

mbedOS 是 ARM 自己打造、主打 IoT 的解决方案，是一个针对 ARM CortexM 系列处理器的嵌入式开源生态。

开发工具介绍

mbedOS 支持三种开发工具：

1. 在线 IDE
2. mbed CLI 控制台
3. 第三方开发工具，如 IAR，MDK

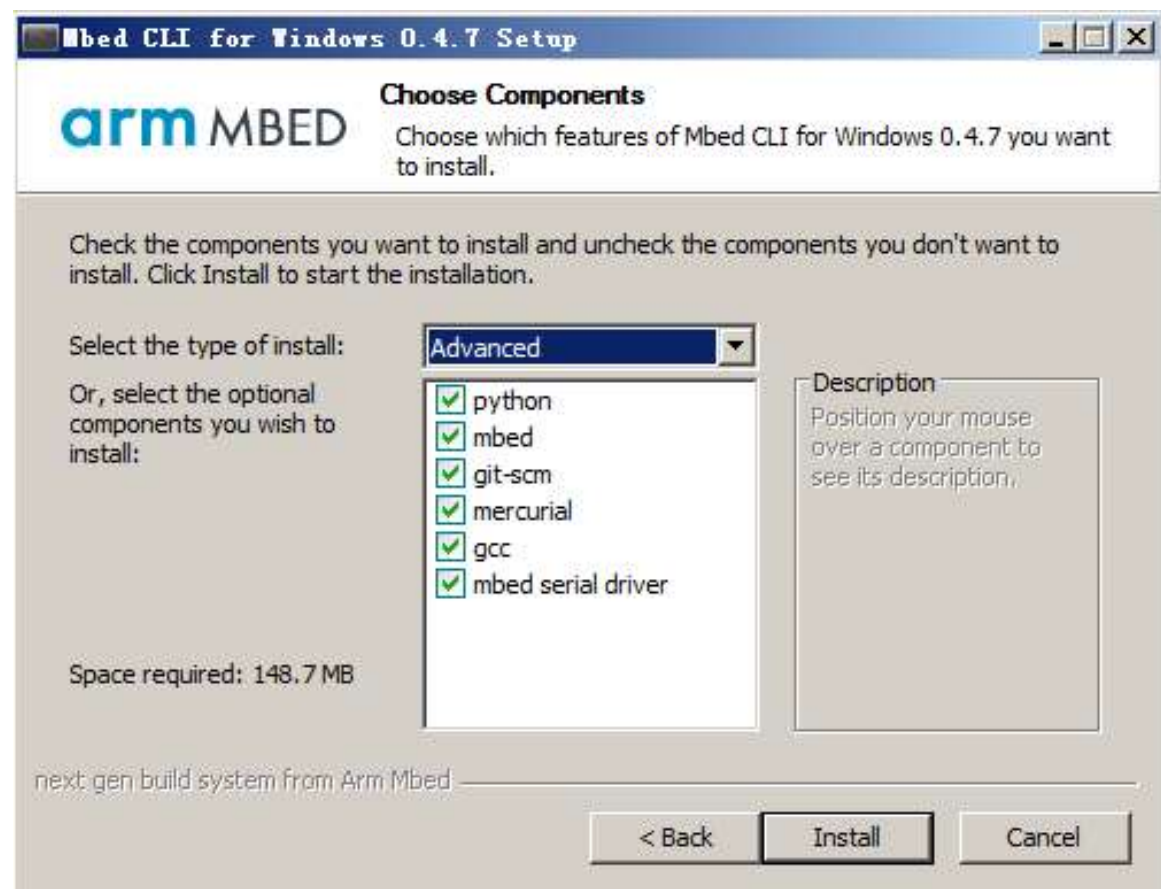
在线 IDE 编译很方便快捷，但没有调试功能。第三方的 IDE 对于我们来说也不是重点。我们着重会来介绍 mbed-cli，它可以代码仓库版本控制、依赖管理、代码发布、从其他地方获取代码、调用编译系统及其他。

Mbed CLI 安装

Windows 安装比较简单，直接到官网

(<https://os.mbed.com/docs/latest/tools/installation-and-setup.html>)

下载安装，下载链接 http://mbed-os.s3-eu-west-1.amazonaws.com/builds/Mbed_installer_v0.4.7.exe



LINUX 安装

需要先安装以下工具

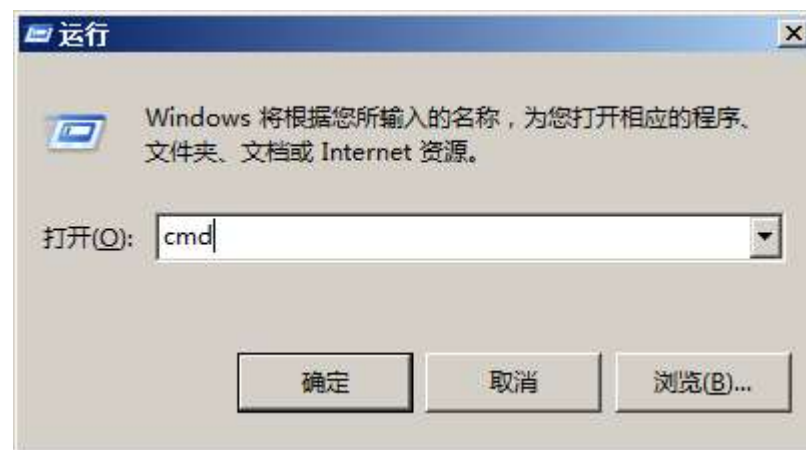
- Python - mbed CLI 是用 Python 写的，并且在 version 2.7.13 上做过完整测试，不兼容 Python3.x.
- Git - version 1.9.5 or later
- Mercurial - version 2.2.2 or later
- GNU ARM - ARM GCC 交叉编译工具

```
$ git clone https://github.com/ARMmbed/mbed-cli
```

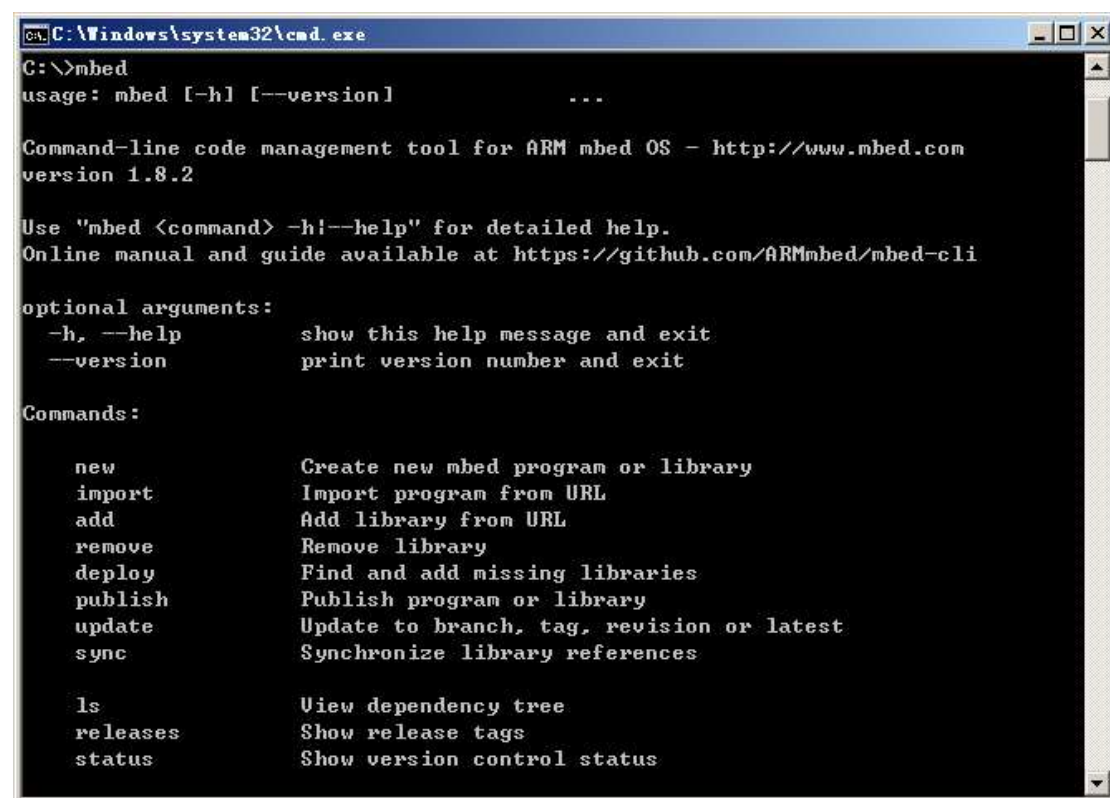
```
$ python setup.py install
```

参考链接 https://docs.mbed.com/docs/mbed-os-handbook/en/5.1/dev_tools/cli/

Windows Mbed CLI 运行



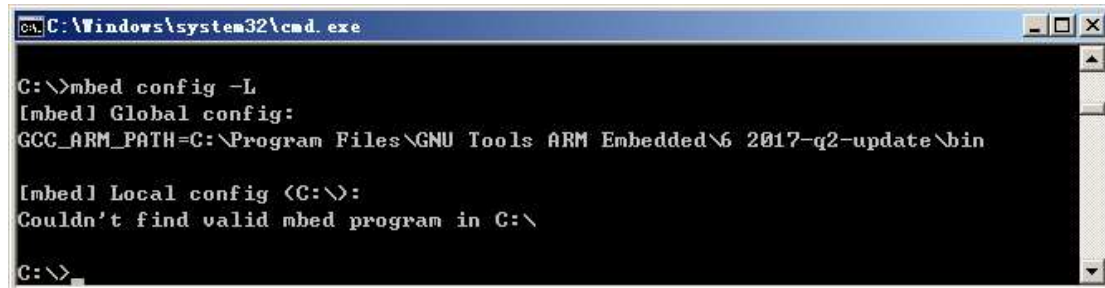
WIN 键+R 键，进入命令行，输入 mbed 运行。



显示 Mbed CLI 的常见参数。

`mbed config -L` 检查配置

如果没有配置下 GCC 的路径（你安装 GNU ARM 的路径），请按如下命令设置：
`mbed config --global GCC_ARM_PATH "C:\Program Files (x86)\GNU Tools ARM Embedded\6 2017-q2-update\bin"`



```
C:\Windows\system32\cmd.exe

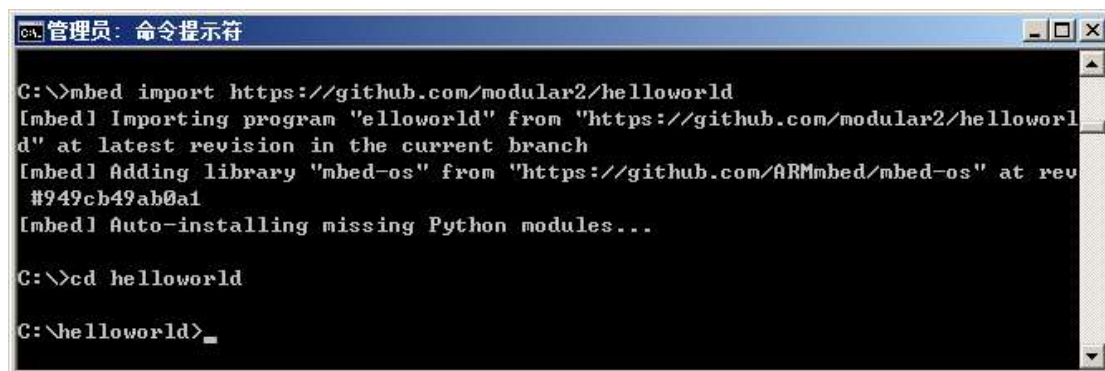
C:\>mbed config -L
[mbed] Global config:
GCC_ARM_PATH=C:\Program Files\GNU Tools ARM Embedded\6 2017-q2-update\bin

[mbed] Local config <C:\>:
Couldn't find valid mbed program in C:\

C:\>
```

快速例子

```
mbed import https://github.com/modular2/helloworld
cd helloworld
```



```
管理员: 命令提示符

C:\>mbed import https://github.com/modular2/helloworld
[mbed] Importing program "elloworld" from "https://github.com/modular2/helloworld" at latest revision in the current branch
[mbed] Adding library "mbed-os" from "https://github.com/ARMmbed/mbed-os" at rev #949cb49ab0a1
[mbed] Auto-installing missing Python modules...

C:\>cd helloworld

C:\helloworld>
```

`mbed compile -S` 检查一下支持列表

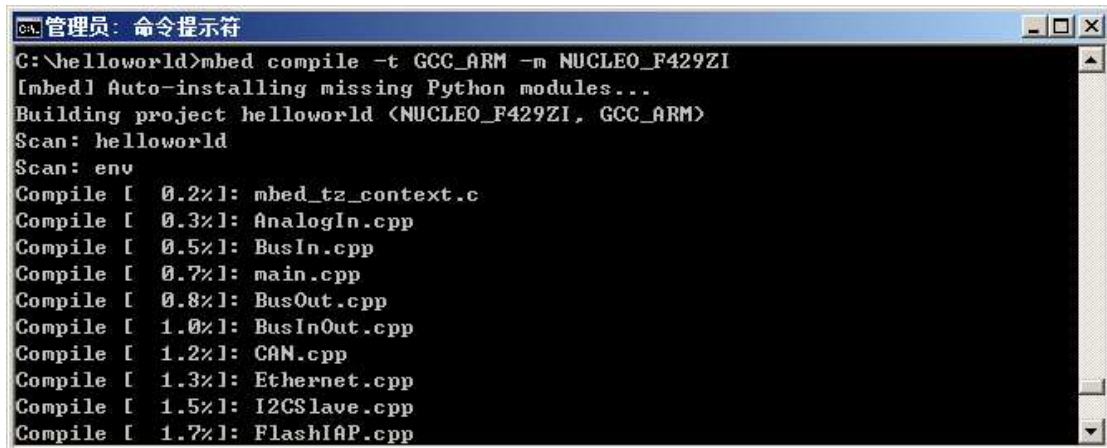


```
管理员: 命令提示符

! NUCLEO_F410RB      ! Supported ! Supported ! Supported ! Supported ! Support
ted ! -            ! Supported ! Supported !
! NUCLEO_F411RE      ! Supported ! Supported ! Supported ! Supported ! Support
ted ! -            ! Supported ! Supported !
! NUCLEO_F412ZG      ! Supported ! Supported ! Supported ! Supported ! Support
ted ! -            ! Supported ! Supported !
! NUCLEO_F413ZH      ! Supported ! Supported ! Supported ! Supported ! Support
ted ! -            ! Supported ! Supported !
! NUCLEO_F429ZI      ! Supported ! Supported ! Supported ! Supported ! Support
ted ! -            ! Supported ! Supported !
! NUCLEO_F439ZI      ! Supported ! Supported ! Supported ! Supported ! Support
ted ! -            ! Supported ! Supported !
! NUCLEO_F446RE      ! Supported ! Supported ! Supported ! Supported ! Support
ted ! -            ! Supported ! Supported !
! NUCLEO_F446ZE      ! Supported ! Supported ! Supported ! Supported ! Support
ted ! -            ! Supported ! Supported !
! NUCLEO_F746ZG      ! Supported ! Supported ! Supported ! Supported ! Support
ted ! -            ! Supported ! Supported !
! NUCLEO_F756ZG      ! Supported ! Supported ! Supported ! Supported ! Support
ted ! -            ! Supported ! Supported !
! NUCLEO_F767ZI      ! Supported ! Supported ! Supported ! Supported ! Support
ted ! -            ! Supported ! Supported !
! NUCLEO_L011K4      ! Supported ! -       ! -       ! -       ! Support
ted ! -            ! -       ! -       ! -       !
```

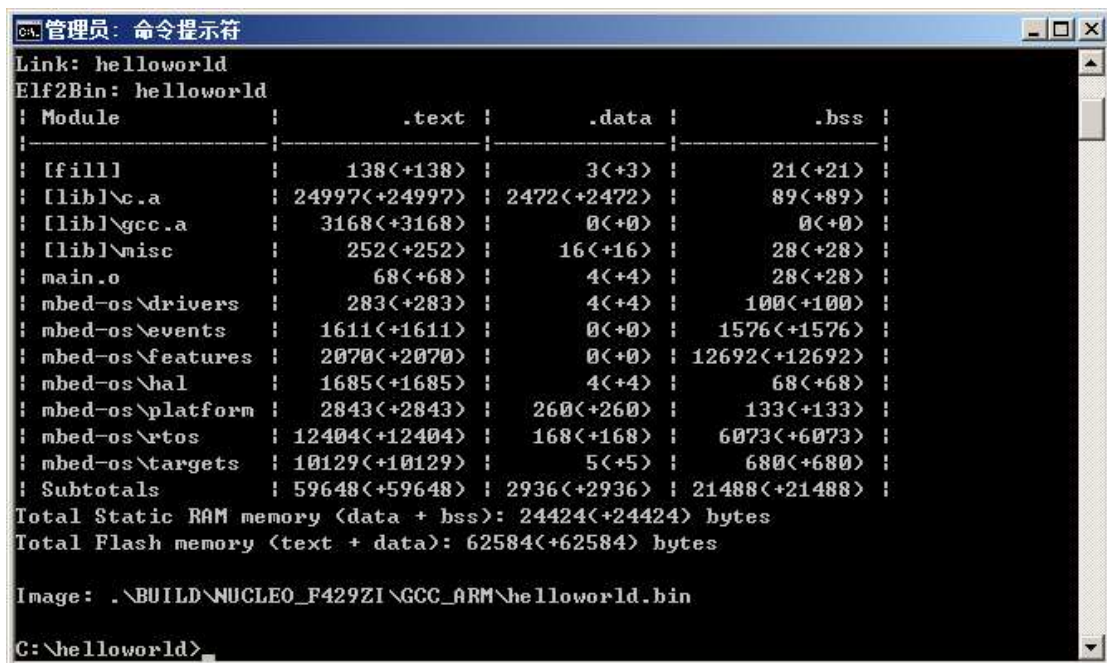
针对 Modular-2 进行编译时，开发板选择使用相同 CPU 的 NUCLEO_F429ZI。

```
mbed compile -t GCC_ARM -m NUCLEO_F429ZI
```



```
C:\helloworld>mbed compile -t GCC_ARM -m NUCLEO_F429ZI
[mbed] Auto-installing missing Python modules...
Building project helloworld (NUCLEO_F429ZI, GCC_ARM)
Scan: helloworld
Scan: env
Compile [ 0.2%]: mbed_tz_context.c
Compile [ 0.3%]: AnalogIn.cpp
Compile [ 0.5%]: BusIn.cpp
Compile [ 0.7%]: main.cpp
Compile [ 0.8%]: BusOut.cpp
Compile [ 1.0%]: BusInOut.cpp
Compile [ 1.2%]: CAN.cpp
Compile [ 1.3%]: Ethernet.cpp
Compile [ 1.5%]: I2CSlave.cpp
Compile [ 1.7%]: FlashIAP.cpp
```

编译结束生成 helloworld.bin 文件。



```
C:\helloworld>mbed compile -t GCC_ARM -m NUCLEO_F429ZI
Link: helloworld
Elf2Bin: helloworld
! Module ! ! .text ! ! .data ! ! .bss !
!-----! !-----! !-----! !-----!
! [fill] ! ! 138(+138) ! ! 3(+3) ! ! 21(+21) !
! [lib]c.a ! ! 24997(+24997) ! ! 2472(+2472) ! ! 89(+89) !
! [lib]gcc.a ! ! 3168(+3168) ! ! 0(+0) ! ! 0(+0) !
! [lib]misc ! ! 252(+252) ! ! 16(+16) ! ! 28(+28) !
! main.o ! ! 68(+68) ! ! 4(+4) ! ! 28(+28) !
! mbed-os\drivers ! ! 283(+283) ! ! 4(+4) ! ! 100(+100) !
! mbed-os\events ! ! 1611(+1611) ! ! 0(+0) ! ! 1576(+1576) !
! mbed-os\features ! ! 2070(+2070) ! ! 0(+0) ! ! 12692(+12692) !
! mbed-os\hal ! ! 1685(+1685) ! ! 4(+4) ! ! 68(+68) !
! mbed-os\platform ! ! 2843(+2843) ! ! 260(+260) ! ! 133(+133) !
! mbed-os\rtos ! ! 12404(+12404) ! ! 168(+168) ! ! 6073(+6073) !
! mbed-os\targets ! ! 10129(+10129) ! ! 5(+5) ! ! 680(+680) !
! Subtotals ! ! 59648(+59648) ! ! 2936(+2936) ! ! 21488(+21488) !
Total Static RAM memory (data + bss): 24424(+24424) bytes
Total Flash memory (text + data): 62584(+62584) bytes

Image: .\BUILD\NUCLEO_F429ZI\GCC_ARM\helloworld.bin
C:\helloworld>
```

联机烧录

1. 将 modular-2 设备通过 USB (DAPLink 接口) 连接开发电脑。
2. 将生成的 bin 文件复制到 modular-2 生成的存储盘符中。
3. 按复位键启动嵌入式程序。

注意事项

最新源码请查看: <https://github.com/modular2>。

新建文件或项目目录名称不能有空格。

使用虚拟机时，VMware Workstation 环境 mbed import 时有 python 工具问题，建议使用 Oracle VM VirtualBox 环境。