

ใบงานที่ 2

เครื่องมือ espressif/editors

อุปกรณ์ที่ต้องใช้ในการทดลอง

1. ESP32Devkit V.1
2. USB / micro USB
3. Computer

การพัฒนาและการทำงานเครื่องมือเบื้องต้น

1. หา Board ที่ต้องการใน device folder
 - a. หากเป็น Linux จะอยู่ที่ “/dev” Folder และโดยส่วนใหญ่จะชื่อ device เป็น “ttyUSB0”

```
~/Documents/ESP  
→ ls /dev/ttyUSB0  
/dev/ttyUSB0  
~/Documents/ESP  
→
```

2. ทำการเรียกใช้ตัว toolchain “espressif/idf” พร้อมทั้งทำการ pass-through board device (ttyUSB0) ให้กับ container ที่ run ผ่านทาง docker command
 - a. Bash – shell command เพื่อใช้สั่งงาน
 - b. device /dev/ttyUSB0 – pass-through device to container
 - c. -v “\$PWD:/data” – ทำการ map current directory ของ Host ไปยัง /data ใน container

```
~/Documents/ESP  
→ docker run --rm -it -v "$PWD:/data" -w "/data" --device /dev/ttyUSB0 espressif/idf:release-v5.1 bash
```

3. เมื่อเข้ามาใน container และแสดง shell เพื่อรับคำสั่งจากผู้ใช้งาน

```
/opt/esp/Tools/boots  
Done! You can now compile ESP-IDF projects.  
Go to the project directory and run:  
  
idf.py build  
root@c60105f62776:/data#
```

4. เมื่อผู้ใช้งานไม่ทราบ parameter ต่างๆ ของคำสั่ง idf.py สามารถเรียกดูได้จาก Help

```
root@c60105f62776:/data# idf.py --help
Usage: idf.py [OPTIONS] COMMAND1 [ARGS]... [COMMAND2 [ARGS]...]...

ESP-IDF CLI build management tool. For commands that are not known to idf.py an attempt to execute it as a build system target will be made. Selected target: None

Options:
  --version                Show IDF version and exit.
  --list-targets           Print list of supported targets and exit.
  -C, --project-dir PATH  Project directory.
  -B, --build-dir PATH    Build directory.
  -w, --cmake-warn-uninitialized / -n, --no-warnings
                        Enable CMake uninitialized variable warnings for CMake files inside the project directory. (--no-warnings is now the default, and doesn't need to be specified.) The default value can be set with the IDF_CMAKE_WARN_UNINITIALIZED environment variable.
  -v, --verbose           Verbose build output.
  --preview              Enable IDF features that are still in preview.
  --ccache / --no-ccache Use ccache in build. Disabled by default. The default value can be set with the IDF_CCACHE_ENABLE environment variable.
  -C, --cmake-args [ARGS] CMake arguments to pass to the build system.
```

5. สร้าง project ใหม่ได้จากคำสั่ง idf.py create-project <ชื่อโปรเจกต์ที่ต้องการ>

```
root@c60105f62776:/data# idf.py create-project Lab02
Executing action: create-project
The project was created in /data/Lab02
root@c60105f62776:/data# ls
Lab02  test.c
root@c60105f62776:/data#
```

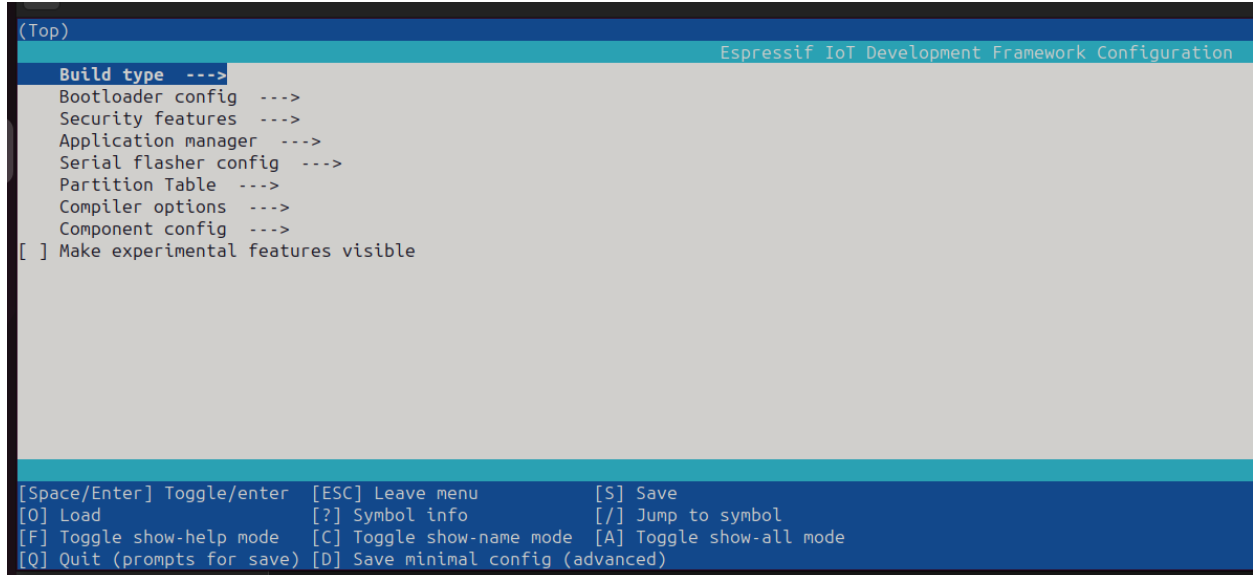
6. จากข้อที่แล้วได้สร้างโปรเจกต์ใหม่ชื่อ Lab02 ให้ทำการเปลี่ยน directory ไปยังโปรเจกต์ดังกล่าว และทำการ set รุ่นของ ESP ตามที่ต้องการในที่นี้คือ esp32 จาก argument set-target ของ idf.py

```
root@13b930f43678:/data/Lab02# idf.py set-target esp32
```

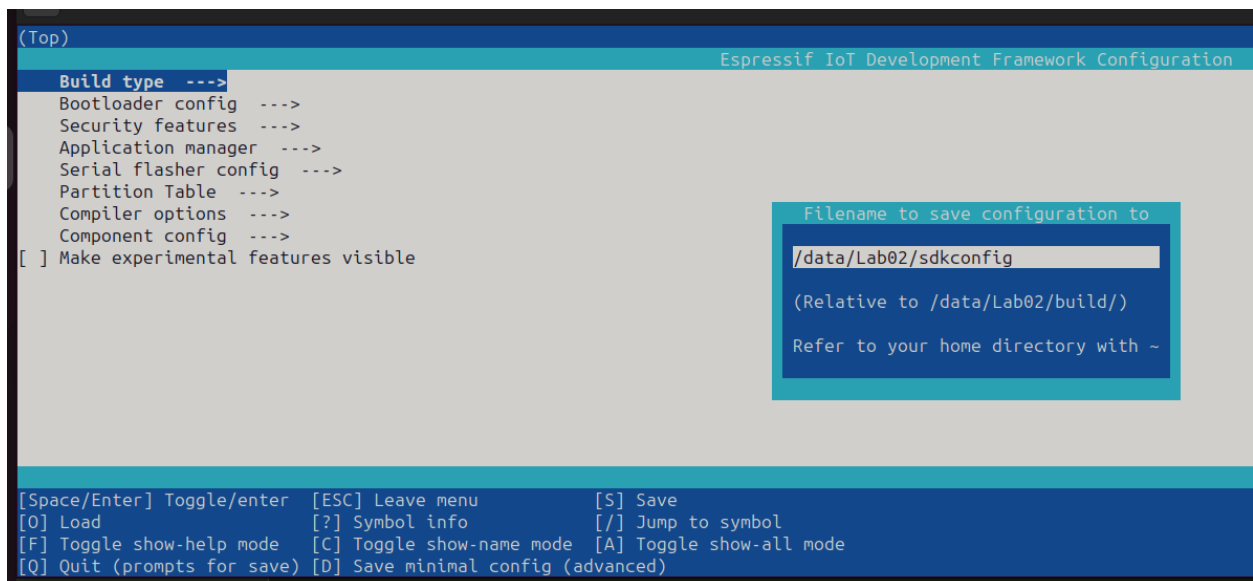
7. เมื่อต้องการ configuration ของบอร์ดสามารถทำได้โดย idf.py menuconfig เช่น เปิด/ปิดการใช้งาน Bluetooth หรือการกำหนดระดับของการ compile (size, performance)

```
root@c60105f62776:/data/Lab02# idf.py menuconfig
```

8. หน้าต่างสำหรับการ configure ค่าต่างๆ ที่เกี่ยวข้อง



9. หน้าต่างแสดงการ save ค่า configuration ลงไปเก็บไว้ในไฟล์ เพื่อให้ compiler มาอ่านและนำไปกำหนดค่าตามที่ใช้ใช้งานต้องการ



10. หลังจากทำการconfiguration ค่าต่างๆ และ save เรียบร้อย จะเห็นได้ว่าจะมีไฟล์เพิ่มขึ้นมา จากนั้นผู้ใช้งานทำการ build ตัวโปรเจกต์เพื่อให้เป็น binary file ได้จาก idf.py build

```
root@c60105f62776:/data/Lab02# ls
CMakeLists.txt  build  main  sdkconfig  sdkconfig.old
root@c60105f62776:/data/Lab02# idf.py build
```

11. หน้าจอตัวอย่างเมื่อตัว tool กำลัง build application ให้เป็น binary file เพื่อให้สามารถ burn ลงบอร์ดได้

```
-- Generating done
-- Build files have been written to: /data/Lab02/build/bootloader
[102/103] Generating binary image from built executablesptool.py v4.5.1
Creating esp32 image...
Merged 1 ELF section
Successfully created esp32 image.
Generated /data/Lab02/build/bootloader/bootloader.bin
[103/103] cd /data/Lab02/build/bootloader/esp-idf/esptool.py && /opt/esp/python_env/idf5.1...eck_sizes.py
Bootloader binary size 0x6820 bytes. 0x7e0 bytes (7%) free.
[612/884] Building C object esp-idf/tcp_transport/CMakeFiles/_idf_tcp_transport.dir/transport_ssl.c.obj
```

12. เมื่อตัว tool ทำการ build เสร็จสิ้นโดยไม่มี error ใดๆ เกิดขึ้น จะมี folder ที่ชื่อ build เพิ่มขึ้นมาซึ่งเป็นที่เก็บไฟล์ที่ผ่านการ compile รวมทั้ง binary file ของ application ด้วย

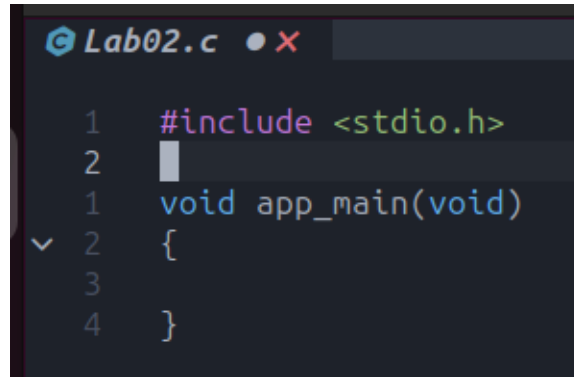
```
root@c60105f62776:/data/Lab02# ls
CMakeLists.txt  build  main  sdkconfig  sdkconfig.old
root@c60105f62776:/data/Lab02#
```

การ Burn หรือ Flash ตัว application ที่เป็น binary file ลงไปที่บอร์ด

13. สามารถทำการ Flash application (Burn) ลงบอร์ดได้ตามคำสั่งด้านล่าง

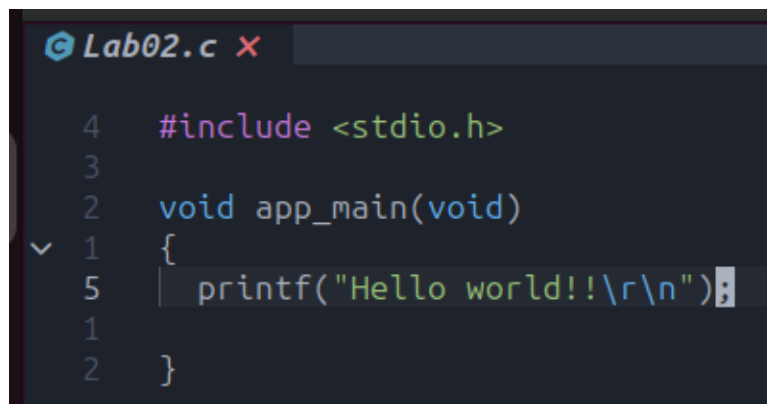
```
root@c60105f62776:/data/Lab02# idf.py -p /dev/ttyUSB0 flash
```

14. ไฟล์ที่ใช้ในการเขียนโค้ดที่ต้องการจะอยู่ที่ <project>/main/<ไฟล์นามสกุล.c>



```
Lab02.c X
1  #include <stdio.h>
2
1  void app_main(void)
2  {
3
4  }
```

15. ทดลองเพิ่มโค้ดเพื่อแสดงข้อความ “Hello world!!” เข้าไป

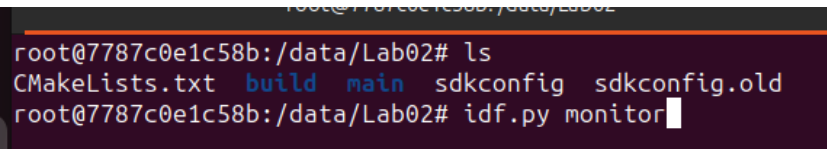


```
Lab02.c X
4  #include <stdio.h>
3
2  void app_main(void)
1  {
5  | printf("Hello world!!\r\n");
1
2  }
```

16. ทำการ flash application ลงบอร์ด

การ monitor ข้อความที่บอร์ดส่งมาที่เครื่องคอมพิวเตอร์ผ่านทางสาย USB

17. ผู้ใช้งานสามารถดูข้อความที่ส่งมาจากบอร์ดโดยใช้คำสั่ง idf.py monitor



```
root@7787c0e1c58b:/data/Lab02# ls
CMakeLists.txt  build  main  sdkconfig  sdkconfig.old
root@7787c0e1c58b:/data/Lab02# idf.py monitor
```

18. Keyboard shortcut ที่ใช้เมื่อทำการใช้คำสั่ง idf.py monitor ไปแล้ว

Keyboard Shortcut	Action	Description
Ctrl+]	Exit the program	
Ctrl+T	Menu escape key	Press and follow it by one of the keys given below.
• Ctrl+T	Send the menu character itself to remote	
• Ctrl+]	Send the exit character itself to remote	
• Ctrl+P	Reset target into bootloader to pause app via RTS line	Resets the target, into bootloader via the RTS line (if connected), so that the board runs nothing. Useful when you need to wait for another device to startup.
• Ctrl+R	Reset target board via RTS	Resets the target board and re-starts the application via the RTS line (if connected).
• Ctrl+F	Build and flash the project	Pauses idf_monitor to run the project flash target, then resumes idf_monitor. Any changed source files are recompiled and then re-flashed. Target encrypted-flash is run if idf_monitor was started with argument -E .
• Ctrl+A (or A)	Build and flash the app only	Pauses idf_monitor to run the app-flash target, then resumes idf_monitor. Similar to the flash target, but only the main app is built and re-flashed. Target encrypted-app-flash is run if idf_monitor was started with argument -E .

19. หน้าจอตัวอย่างเมื่อ monitor ทำการเชื่อมต่อไปยัง target board

```

root@7787c0e1c58b:/data/Lab02# ls
CMakeLists.txt  build  main  sdkconfig  sdkconfig.old
root@7787c0e1c58b:/data/Lab02# idf.py monitor
Executing action: monitor
Serial port /dev/ttyUSB0
Connecting.....
Detecting chip type... Unsupported detection protocol, switching and trying again...
Connecting...

```

20. ตัวอย่างข้อความ “Hello world!!” แสดงผลผ่านทาง Terminal ของเครื่องคอมพิวเตอร์

```

I (295) spi_flash: detected chip: generic
I (298) spi_flash: flash io: dio
W (302) spi_flash: Detected size(4096k) larger
I (315) app_start: Starting scheduler on CPU0
I (320) app_start: Starting scheduler on CPU1
I (320) main_task: Started on CPU0
I (330) main_task: Calling app_main()
Hello world!!
I (330) main_task: Returned from app_main()

```

ทดลอง

ให้นักศึกษาทำการเพิ่มข้อความตามข้อกำหนดด้านล่าง และ flash ลงบอร์ด พร้อมแสดงข้อความบนหน้าจอเครื่องคอมพิวเตอร์ของนักศึกษา

- ชื่อ – นามสกุล
- รหัสนักศึกษา