总目录

*开发环境配置总结

内容总结

前置知识

1. **电脑怎么运行程序以及什么是指令集**

   参考资料

   【CPU 的 x86 架构和 ARM 架构有啥区别？指令集又是什么？】
   https://www.bilibili.com/video/BV1xs411V7od/

2. **什么是编译、链接**

   参考资料

   【「Coding Master」第 49 话 庖丁解牛，谭老师详解编译过程！】
   https://www.bilibili.com/video/BV1bv411u7ua/

3. **GNU 标准以及 llvm 标准**

   GNU 和 LLVM 是不同的编译标准，有不同的编译流程。

   一般 gcc g++ gdb 遵循 gnu 标准，而 clang cl lldb 等遵循 llvm 标准.

   参考资料

   【【熟肉】100 秒介绍 LLVM】

   https://www.bilibili.com/video/BV1RF411K7F5/

4. **什么是环境变量**

   环境变量（environment varieable）是操作系统在环境中存储的一些变量，这些变量的值一般是一些需要在任何目录位置都可以访问到的，其中常用的环境变量为 PATH，它记录了一系列绝对目录位置，每当一个程序想要寻找 dll 动态链接库和其他程序时，它们会优先在程序所在目录或者 PATH 中的目录寻找。

   注意，gdb 调试器可能对含有汉字和其他字符的路径判断错误，请尽可能让 PATH 的内容里面没有 ASCII 标准之外的字符

开发环境所需的软件

软件名称（紫色的是可能需要登陆或付费的商业软件）	功用	相关信息	安装方法 windows 操作系统
Stlink drive 驱动程序	让电脑正确识别 stlink  stlink 是协助我们调试单片机上程序的硬件	下载地址 www.st.com/en/development-tools/stsw-link009.html	运行安装程序
Clion	jetbrian 公司开发的 集成开发环境，是一个商业软件。提供自动补全、代码高亮、组织程序文件等功能。	官网 www.jetbrains.com/clion/	1 运行安装程序 2 手动配置各种工具

*Cmake *Clion 安装时一般会被自动安装*	开源软件，用来控制编译的过程，简化复杂的编译生成过程。生成程序的规则存储在 CMakeLists.txt 里。 一般的项目不只有一个源码文件，Cmake 会根据这个规则指挥编译器，让其按照规则把各个源码编译、链接。  Cmake 工作的简单流程如下 根据 CMakeLists.txt 生成 makefile，然后 Make 根据 makefile 组织编译器进行编译。	多文件工程组织	和 clion 一同被自动安装
MinGW x86\i386 指令集 gnu 标准 (不是很严谨地，32 位 x86 也称作 i386，64 位 x86 也称作 x86_64)	一个 GNU 标准的编译工具集(亦译作工具链)，可以编译 c 程序到 x86 指令集的可执行文件(这个编译器的指令集由开发环境所在的计算机决定，一般是 x86)。 由于要使用 gdb 调试器，所以需要使用 gnu 标准的编译器。	mingw64 项目官网 www.mingw-w64.org/ 适合的 windows 发行版 github.com/niXman/mingw-builds-binaries/releases	1 下载或安装符合要求的版本 2 手动解压或使用安装程序安装 3 手动将其配置到环境变量
Gnuarm-none-eabi-gcc arm 指令集 gnu 标准	一个 GNU 标准的工具集(亦译作工具链)，可以编译 C 程序到 arm 架构的指令集。功能与 mingw 类似，但是生成的程序的指令集不一样。		1 运行安装程序或手动解压 2 手动将其配置到环境变量 3 手动将其配置到 Clion 的工具链里
OpenOCD	On Chip Debugger OpenOCD provides on-chip programming and debugging support with a layered architecture of JTAG interface and TAP support including: - (X)SVF playback to facilitate automated boundary scan and FPGA/CPLD programming; - debug target support (e.g. ARM, MIPS): single-	openocd.org	1 手动解压 2 手动将其配置到环境变量 3 手动将其配置到 Clion 设置的嵌入式开发 Embedded development 的选项

	stepping, breakpoints/watchpoints, gprof profiling, etc; - flash chip drivers (e.g. CFI, NAND, internal flash); - embedded TCL interpreter for easy scripting. Several network interfaces are available for interacting with OpenOCD: telnet, TCL, and GDB. The GDB server enables OpenOCD to function as a "remote target" for source- level debugging of embedded systems using the GNU GDB program (and the others who talk GDB protocol, e.g. IDA Pro). 复制自 openOCD 的 readme 文件		
STM32CubeMX	用来配置、生成对应单片机的项目文件模板和各种配置文件。		1 运行安装程序，登陆意法半导体账户。 2 手动将其配置到 Clion 设置的嵌入式开发 Embedded development 的选项

开发过程中各种常见名称、类别的作用

文件、目录名称	特殊
.cpp/.hpp/*.c/*.h 文件	C 语言代码源文件。
*.cfg	configuration（配置文件）用来记录一些配置信息，不过内容格式依据情况而定。
*.s 文件	汇编代码。
CMakeLists.txt	cmake 项目文件。
*.ioc 文件	stm32CubeMX 配置文件，记录项目的各种设置，stm32CubeMX 就是根据这个文件生成的模板。
*.log 文件	log(日志)，用来记录程序、项目的运行情况。如果出错，查阅 log 文件可以帮助我们发现错误原因。
src 文件夹	sourece(源) 用来存储源码的文件夹。
bin 文件夹	binary(二进制) 用来存储可执行文件、二进制

	文件的文件夹。
inc 文件夹	include(头文件)用来存储一些头文件、接口的文件夹。

大家感兴趣也可以网上搜索相关内容

项目 Create, Build & Launch

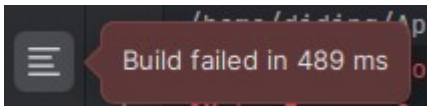
名称	过程	具体过程
Create	**项目的创建**	用户使用 Stm32CubeMX 配置工程的配置文件(*.ioc) Stm32CubeMX 生成各种库、预制代码、CmakeLists 等
Build	**项目的编译生成** 从项目源码生成 *.elf 二进制文件	使用 Clion 加载项目。 Clion 会尝试使用 Cmake，编译、生成项目(*.elf 文件).
Launch	**烧录、使用 OCD 调试** 根据*.cfg 配置文件，把.elf 烧录到单片机后，文件调试程序	把程序烧录到单片机里面。 OpenOcd 调用 GDB 调试器，借助 stlink 调试单片机里的程序

*经典程序错误汇总

错误之--无法编译

错误特征

1.无法构建(Build Fail)



2.有关 Cmake 的错误(Cmake Error)

编译器找不到

CMake 找不到对应的编译器，可能是编译器没有被正确安装

```
essages    Build
====================[ Build | holaEmdiona.elf | Debug ]=========================
/home/diding/Applications/clion-2024.2.2/bin/cmake/linux/x64/bin/cmake --build /home/diding/CLionProjects/holaEmdiona/c
-- The C compiler identification is unknown
CMake Error at CMakeLists.txt:18 (project): Debug
  The CMAKE_C_COMPILER:

    -whatever-gcc

  is not a full path and was not found in the PATH.

  Tell CMake where to find the compiler by setting either the environment
  variable "CC" or the CMake cache entry CMAKE_C_COMPILER to the full path to
  the compiler, or to the compiler name if it is in the PATH.


-- Configuring incomplete, errors occurred!
gmake: *** [Makefile:701: cmake_check_build_system] Error 1
```

因为 CMakeLists 里面指定需要这些编译器

解决这个问题需要检查 arm 架构的编译器（arm-none-eabi-gcc）和 x86 架构的工具集(MinGW)是否安装好，配置好环境变量。

注意：对于一些操作系统，环境变量变更需要重启后才生效。

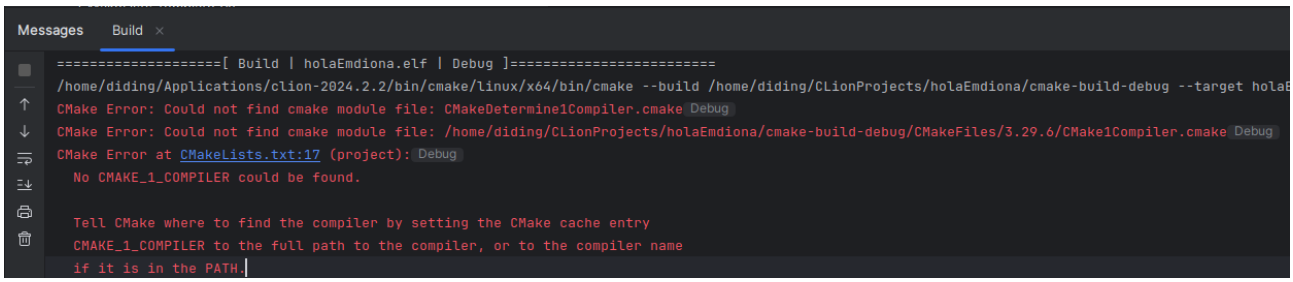
```
# specify cross-compilers and tools

set(CMAKE_C_COMPILER arm-none-eabi-gcc)
set(CMAKE_CXX_COMPILER arm-none-eabi-g++)
set(CMAKE_ASM_COMPILER  arm-none-eabi-gcc)
set(CMAKE_AR arm-none-eabi-ar)
set(CMAKE_OBJCOPY arm-none-eabi-objcopy)
set(CMAKE_OBJDUMP arm-none-eabi-objdump)
set(SIZE arm-none-eabi-size)
set(CMAKE_TRY_COMPILE_TARGET_TYPE STATIC_LIBRARY)
# project settings
project(holaEmdiona C CXX ASM)
set(CMAKE_CXX_STANDARD 17)
set(CMAKE_C_STANDARD 11)
```

项目名称有空格

如下图所示，将项目命名为 “holaEmdiona 1”时，stm32CubeMX 自动生成的 cmakelists 里面会有这一行。

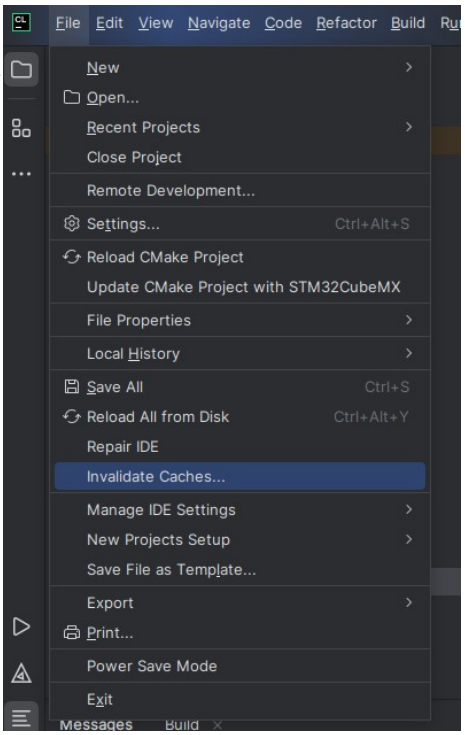
```
# project settings
project(holaEmdiona 1 C CXX ASM)
set(CMAKE_CXX_STANDARD 17)
set(CMAKE_C_STANDARD 11)

#Uncomment for hardware floating point
#add_compile_definitions(ARM_MATH_CM4;ARM_MATH_MATRIX_CHECK;ARM_MATH
```

Cmake 命令的参数以空格为分隔符号，而 project 命令是将第一个参数作为项目名而随后的若干参数作为项目使用的语言。这里把“1” 当作了一种编程语言。故会报错



错误的意思就是找不到名字叫做 1 的编程语言。

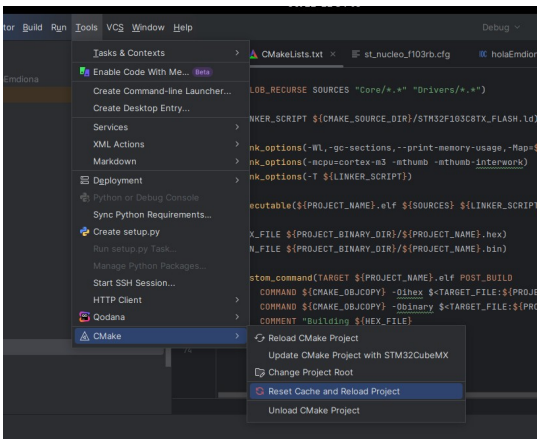
错误之--Clion 对变更的迟滞



当 Clion 无法对已经有的变更做出正确响应，可以通过如下方法刷新开发的工作空间

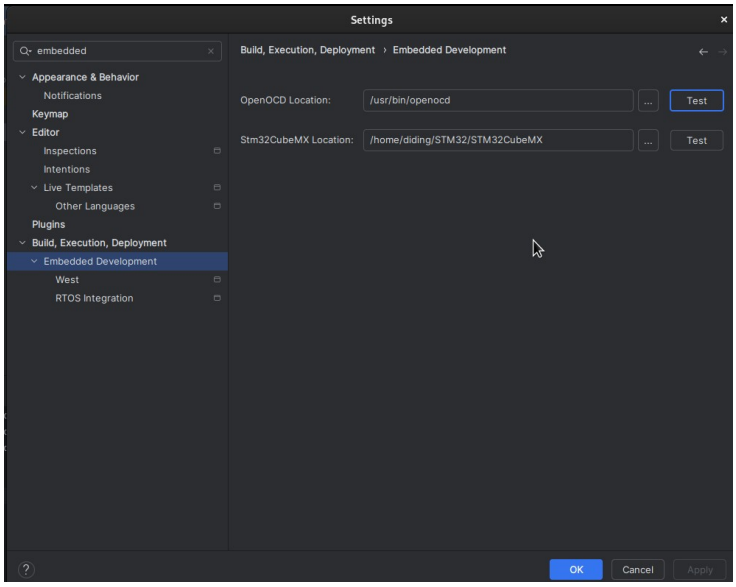
展开“文件(File)”下拉菜单，使缓存失效。这会让 Clion 刷新他的缓存并重启。（右图）

如果只是 CMake 有些反应迟钝，可以尝试刷新 CMake 的缓存。展开工具选项卡，展开 CMake 选项卡，就有清空缓存并重载入项目。（下图）



错误之--Clion 外部工具错误

配置 Stm32CubeMX\OpenOCD



检测他们有没有配置好

配置工具链

Environment 需要指定到 MinGW.

但是 C/Cpp 编译器需要指定到 arm-none-eabi-gcc/arm-none-eabi-gcc.

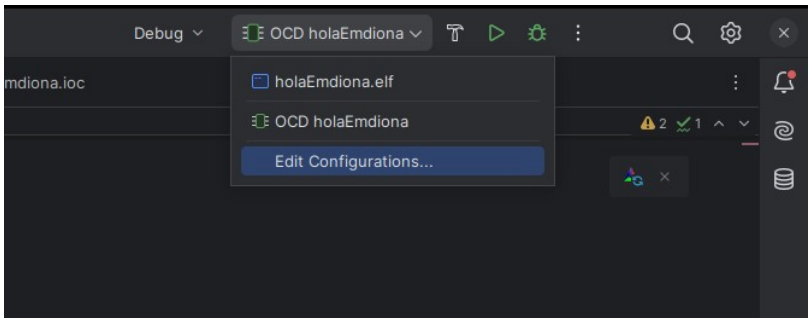
注意检查 minGW 工具集是不是 GNU 标准、x86\i386 指令集的 mingw.

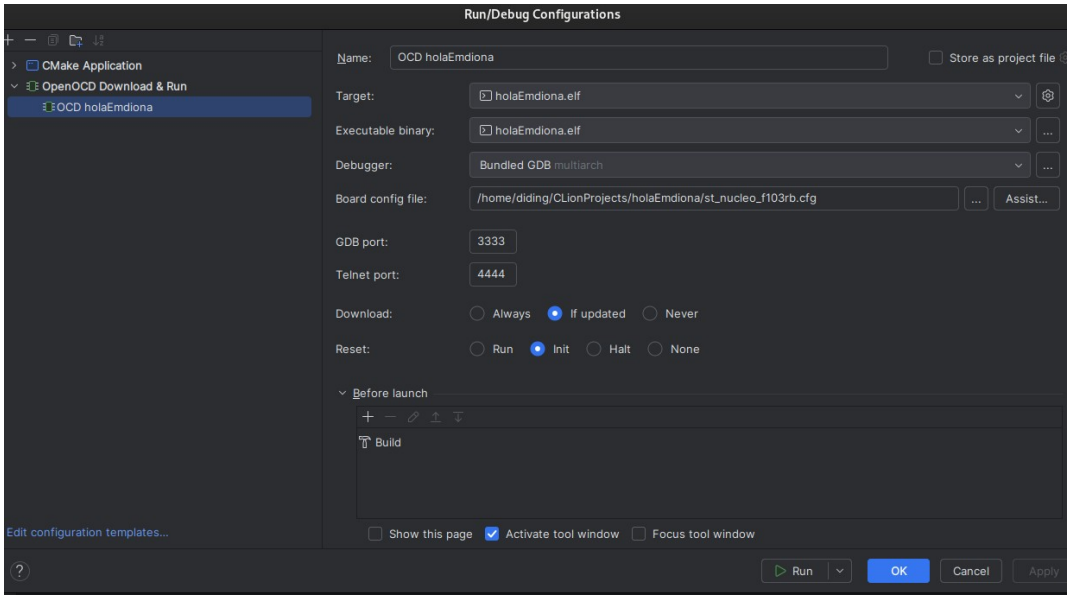
配置 Cmake

（一般是自带的，配置好的）如果出故障，可以手动安装一个 cmake 或者检查项目创建时是不是完整的(如果项目不完整可能没有 CMakeLists.txt）

使用 openOCD 调试

检查 GDB 调试器是不是正确地被配置。（操作系统不一样，仅供参考）



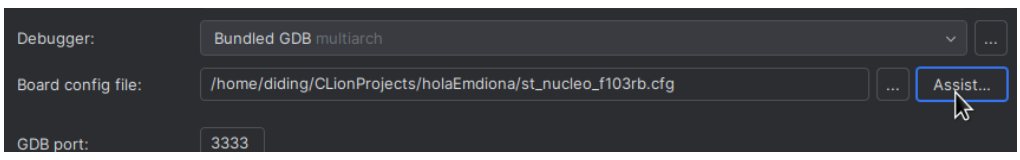


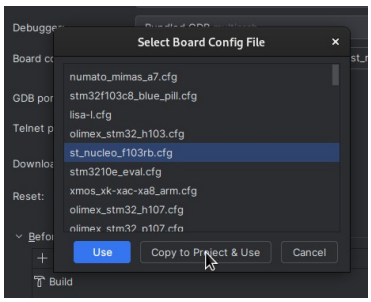
注意：

*.elf 文件是需要 build 生成的，如果编译不通过就找不到。

*如果一开始就没有 OpenOCD build and run 选项，就使用左上角“+”按钮创建一个。

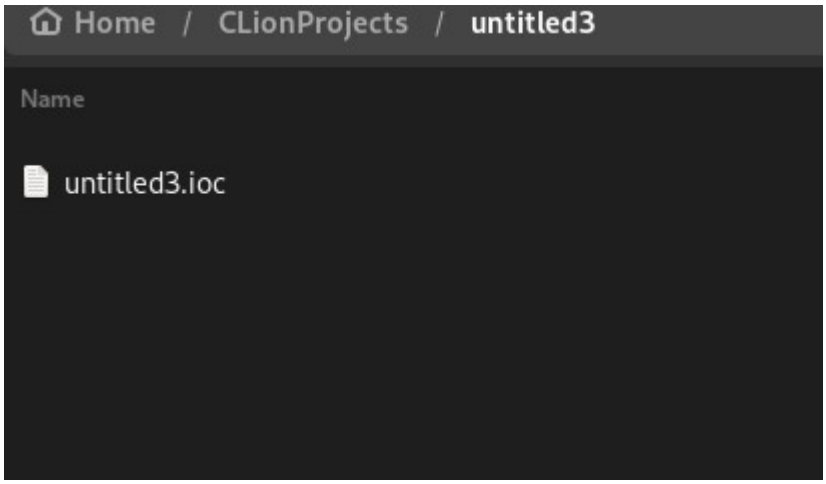
*Board config file 面板配置文件（这个翻译也许有问题？）如果没有（一般在创建项目时会询问），需要在协助 Assis 里选择，并且需要复制到当前目录，使其可以修改。





错误之--项目文件破损

项目中仅有一个*.ioc 文件，或者缺少很多关键文件



请检查 stm32CUBEMX 是否登陆，一些关键组件需要登陆才能下载。