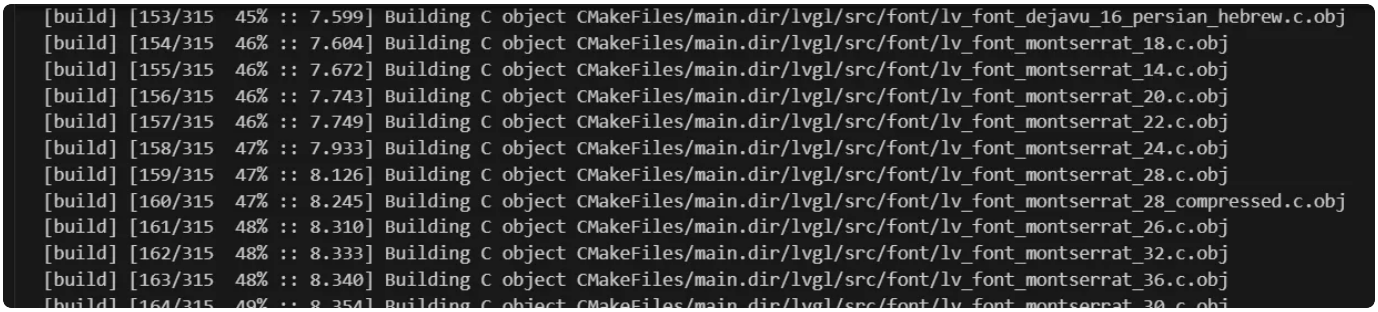
此时输出如下，表示配置完成。



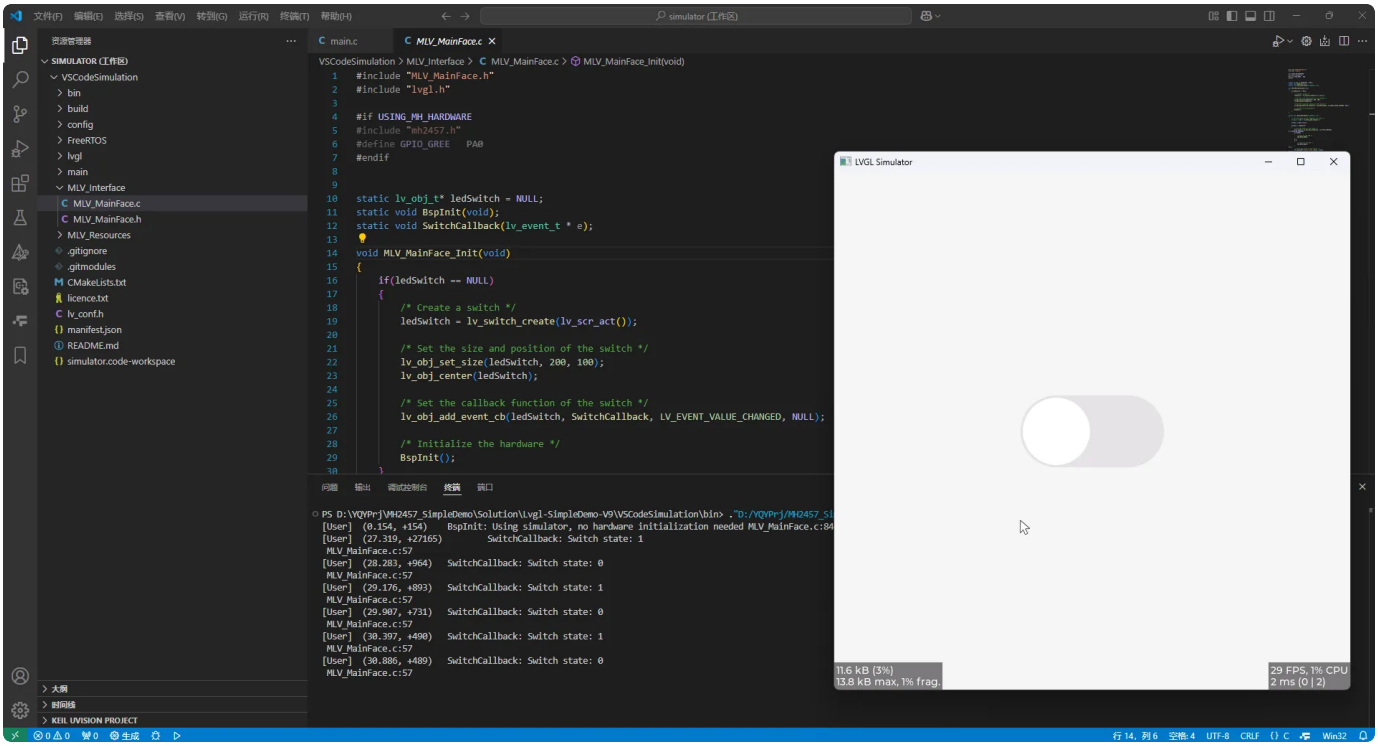
最后一步，点击编译运行：



等待系统编译：



完成，获得模拟器界面：



模拟器常见问题：

- 更换工程文件夹后编译报错：
  - 手动删除模拟器工程下的build文件夹，之后右键CMakeLists.txt，选择清理所有项目并重新配置。
- 如何添加新文件?
  - 在工程文件夹下MLV_Interface文件夹添加新的.c/.h文件后，右键CMakeLists.txt，选择清理所有项目并重新配置。
- 如何添加新文件夹?
  - 需要修改CMakeLists.txt文件，需要有CMake基础知识。可参考CMakeLists.txt文件中86~87行变量MLV_INTERFACE_SOURCES和变量MLV_INTERFACE_INCLUDE的写法。
  - 另附：CMake相关学习教程：https://www.runoob.com/cmake/cmake-tutorial.html
- 遇到了其它问题?
  - 模拟器配置涉及到的都是软件开发的基础操作，网上通常都有大量资料。
  - 推荐询问GPT软件，如ChatGPT，或国内的质谱清言等。

附件：

LVGL相关资料：

官网（内含图片转换和字体转换工具）

> https://lvgl.io/

LVGL API参考手册，推荐百问网：

> https://lvgl.100ask.net/9.2/

参考文章：

windows安装GCC：

> https://blog.csdn.net/K346K346/article/details/130870292?ops_request_misc=%257B%2522request%255Fid%2522%253A%25224b11c7a955f82bd71f1e3fd67c3243b4%2522%252C%2522scm%2522%253A%252220140713.130102334..%2522%257D&request_id=4b11c7a955f82bd71f1e3fd67c3243b4&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2~all~top_click~default-1-130870292-null-null.142^v101^pc_search_result_base9&utm_term=windows%20gcc%E5%AE%89%E8%A3%85&spm=1018.2226.3001.4187

vcpkg工具安装，基本使用：

> https://blog.csdn.net/qq_41023270/article/details/109274433