Embedded System Design with MCU and FPGA

LAB01 0858706 黃品嫚

Goal

- To know how does Arduino do when it compiles., and know how to use Arduino Software IDE.

Environment

- **Mac**

Problem

1. Where is the main( )?

- `main( )` is in main.cpp and it includes `init`, `setup`, `loop` and a few other things.
- The location of main.cpp is "**/Applications/Arduino.app/Contents/Java/hardware/arduino/avr/cores/arduino/main.cpp**" on Mac.
- From the above location, we could know that the Arduino Software IDE is written in Java, so it is a cross-platform software.
- When we write code on Arduino Software IDE, we will find its filename extension is .ino. The .ino files are all combined and presented to the compiler as a single .cpp file to compile. (Here is more information about the relationship between the .ino file and .cpp file.)

2. What is the name of the cross compiler?

- Arduino uses the gcc compiler.
- We could find its compiler named **avr-g++ and avr-gcc** from the detail log.

3. Where is the location of the cross compiler?

- The cross compiler, avr-g++ and avr-gcc, is under "**/Applications/Arduino.app/Contents/Java/hardware/tools/avr/bin/**"

4. (Bonus) How do you use the cross compiler without using Arduino IDE?

- There are three methods that an AVR chip can be programmed.
- First, use assembly. Although it may be a little difficult to write and read, it can be very efficient.
- Second, use Atmel's official IDE Atmel Studio. But it is only available on Windows.
- Third, use your favorite **editor** to code and flash using **avr-gcc and avrdude**. Arduino Software IDE also called avr-gcc or acr-g++ to compile, and then sketches are also uploaded by avrdude. [1]

```
uno.upload.tool=avrdude
uno.upload.protocol=arduino
uno.upload.maximum_size=32256
uno.upload.maximum_data_size=2048
uno.upload.speed=115200

uno.bootloader.tool=avrdude
uno.bootloader.low_fuses=0xFF
uno.bootloader.high_fuses=0xDE
uno.bootloader.extended_fuses=0xFD
uno.bootloader.unlock_bits=0x3F
uno.bootloader.lock_bits=0x0F
uno.bootloader.file=optiboot/optiboot_atmega328.hex
```

5. (Bonus) Where is the memory configuration file?

- Memory allocation data is in "**/Applications/Arduino.app/Contents/Java/hardware/arduino/avr/board.txt**" file. In this file, it contains the memory address, size, and other variables.
- And in "/Applications/Arduino.app/Contents/Java/hardware/arduino/avr/platform.txt" it contains the compile variables.

Reference

[1] https://github.com/arduino/Arduino/wiki/Build-Process