

實驗內容

1. 安裝 Mbed CLI 2

我一開始照著課程投影片的做法，在 WSL (Ubuntu 20.04) 上進行安裝，過程中都沒遇到問題。裝好以後，我使用 mbed-tools detect 卻找不到裝置，這時我才想到 WSL 是不支援 USB 的。所以我只能直接裝在 Windows 上，但沒想到遇到的問題還滿多的。像是直接在系統下用 pip 安裝 mbed-tools 以後，我用 `py -m mbed-tools` 卻怎麼也無法找到 module，最後我還是用 pipenv 開了一個虛擬環境才能正常使用 mbed-tools。除此之外，CMake 必須加到 PATH 裡，在安裝 CMake 時要勾選加入環境變數，且安裝完成後要自己重新開機。

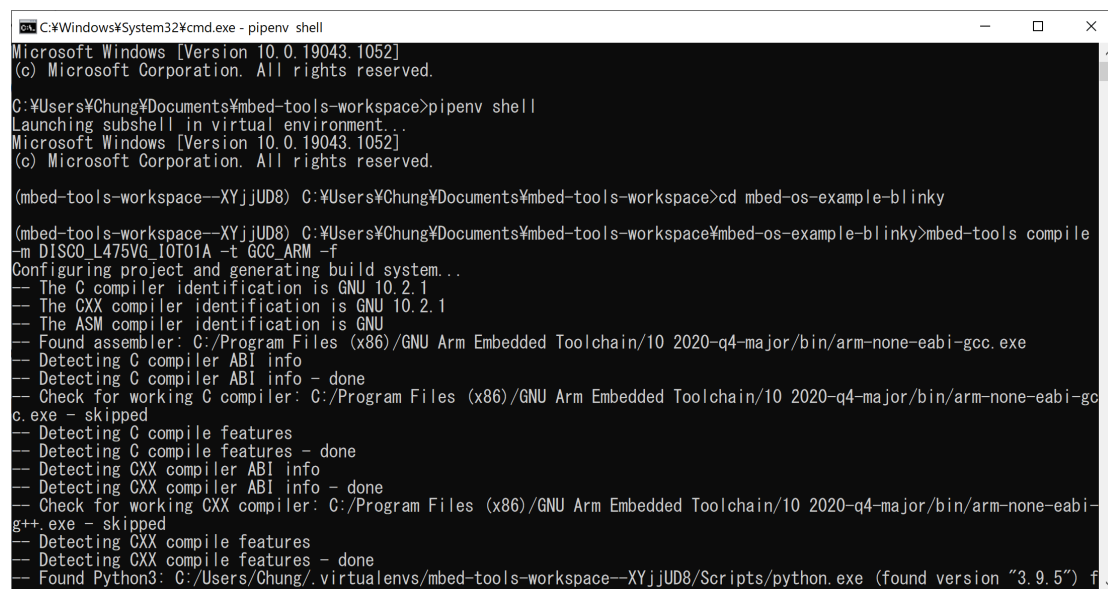
2. 編譯、執行

假設上一步驟都有成功的話，照著課程投影片 import、compile 就可以了。我在本次作業中也是 import mbed-os-example-blinky。

實驗結果

GitHub: <https://github.com/chun9temp/2021ESLab/tree/main/HW9>

圖 1 是 compile 開始時的畫面；圖 2 是 compile 結束後的畫面。完成後 STM32 上的 LED 就會閃了；這部分因為之前也有做過，這次就不再錄實際執行的狀況了。



```
C:\Windows\System32\cmd.exe - pipenv shell
Microsoft Windows [Version 10.0.19043.1052]
(c) Microsoft Corporation. All rights reserved.

C:\Users\YChung\Documents\mbed-tools-workspace>pipenv shell
Launching subshell in virtual environment...
Microsoft Windows [Version 10.0.19043.1052]
(c) Microsoft Corporation. All rights reserved.

(mbed-tools-workspace--XYjjUD8) C:\Users\YChung\Documents\mbed-tools-workspace>cd mbed-os-example-blinky
(mbed-tools-workspace--XYjjUD8) C:\Users\YChung\Documents\mbed-tools-workspace\mbed-os-example-blinky>mbed-tools compile
--m DISCO_L475VG_I0T01A -t GCC_ARM -f
Configuring project and generating build system...
-- The C compiler identification is GNU 10.2.1
-- The CXX compiler identification is GNU 10.2.1
-- The ASM compiler identification is GNU
-- Found assembler: C:/Program Files (x86)/GNU Arm Embedded Toolchain/10 2020-q4-major/bin/arm-none-eabi-gcc.exe
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Check for working C compiler: C:/Program Files (x86)/GNU Arm Embedded Toolchain/10 2020-q4-major/bin/arm-none-eabi-gc
c.exe - skipped
-- Detecting C compile features
-- Detecting C compile features - done
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Check for working CXX compiler: C:/Program Files (x86)/GNU Arm Embedded Toolchain/10 2020-q4-major/bin/arm-none-eabi-
g++.exe - skipped
-- Detecting CXX compile features
-- Detecting CXX compile features - done
-- Found Python3: C:/Users/Chung/.virtualenvs/mbed-tools-workspace--XYjjUD8/Scripts/python.exe (found version "3.9.5") f
```

圖 1 Compile 開始

```

C:\Windows\System32\cmd.exe - pipenv shell
but argument 8 has type 'uint32_t' {aka 'long unsigned int'} [-Wformat=]
375 |     debug_if(qspi_api_c_debug, "qspi_prepare_command Out: InstructionMode %x Instruction %x AddressMode %x AddressSize %x Address %x DataMode %x\n",
    |                                     ~^
    |                                     |
    |                                     unsigned int
    |                                     %lx
376 |     st_command->InstructionMode, st_command->Instruction, st_command->AddressMode, st_command->AddressSize, st_command->Address, st_command->DataMode);
    |                                     ~~~~~
    |                                     |
    |                                     uint32_t {aka long unsigned int}
[263/263] Linking CXX executable mbed-os-example-blinky.elf
-- built: C:/Users/Chung/Documents/mbed-tools-workspace/mbed-os-example-blinky/cmake_build/DISCO_L475VG_IOT01A/develop/GCC_ARM/mbed-os-example-blinky.bin
-- built: C:/Users/Chung/Documents/mbed-tools-workspace/mbed-os-example-blinky/cmake_build/DISCO_L475VG_IOT01A/develop/GCC_ARM/mbed-os-example-blinky.hex
Copied C:\Users\Chung\Documents\mbed-tools-workspace\mbed-os-example-blinky\cmake_build\DISCO_L475VG_IOT01A\develop\GCC_ARM\mbed-os-example-blinky.bin to 1 device(s).
(mbed-tools-workspace--XYjjUD8) C:\Users\Chung\Documents\mbed-tools-workspace\mbed-os-example-blinky>

```

圖 2 Compile 結束

討論

1. Linux 中的 source 指令會在當下的 shell 執行所指定檔案所包含的指令，常用來載入環境設定，像是 PATH、license。
2. Mbed CLI 2 利用 CMake 作為 build generator、Ninja 作為 build system，能用簡單的指令更換 compiler。
3. 可以使用指令「cmake_build/<target>/<profile>/<toolchain>」，此指令會建立一個屬於該設定組合的子路徑，並在該路徑下 build project。