

ARC IoTDK

- Environment Build -

Contents

1. Installation
 - GNU Toolchain
 - VSCode Editor(**recommaned**)
 - Putty
2. USB Driver Adjustment
 - Zadig
3. Test
 - embarc_osp

Installation

- GNU Toolchain

Download Link :

<https://github.com/foss-for-synopsys-dwc-arc-processors/toolchain/releases/tag/arc-2020.03-release>

Version : 2020.03

	Linux x86_64	Windows x86_64	Linux ARC HS	macOS x86_64
Baremetal	Little endian \ Big endian			Little endian \ Big endian
Linux/uClibc ARC700	Little endian \ Big endian			
Linux/uClibc ARC HS	Little endian \ Big endian		Little endian	
Linux/glibc ARC HS	Little endian \ Big endian			
IDE	Download	Download		Download



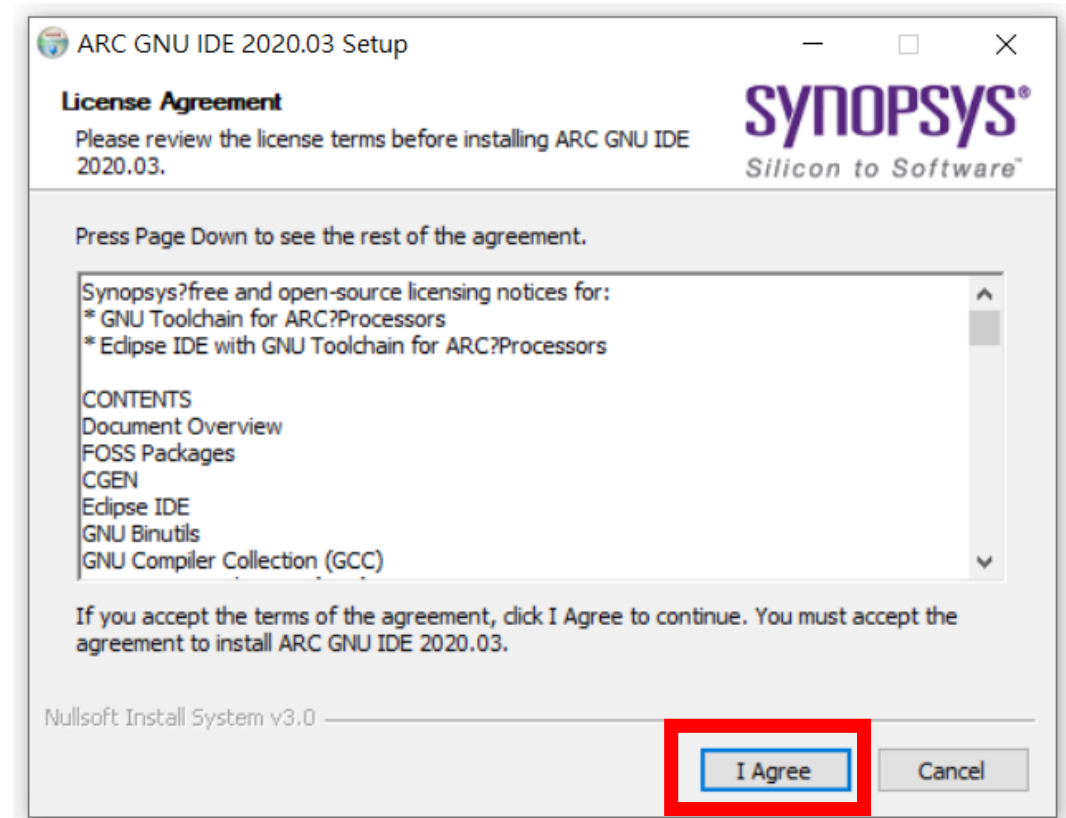
for windows



for mac

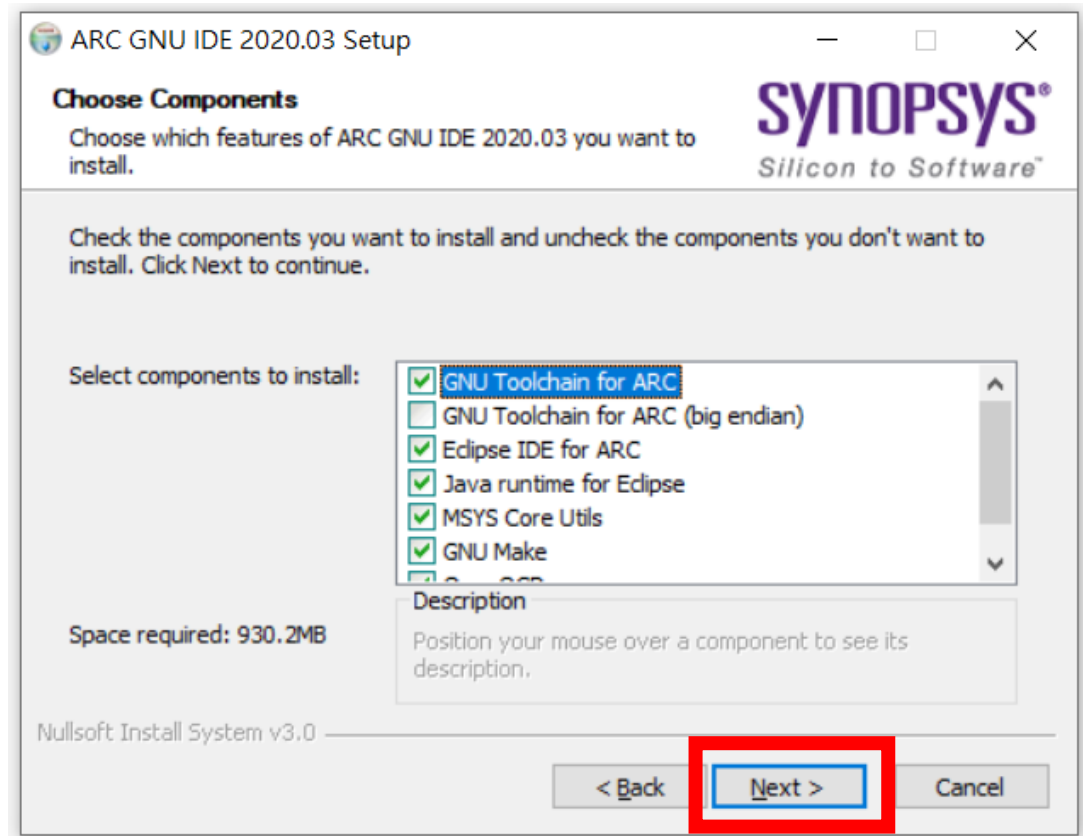
Installation

- GNU Toolchain



Installation

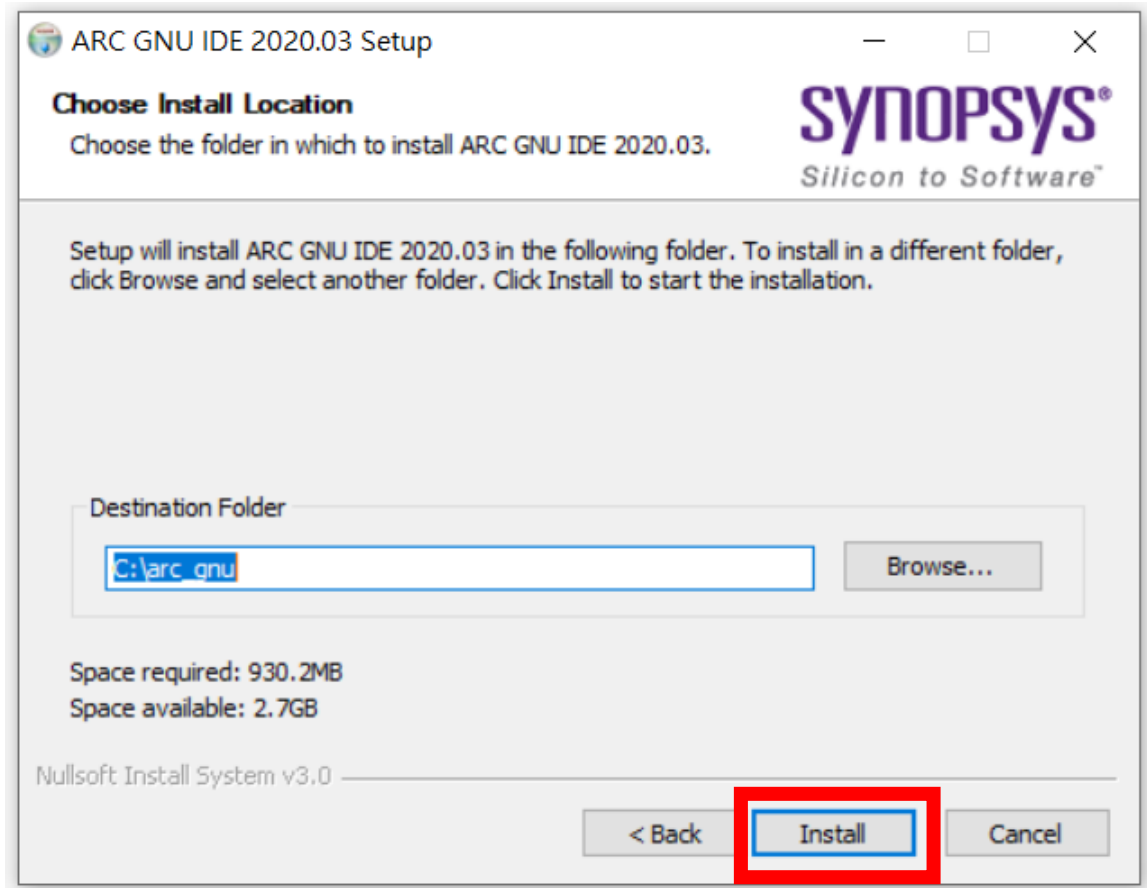
- GNU Toolchain



Installation

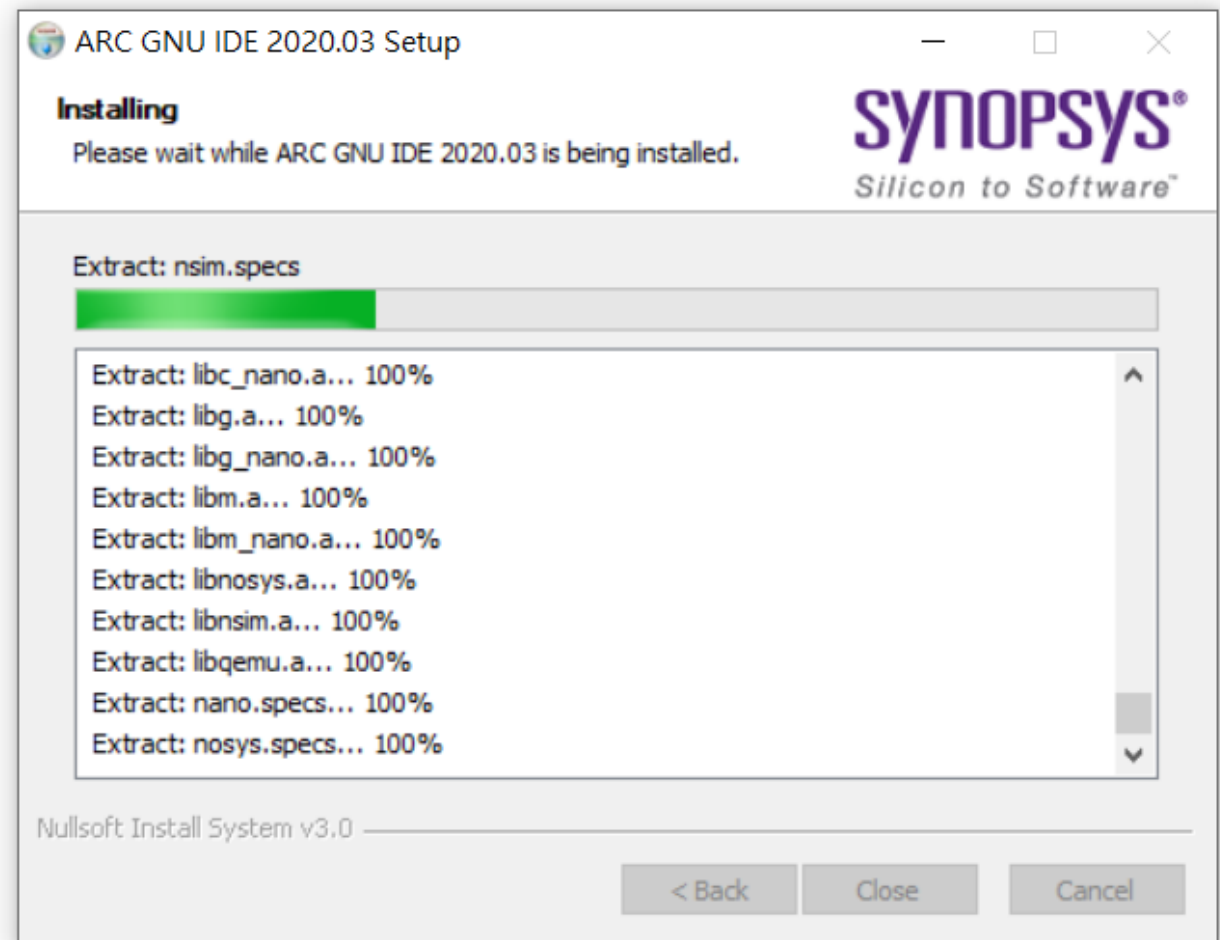
- GNU Tollchain

You can choose your own Destination Folder



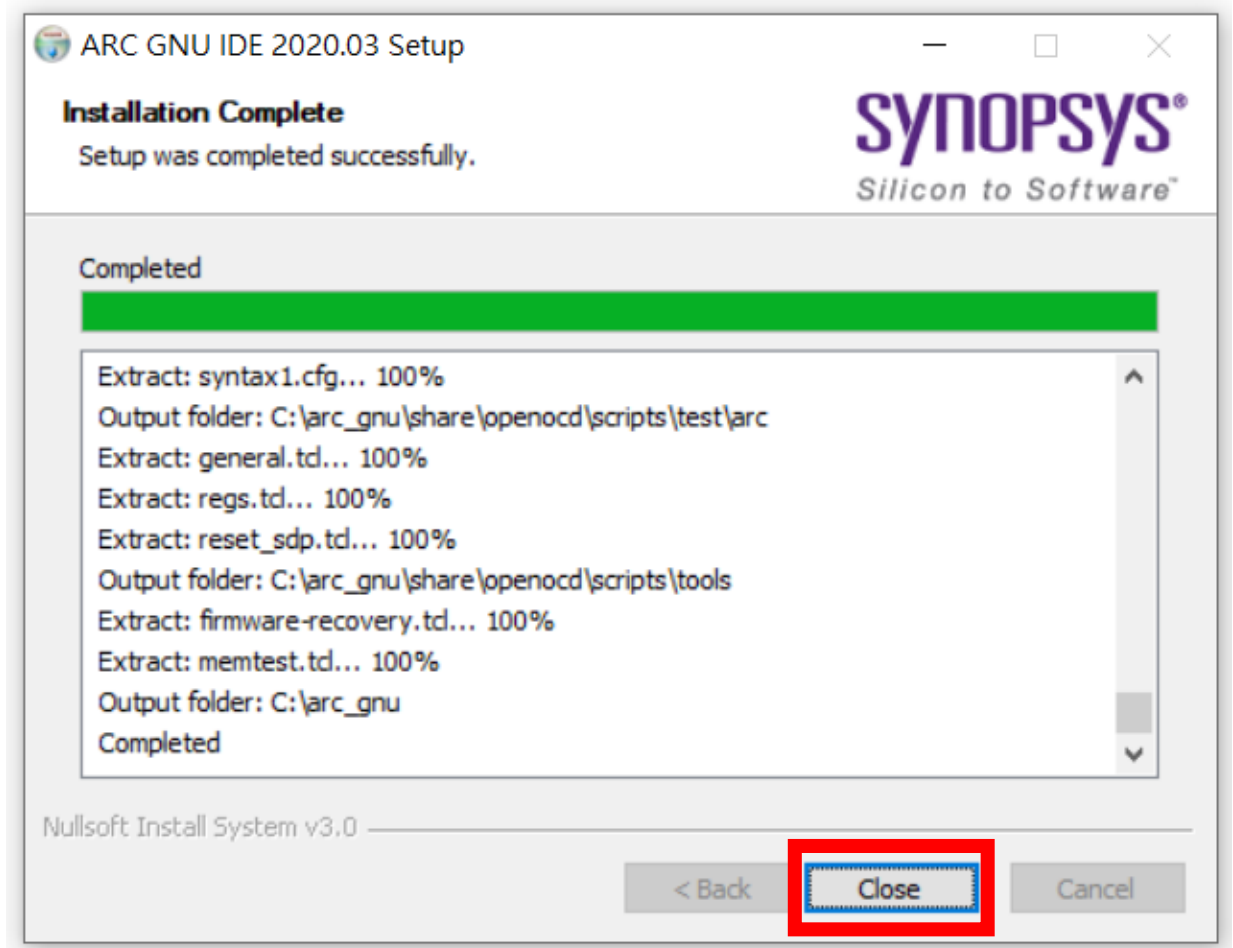
Installation

- GNU Tollchain



Installation - GNU Toolchain

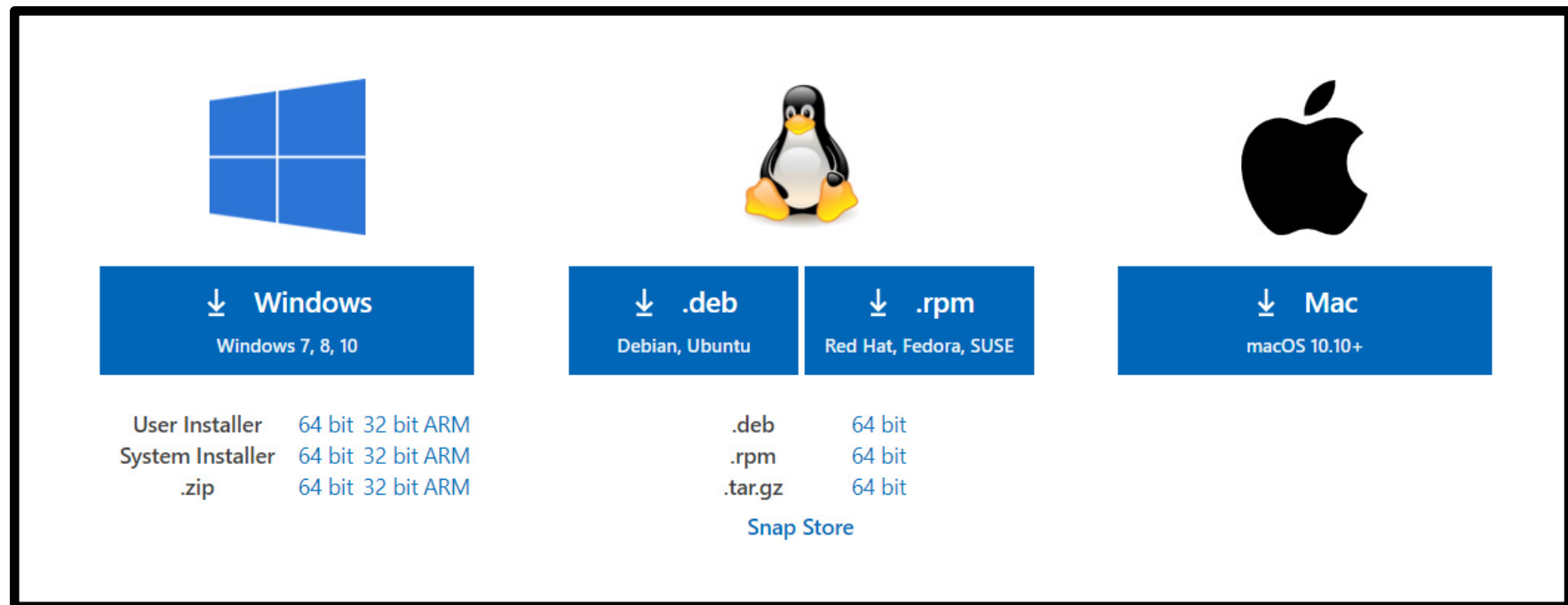
Done !



Installation - VSCode

Download Link :

<https://code.visualstudio.com/#alt-downloads>



The screenshot displays the download options for VS Code on different operating systems. It is organized into three main columns: Windows, Linux, and Mac.

- Windows:** Represented by the Windows logo. A blue button with a download icon and the text "Windows" and "Windows 7, 8, 10" is shown. Below it, a table lists installation methods and their supported architectures.
- Linux:** Represented by the Tux penguin logo. Two blue buttons are shown: one for ".deb" (Debian, Ubuntu) and one for ".rpm" (Red Hat, Fedora, SUSE). Below these, a table lists package formats and their supported architectures, along with the "Snap Store" option.
- Mac:** Represented by the Apple logo. A blue button with a download icon and the text "Mac" and "macOS 10.10+" is shown.

OS	Installation Method	Supported Architectures
Windows	User Installer	64 bit, 32 bit, ARM
	System Installer	64 bit, 32 bit, ARM
	.zip	64 bit, 32 bit, ARM
Linux	.deb	64 bit
	.rpm	64 bit
	.tar.gz	64 bit
Linux	Snap Store	-
Mac	Mac	macOS 10.10+

Installation
- VSCode

Please learn how to use **VSCode** by
yourself on the Internet

ex : 改中文介面、常用快捷鍵 ... etc

簡單了解即可

Installation

- Putty

Download Link :

<https://www.puttygen.com/download-putty>

Please follow the steps to install !

USB Driver Adjustment - Download Zadig

Download Link :

<https://zadig.akeo.ie>

Download

Updated 2020.03.28:

- **Zadig 2.5** (4.9 MB)

- [Other versions](#)

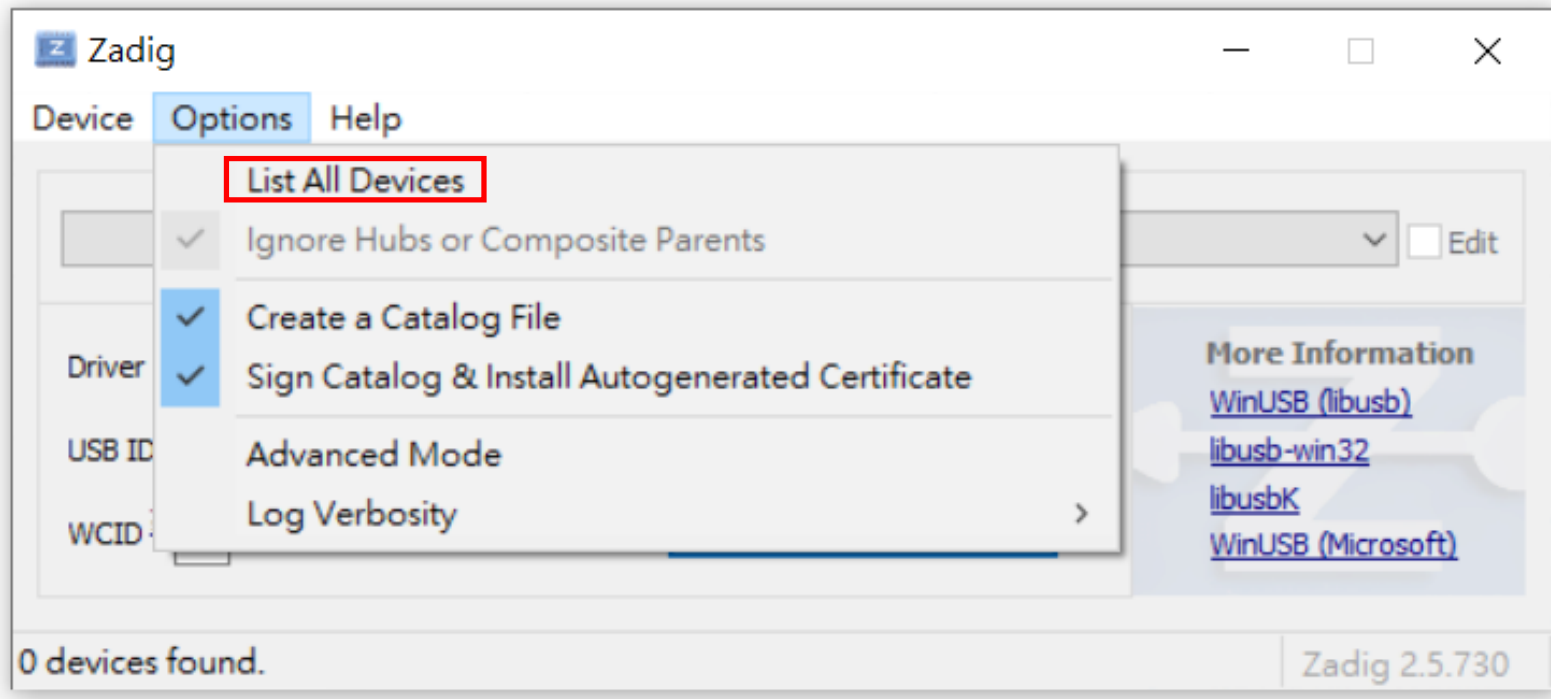
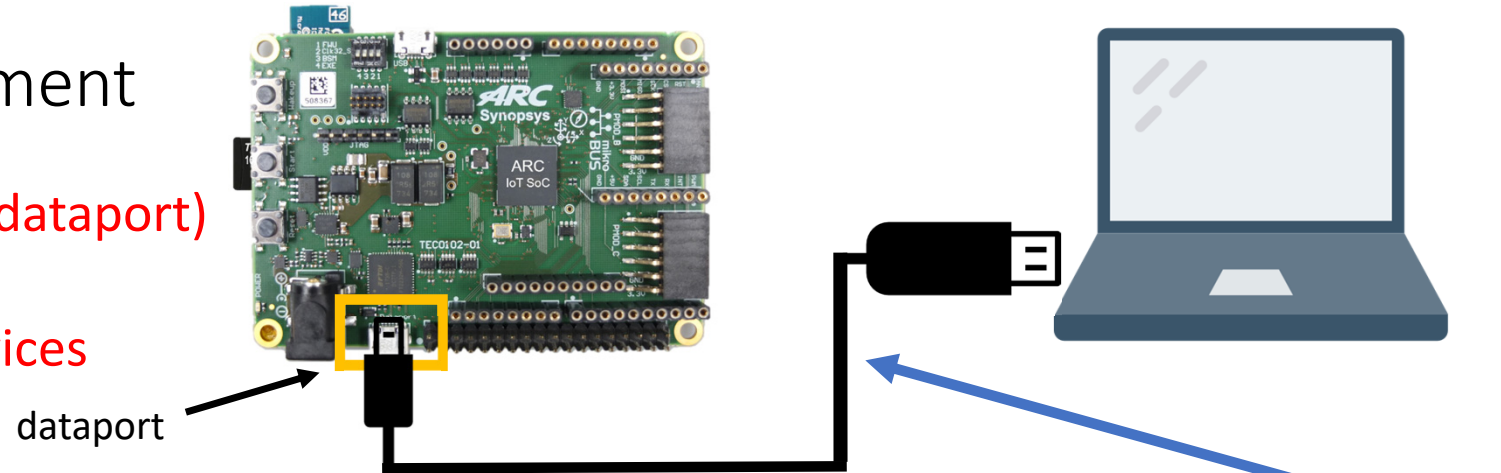
System Requirements:

Windows 7 or later.

Windows XP and Windows Vista are **NO LONGER SUPPORTED**.

USB Driver Adjustment

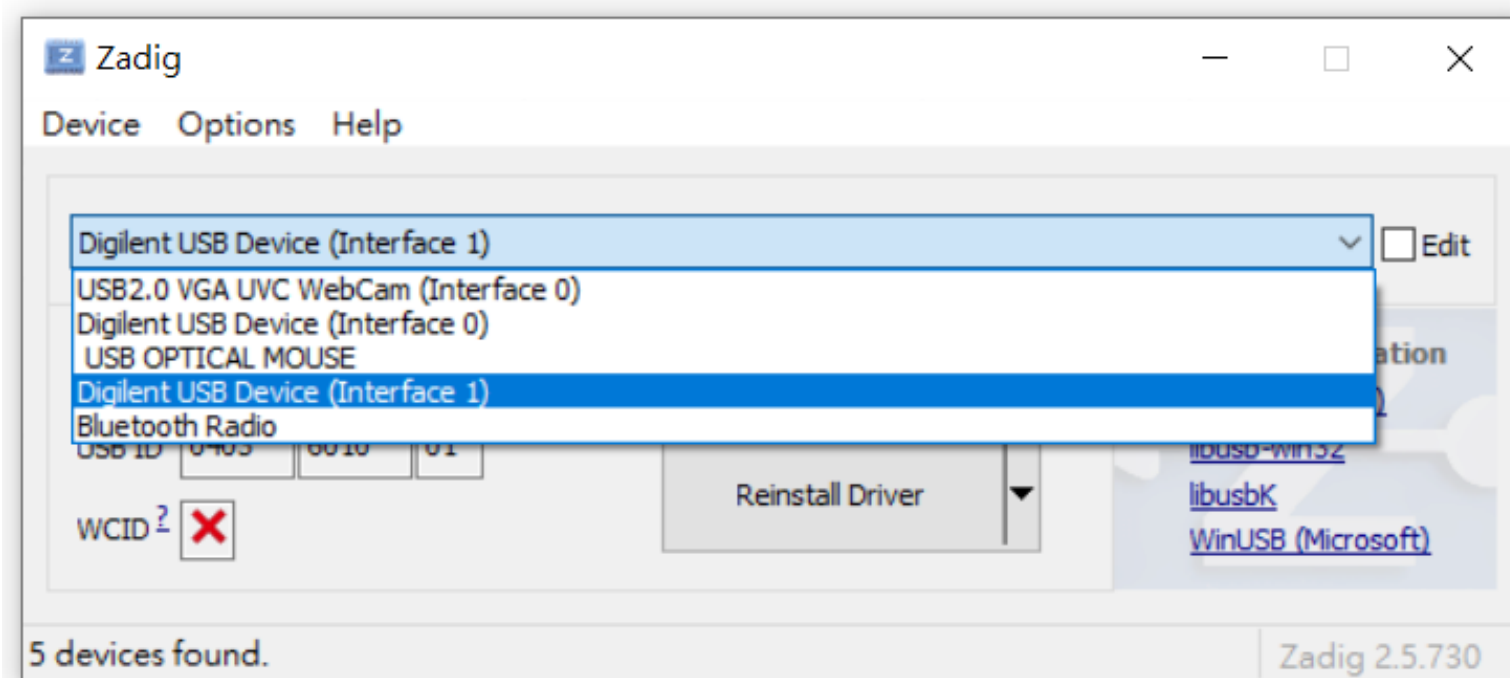
1. Connect PC & IoTDK (dataport)
2. Open Zadig-2.5.exe
3. Options -> List All Devices



USB to microUSB

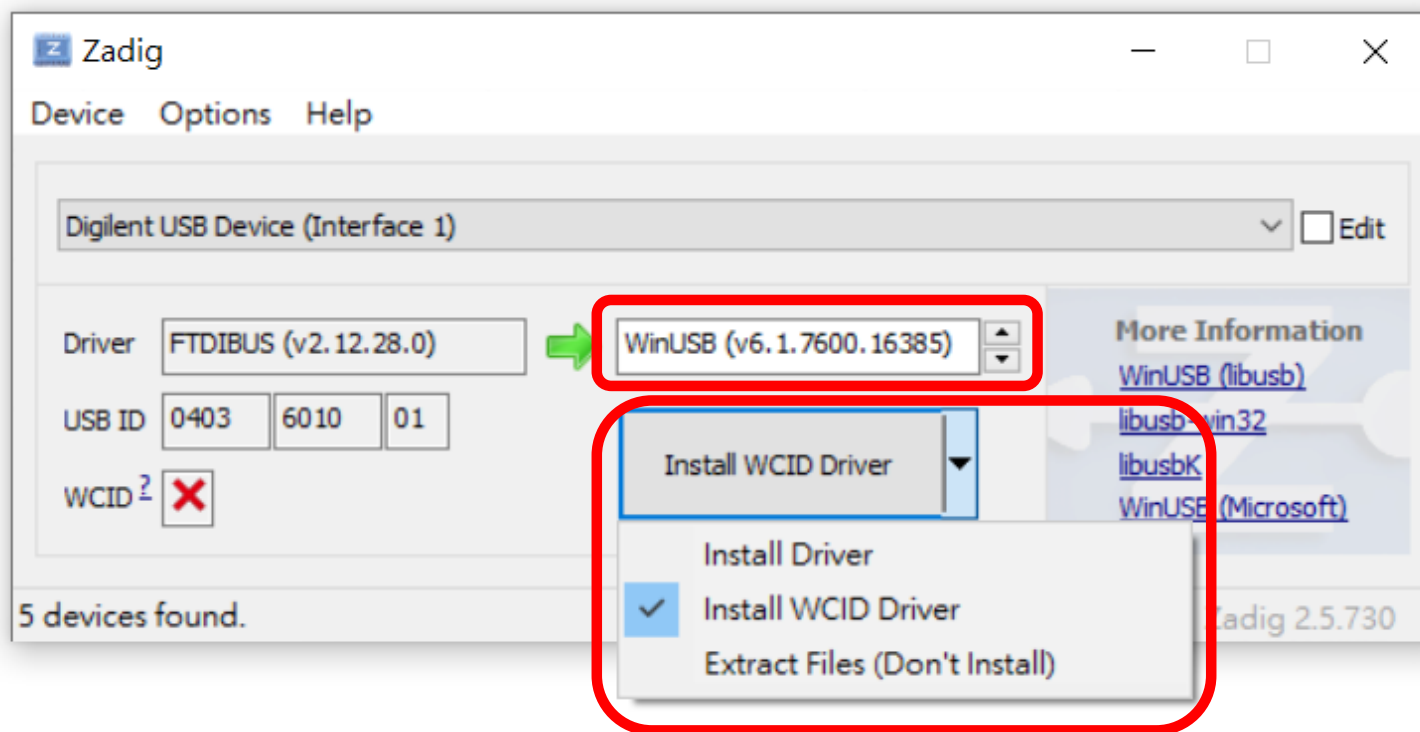
USB Driver Adjustment

Choose Digilent USB Device(Interface 1)



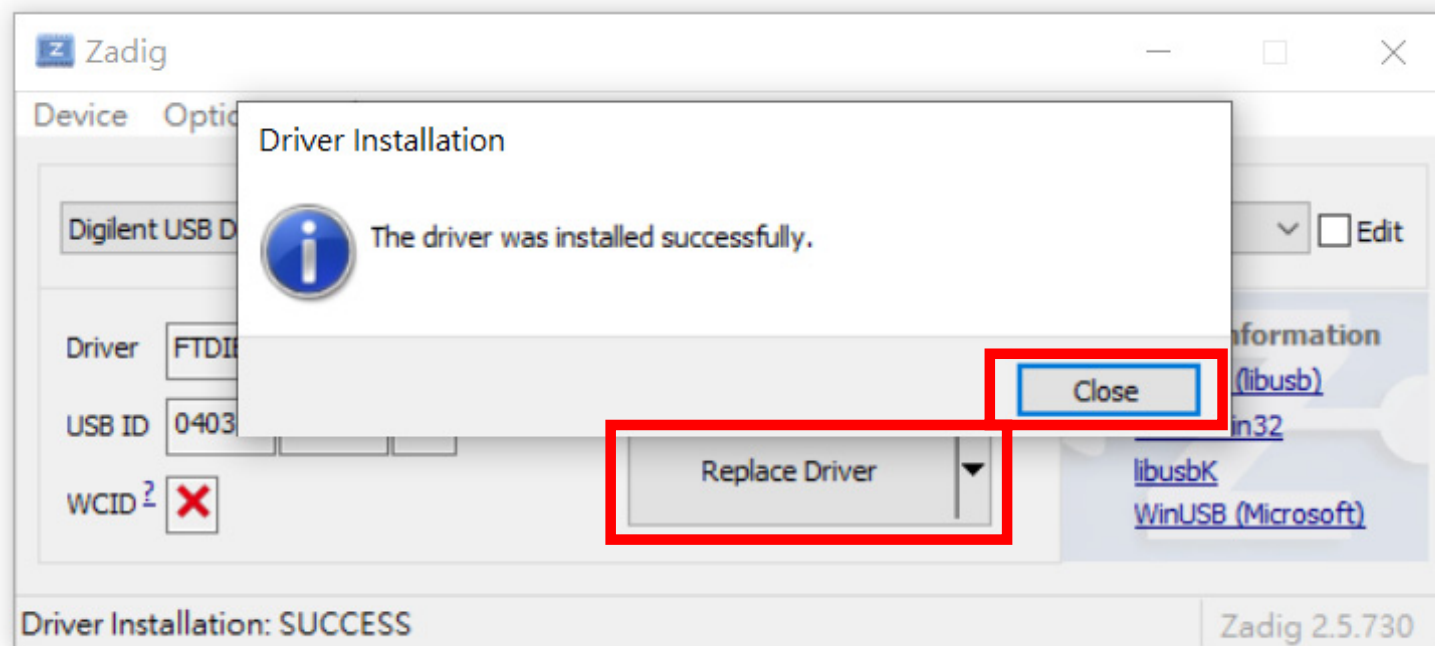
USB Driver Adjustment

1. Choose WinUSB (v6.1.7600.16385)
2. Install Driver or Install WCID Driver
(※if you have installed this driver before, it will display Replace Driver, just click it)
3. Then, wait a second



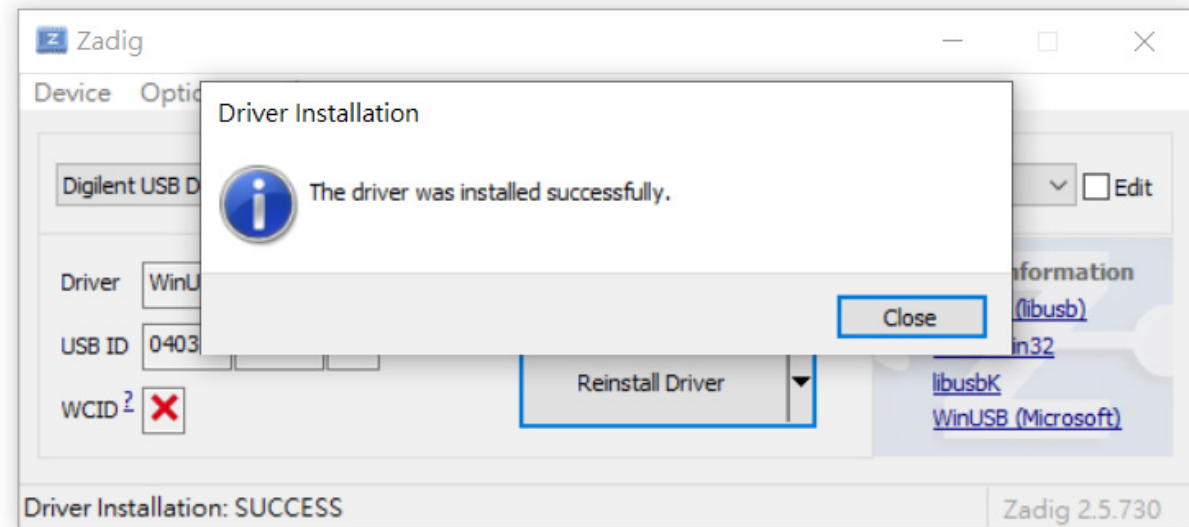
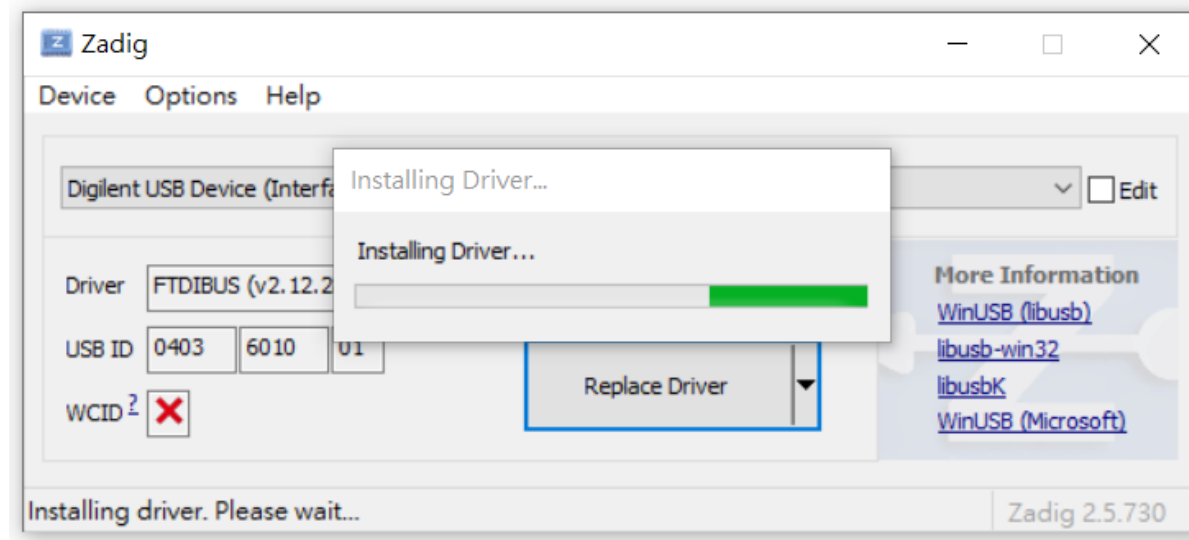
USB Driver Adjustment

1. Click Close
2. Click Replace Driver



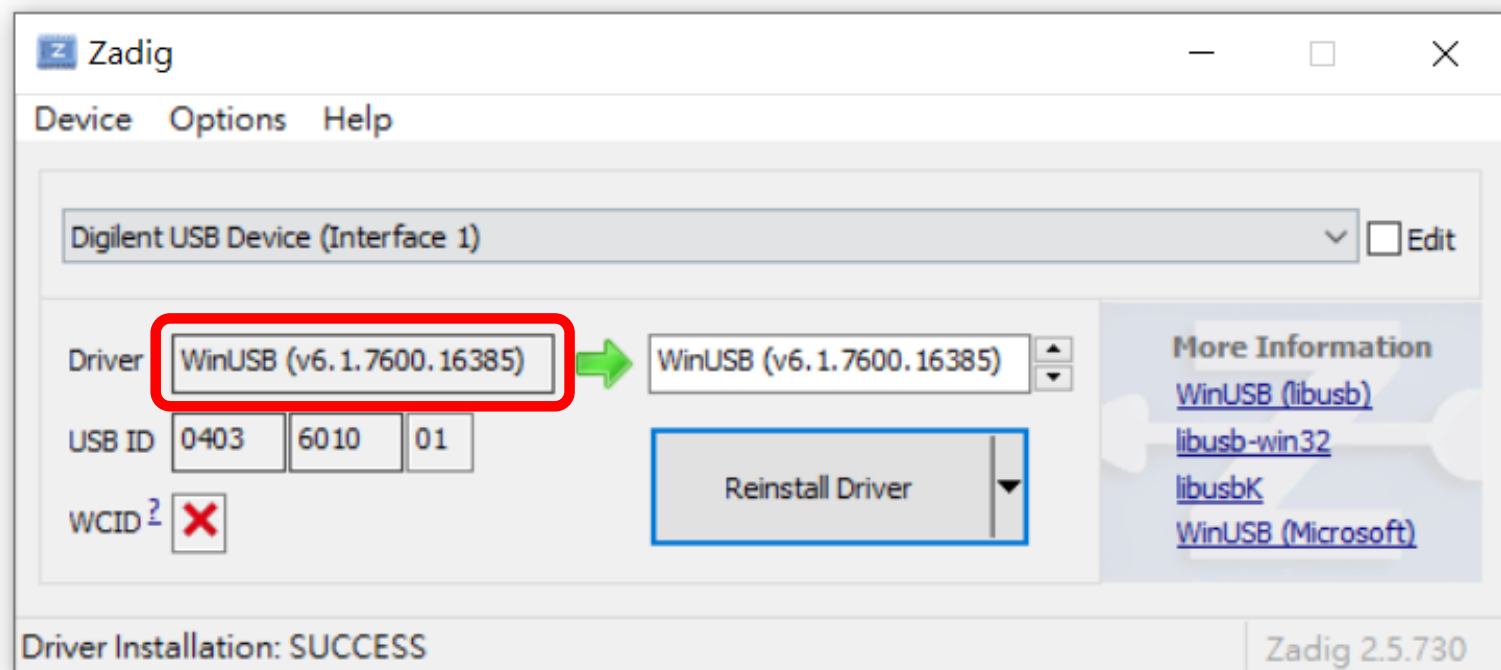
USB Driver Adjustment

Wait a second



USB Driver Adjustment

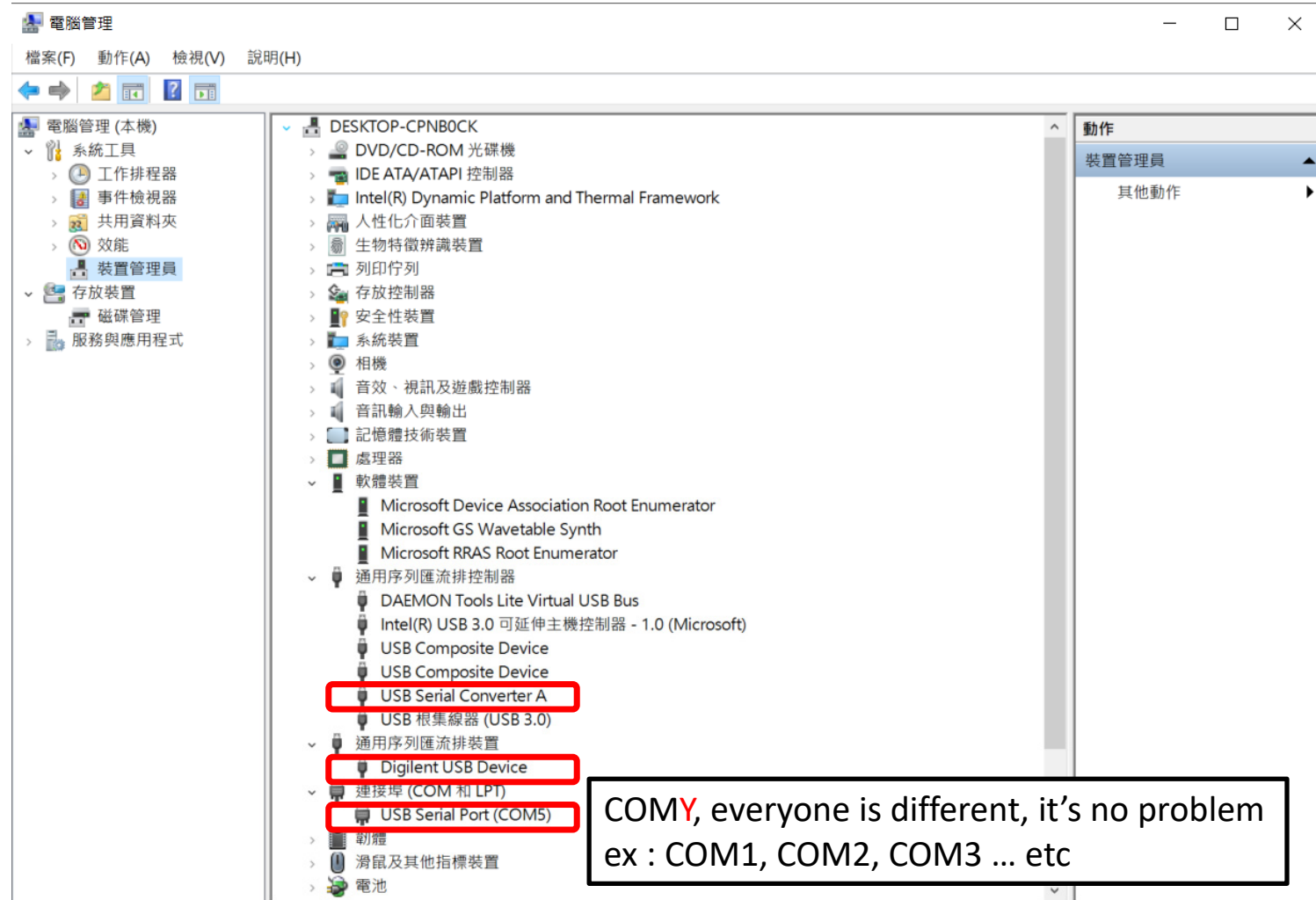
Then you can find Driver is changed to WinUSB
Now, please restart your pc



USB Driver Adjustment

After restart,
you can check your USB
Driver in “Device Manager”

Done !



Test

- Download embarc_osp

Download Link :

https://github.com/foss-for-synopsys-dwc-arc-processors/embarc_osp

The screenshot shows the GitHub repository page for 'foss-for-synopsys-dwc-arc-processors/embarc_osp'. The repository is currently on the 'master' branch, with 8 branches and 14 tags. A dropdown menu is open under the 'Code' button, showing options to clone with HTTPS, use SSH, or open with GitHub Desktop. The 'Download ZIP' option is highlighted with a red box and labeled '2. Click here'. A red arrow points to the 'Code' button, labeled '1. Click here'. The repository description is 'embARC Open Software Platform (OSP) - An embedded software distribution for IoT and other embedded applications for ARC'. The repository has 28 watchers, 35 stars, and 44 forks. The 'About' section includes the website 'www.embarc.org/' and tags for 'iot-platform', 'iot-framework', 'arc', 'iot', and 'synopsys'. The 'Releases' section shows 14 releases, with the latest release being 'embARC Open Software Platf...' on 3 Sep 2019.

1. Click here

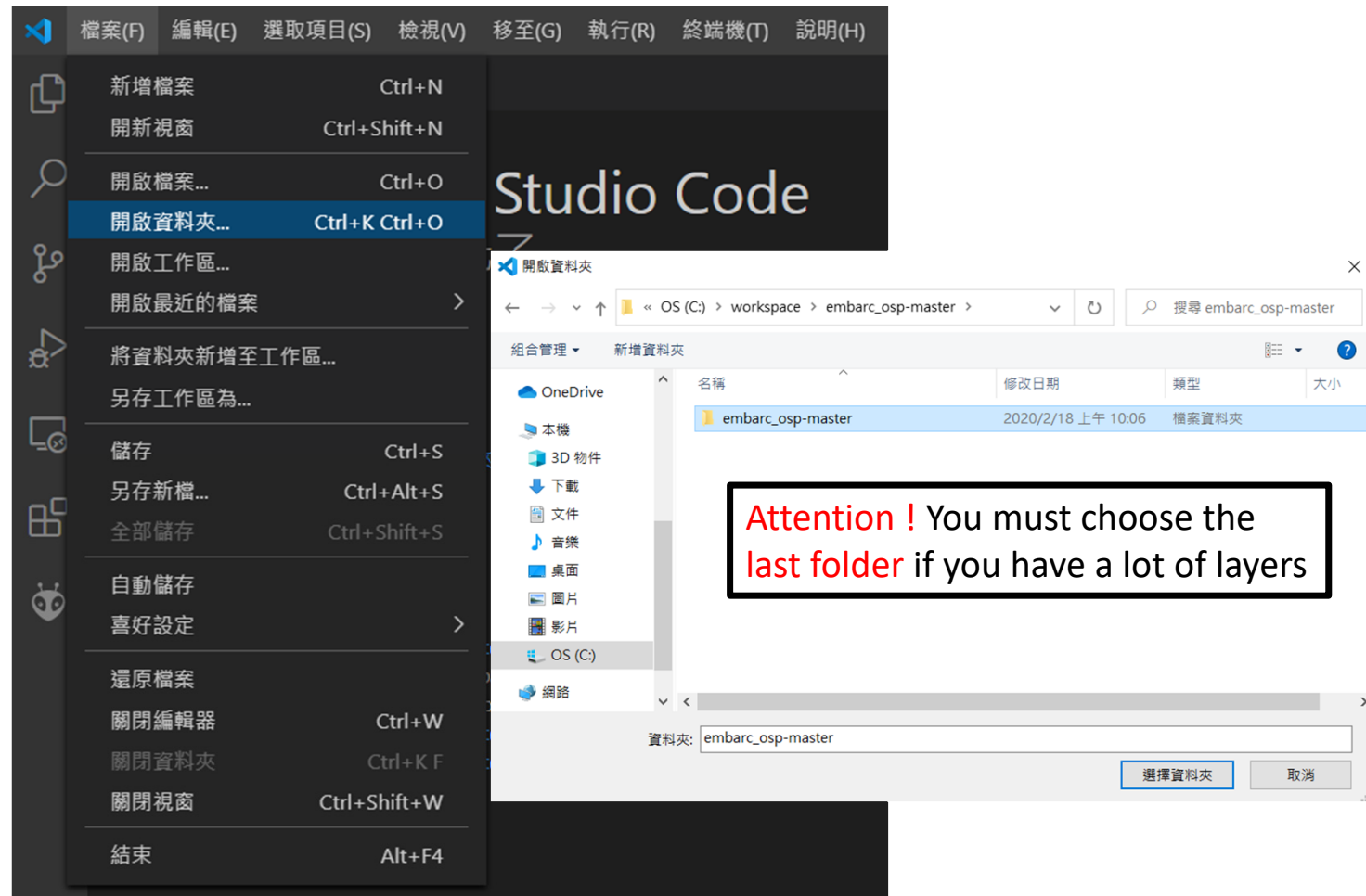
2. Click here

Test

1. unzip embarc_osp-master.zip
2. Open with VSCode

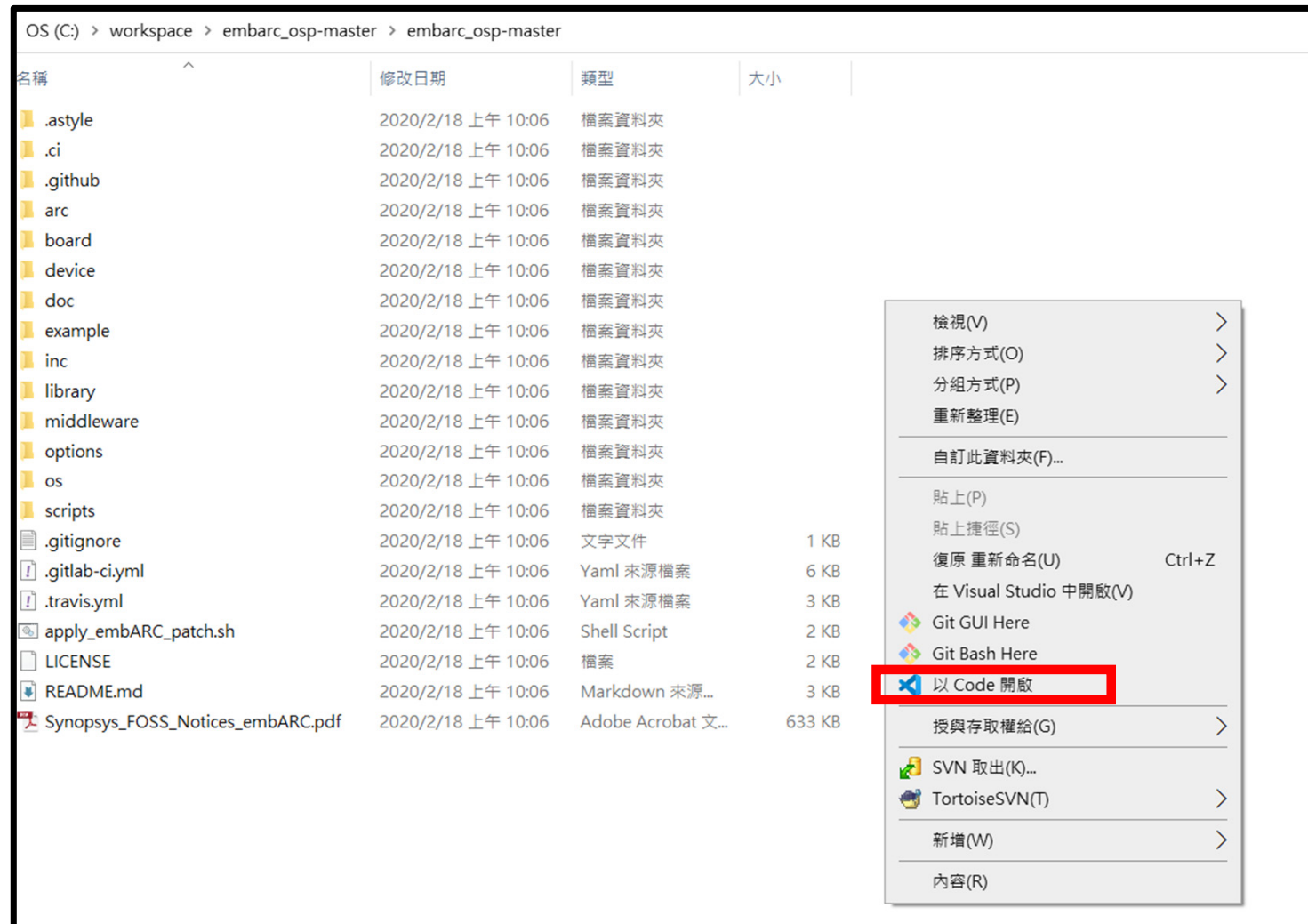
Open with VSCode

First way :



Test

Open with VSCode
Second way :



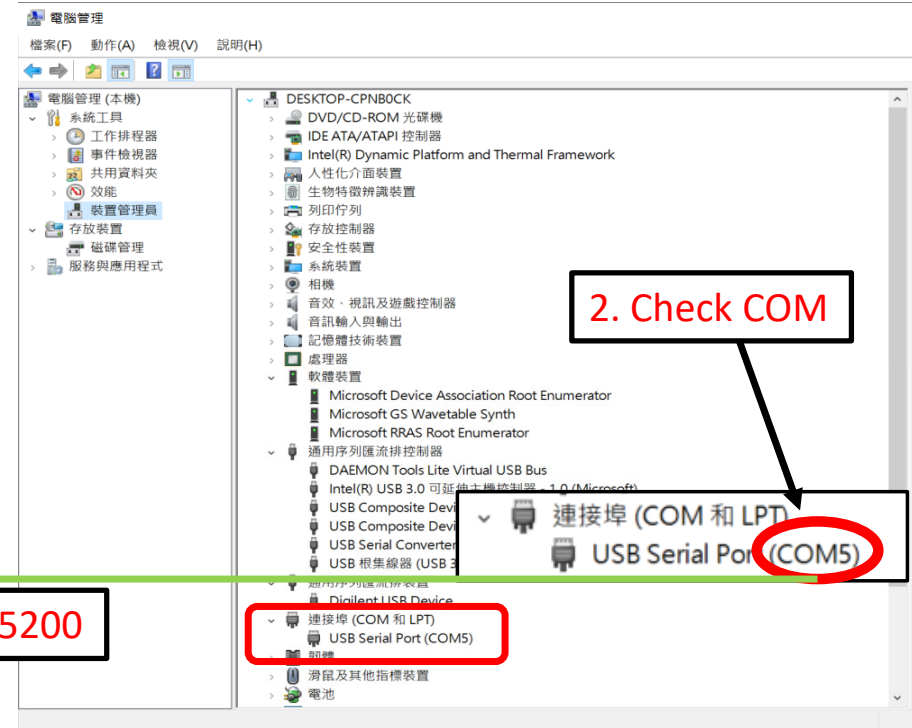
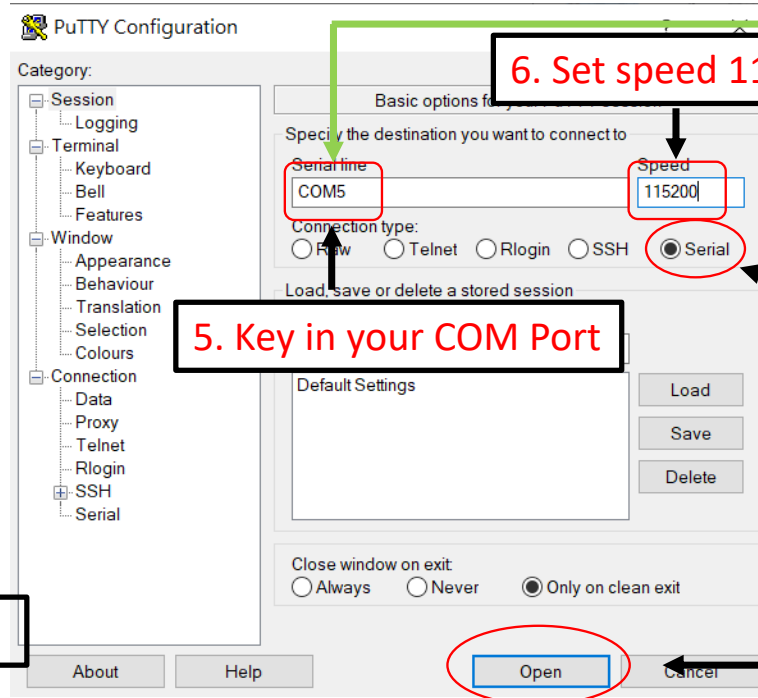
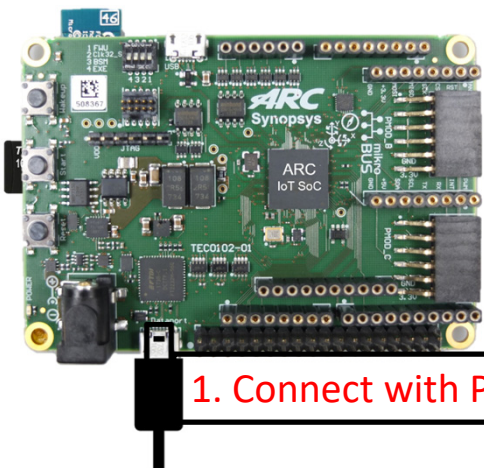
Test

This is begin page, you can see embarc_osp file at left side



Test

1. Connect PC & IoTDK (dataport)
2. Check COM Port Number in "Device Manager"
3. Open Putty
4. Click Serial
5. Key in your COM Port Number
6. Set Speed to 115200
7. Click Open



1. Connect with PC

5. Key in your COM Port

6. Set speed 115200

4. Click Serial

7. Click Open

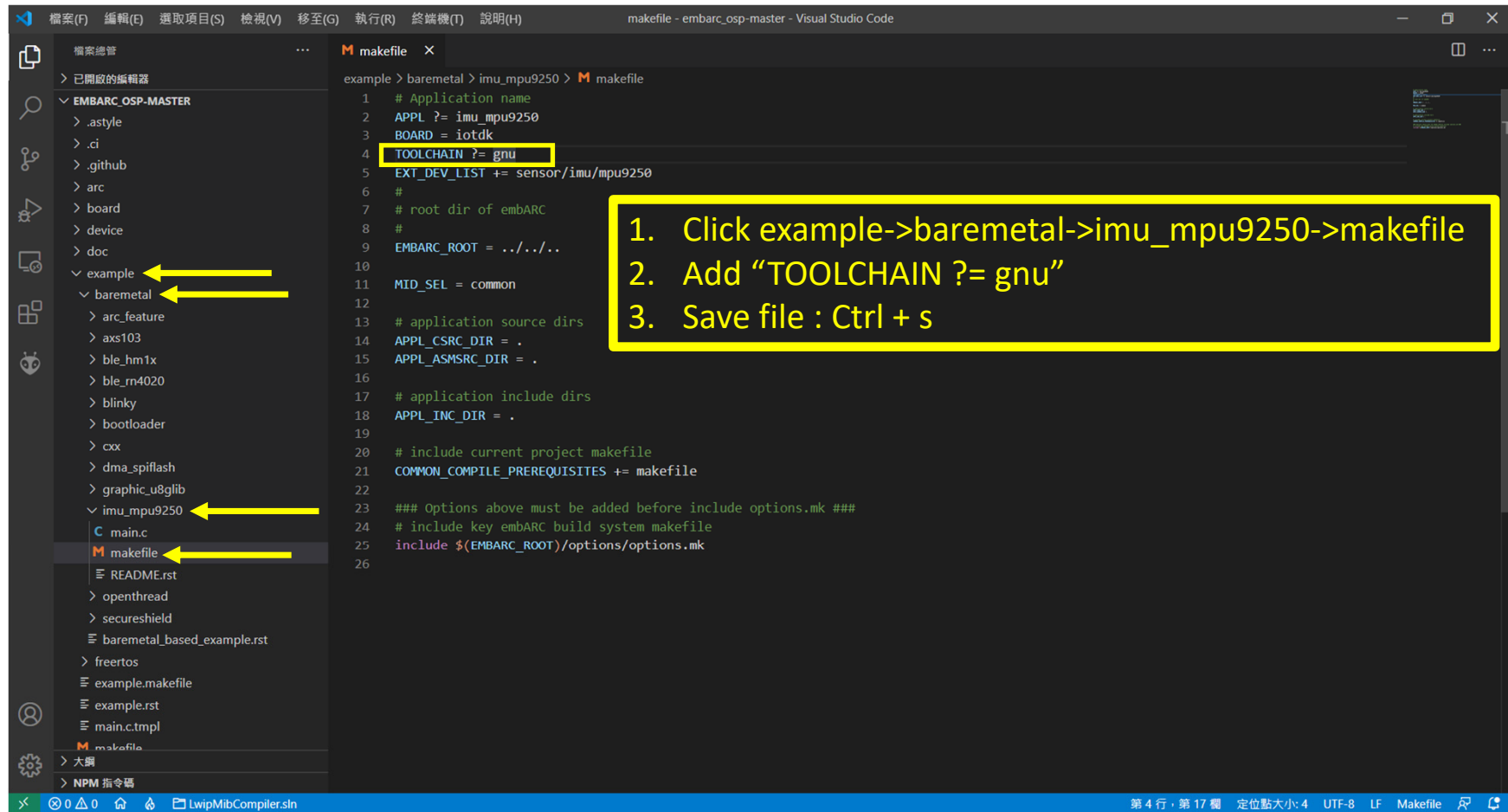
2. Check COM

Test

You will see this window
Then turn back to VSCode



Test



The screenshot shows the Visual Studio Code interface with a file explorer on the left and a code editor on the right. The file explorer shows a project structure with a folder named 'example' containing a sub-folder 'baremetal', which in turn contains a folder 'imu_mpu9250'. The 'makefile' file in the 'imu_mpu9250' folder is selected. The code editor shows the content of the 'makefile' file, with the line 'TOOLCHAIN ?= gnu' highlighted in yellow. A yellow box with a black border contains the following instructions:

1. Click example->baremetal->imu_mpu9250->makefile
2. Add "TOOLCHAIN ?= gnu"
3. Save file : Ctrl + s

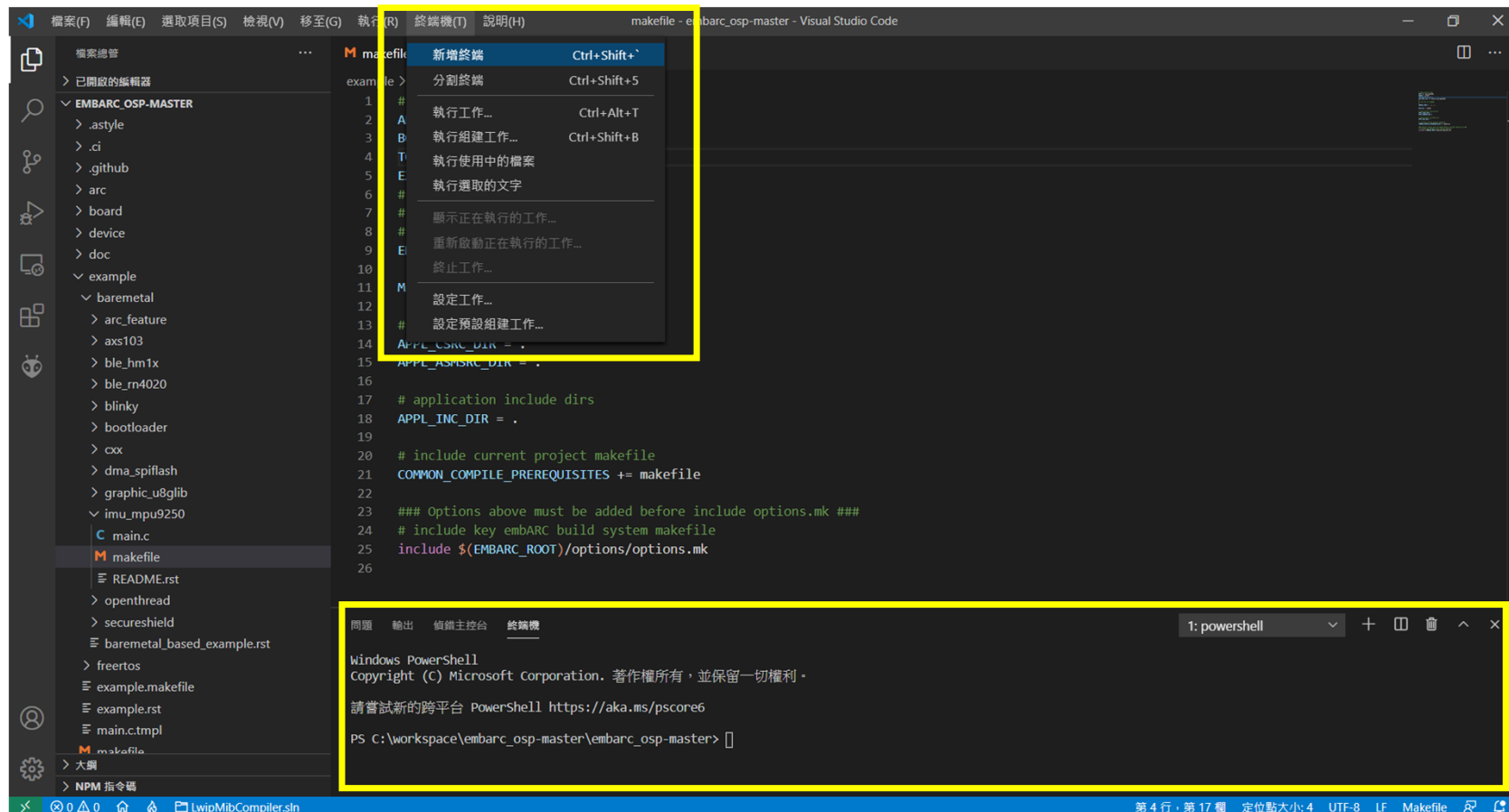
```
example > baremetal > imu_mpu9250 > M makefile
1  # Application name
2  APPL ?= imu_mpu9250
3  BOARD = iotdk
4  TOOLCHAIN ?= gnu
5  EXT_DEV_LIST += sensor/imu/mpu9250
6  #
7  # root dir of embARC
8  #
9  EMBARC_ROOT = ../../..
10
11 MID_SEL = common
12
13 # application source dirs
14 APPL_CSRC_DIR = .
15 APPL_ASMSRC_DIR = .
16
17 # application include dirs
18 APPL_INC_DIR = .
19
20 # include current project makefile
21 COMMON_COMPILE_PREREQUISITES += makefile
22
23 ### Options above must be added before include options.mk ###
24 # include key embARC build system makefile
25 include $(EMBARC_ROOT)/options/options.mk
26
```

At the bottom of the window, the status bar shows: 第 4 行 · 第 17 欄 定位點大小: 4 UTF-8 LF Makefile

Test

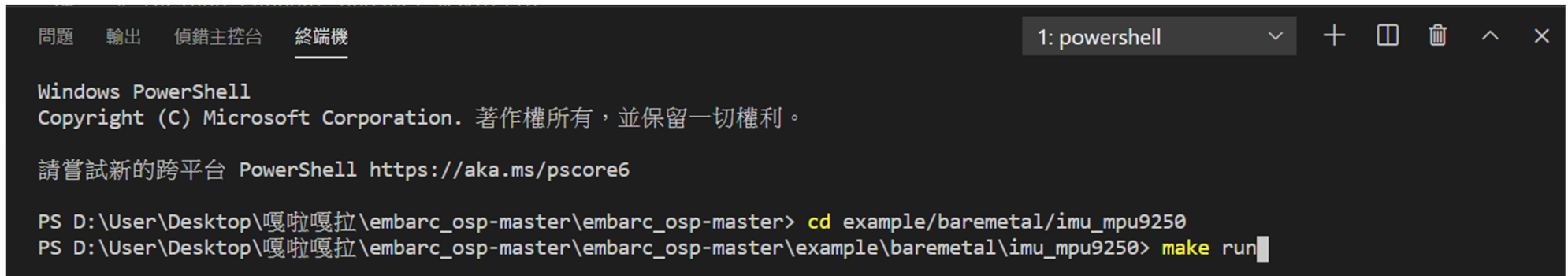
Hot Key : Ctrl + Shift + `

Then you can find a new terminal appearing at bottom



Test

1. Key in “cd example/baremetal/imu_mpu9250”
2. Key in “make run”
3. Wait a second, it is compiling file and preparing to run now



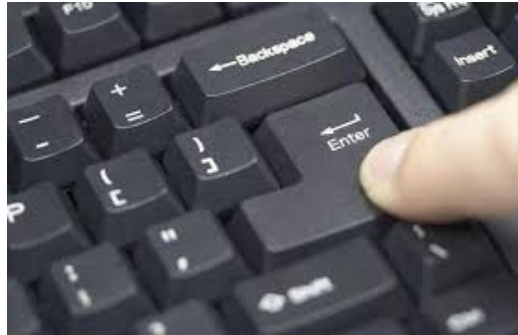
```
問題 輸出 偵錯主控台 終端機 1: powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. 著作權所有，並保留一切權利。

請嘗試新的跨平台 PowerShell https://aka.ms/pscore6

PS D:\User\Desktop\嘎啦嘎拉\embarc_osp-master\embarc_osp-master> cd example/baremetal/imu_mpu9250
PS D:\User\Desktop\嘎啦嘎拉\embarc_osp-master\embarc_osp-master\example\baremetal\imu_mpu9250> make run
```

Test

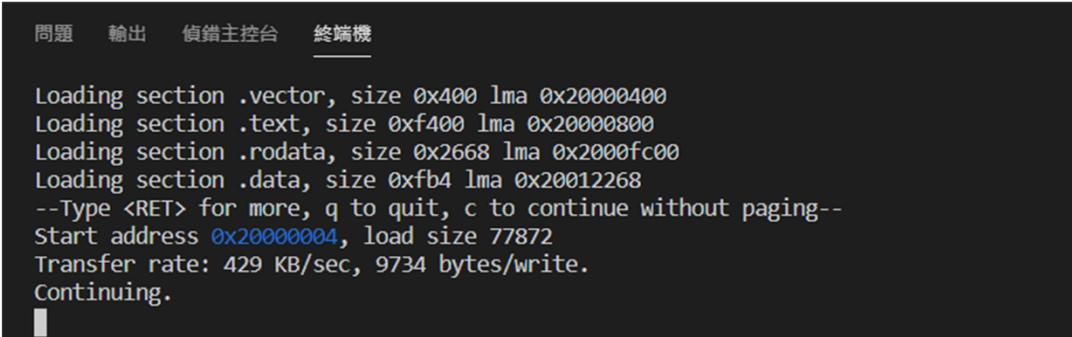
1. If you see the line
--Type <RET> for more, q to quit, c to continue without paging--
just key in “Enter”



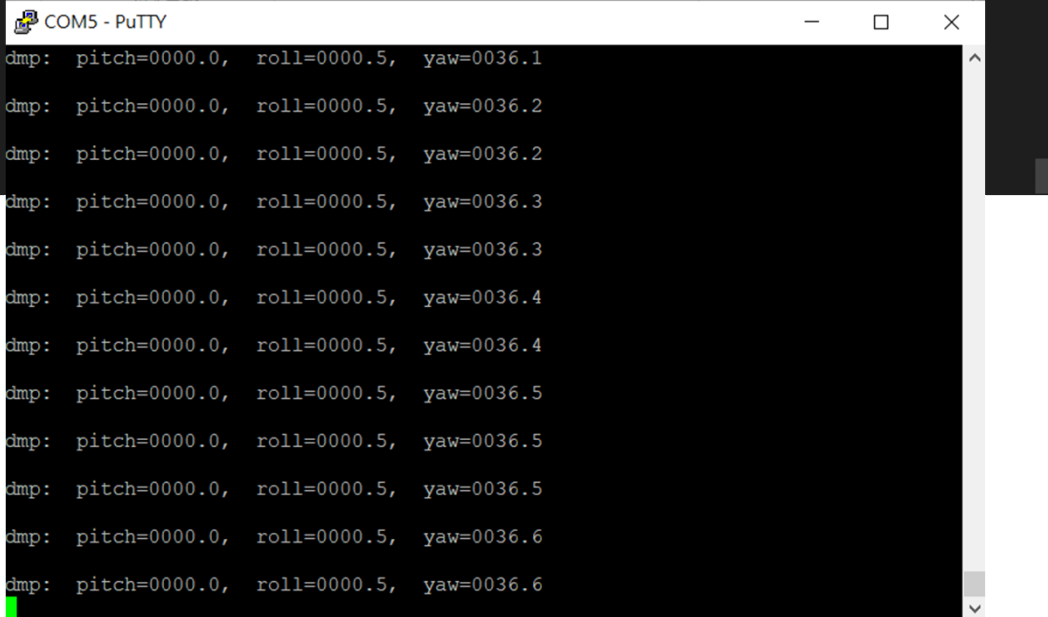
```
問題 輸出 偵錯主控台 終端機 1: gmake
GNU gdb (ARCompact/ARCV2 ISA elf32 toolchain 2020.03) 10.0.50.20200307-git
Copyright (C) 2020 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "--host=i686-w64-mingw32 --target=arc-elf32".
Type "show configuration" for configuration details.
--Type <RET> for more, q to quit, c to continue without paging--
```

Test

After a lot of “Enter” (about two or three), you will see “Continuing.”
Then, Open Serial COM Port window, you will see like below



```
問題 輸出 偵錯主控台 終端機
1: gmake
Loading section .vector, size 0x400 lma 0x20000400
Loading section .text, size 0xf400 lma 0x20000800
Loading section .rodata, size 0x2668 lma 0x2000fc00
Loading section .data, size 0xfb4 lma 0x20012268
--Type <RET> for more, q to quit, c to continue without paging--
Start address 0x20000004, load size 77872
Transfer rate: 429 KB/sec, 9734 bytes/write.
Continuing.
```



```
COM5 - PuTTY
dmp: pitch=0000.0, roll=0000.5, yaw=0036.1
dmp: pitch=0000.0, roll=0000.5, yaw=0036.2
dmp: pitch=0000.0, roll=0000.5, yaw=0036.2
dmp: pitch=0000.0, roll=0000.5, yaw=0036.3
dmp: pitch=0000.0, roll=0000.5, yaw=0036.3
dmp: pitch=0000.0, roll=0000.5, yaw=0036.4
dmp: pitch=0000.0, roll=0000.5, yaw=0036.4
dmp: pitch=0000.0, roll=0000.5, yaw=0036.5
dmp: pitch=0000.0, roll=0000.5, yaw=0036.5
dmp: pitch=0000.0, roll=0000.5, yaw=0036.5
dmp: pitch=0000.0, roll=0000.5, yaw=0036.6
dmp: pitch=0000.0, roll=0000.5, yaw=0036.6
```

If you see a lot of “dmp: “ is printed in the window, this is that you succeed !

If not, please ask TA for help
Thanks ~

Test

If you want to stop the program

1. Go back to terminal
2. Hot Key : Ctrl + c, then you will see (gdb)

```
問題 輸出 偵錯主控台 終端機 1: gmake + □ □ ^ ×
Loading section .data, size 0xfb4 lma 0x20012268
--Type <RET> for more, q to quit, c to continue without paging--
Start address 0x20000004, load size 77872
Transfer rate: 429 KB/sec, 9734 bytes/write.
Continuing.

Program received signal SIGINT, Interrupt.
0x2000ea74 in __udivdi3 ()
(gdb) █
```

3. Key in “q”

```
(gdb) q
```

4. Key in “y”

```
Quit anyway? (y or n) y
```

5. Key in “y”

```
要終止批次工作嗎 (Y/N)? y
```

6. You will see bellow represented that you stop the program succeed !

```
gmake: *** [run] Error 255
PS C:\workspace\embarc_osp-master\embarc_osp-master\example\baremetal\imu_mpu9250> █
```

Done !

If you have any question,
please ask TA for help ~