

Tutorial 2 – ARC EM9D AIoT DK Project Environment Setup & Development Flow

Willie Tsai
2023/02/06

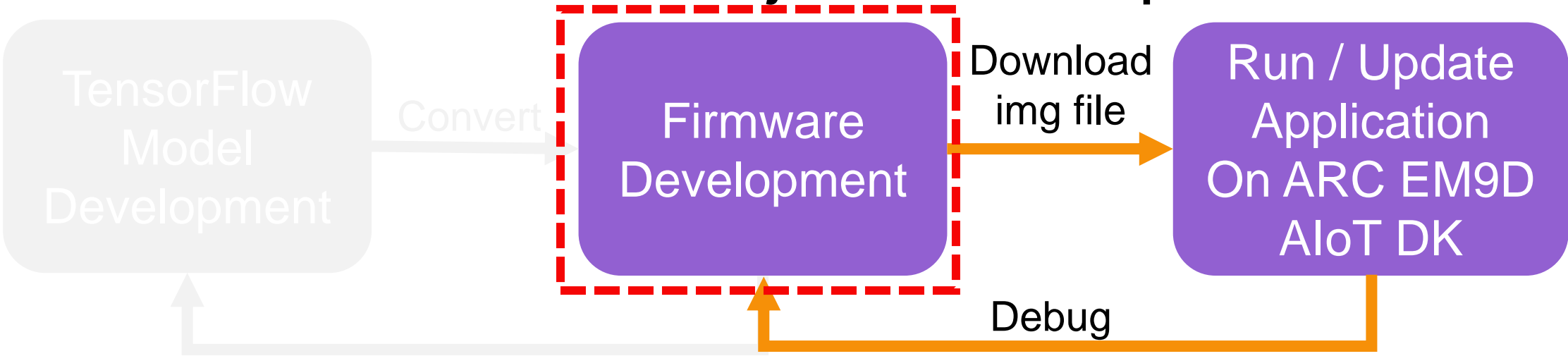


ARC EM9D AIoT DK Project Development Flow



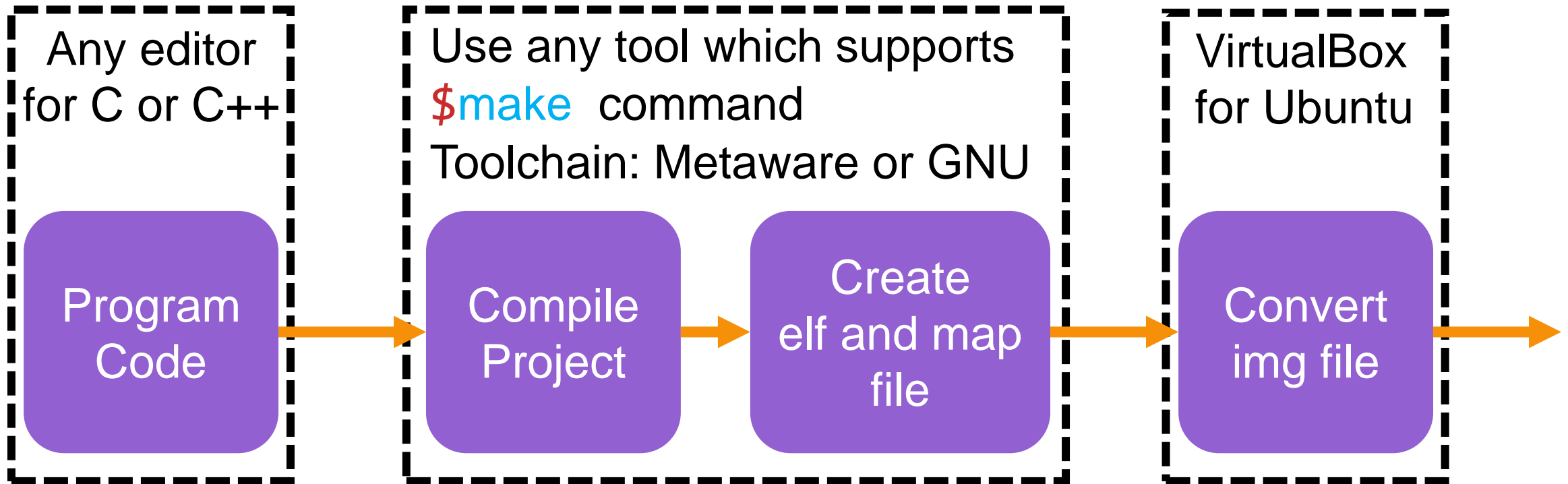
Stage	TensorFlow Model Development	Firmware Development	Run / Update Application On ARC EM9D AIoT DK
Tool	Anaconda Cygwin	Cygwin Metaware or ARC GNU VirtualBox (Ubuntu 20.04)	JTAG Himax-FT4222-GUI USB Cable
Language	Python 3	C language C++ language	

ARC EM9D AIoT DK Project Development Flow



Stage	TensorFlow Model Development	Firmware Development	Run / Update Application On ARC EM9D AIoT DK
Tool	Anaconda Cygwin	Cygwin Metaware or ARC GNU VirtualBox (Ubuntu 20.04)	JTAG Himax-FT4222-GUI USB Cable
Language	Python 3	C language C++ language	

Firmware Development



Download and Setup SDK



Download and Setup SDK

1. Open Cygwin64 Terminal

\$ cd c:

\$ cd Users/{username}/ (to your working file path)

\$ mkdir VM (Suggest create a new folder named “VM”)

\$ cd VM

A screenshot of a Cygwin64 terminal window. The title bar shows the path /cygdrive/c/Users/williet/VM. The terminal content shows the following sequence of commands and prompts: williet@WILLIET-7490 ~, \$ cd c:, williet@WILLIET-7490 /cygdrive/c, \$ cd Users/williet/, williet@WILLIET-7490 /cygdrive/c/Users/williet, \$ mkdir VM, williet@WILLIET-7490 /cygdrive/c/Users/williet, \$ cd VM, williet@WILLIET-7490 /cygdrive/c/Users/williet/VM, and a final \$ prompt.

```
williet@WILLIET-7490 ~  
$ cd c:  
  
williet@WILLIET-7490 /cygdrive/c  
$ cd Users/williet/  
  
williet@WILLIET-7490 /cygdrive/c/Users/williet  
$ mkdir VM  
  
williet@WILLIET-7490 /cygdrive/c/Users/williet  
$ cd VM  
  
williet@WILLIET-7490 /cygdrive/c/Users/williet/VM  
$
```

Download and Setup SDK

2. Download SDK and unzip to folder “*C:\Users\{username}\VM*”
Please contact Synopsys Taiwan for the SDK

Download and Setup SDK

After these steps, your file structure will be like:

Synopsys_SDK_Vxx

|

---- *Example_Project*

|

---- *Labx (Firmware project)*

---- *Labx (Python project for TensorFlow Pproject)*

---- *LabPY (Python project for data convert)*

---- *others (Library, toolchain setting ...)*

---- *tools*

|

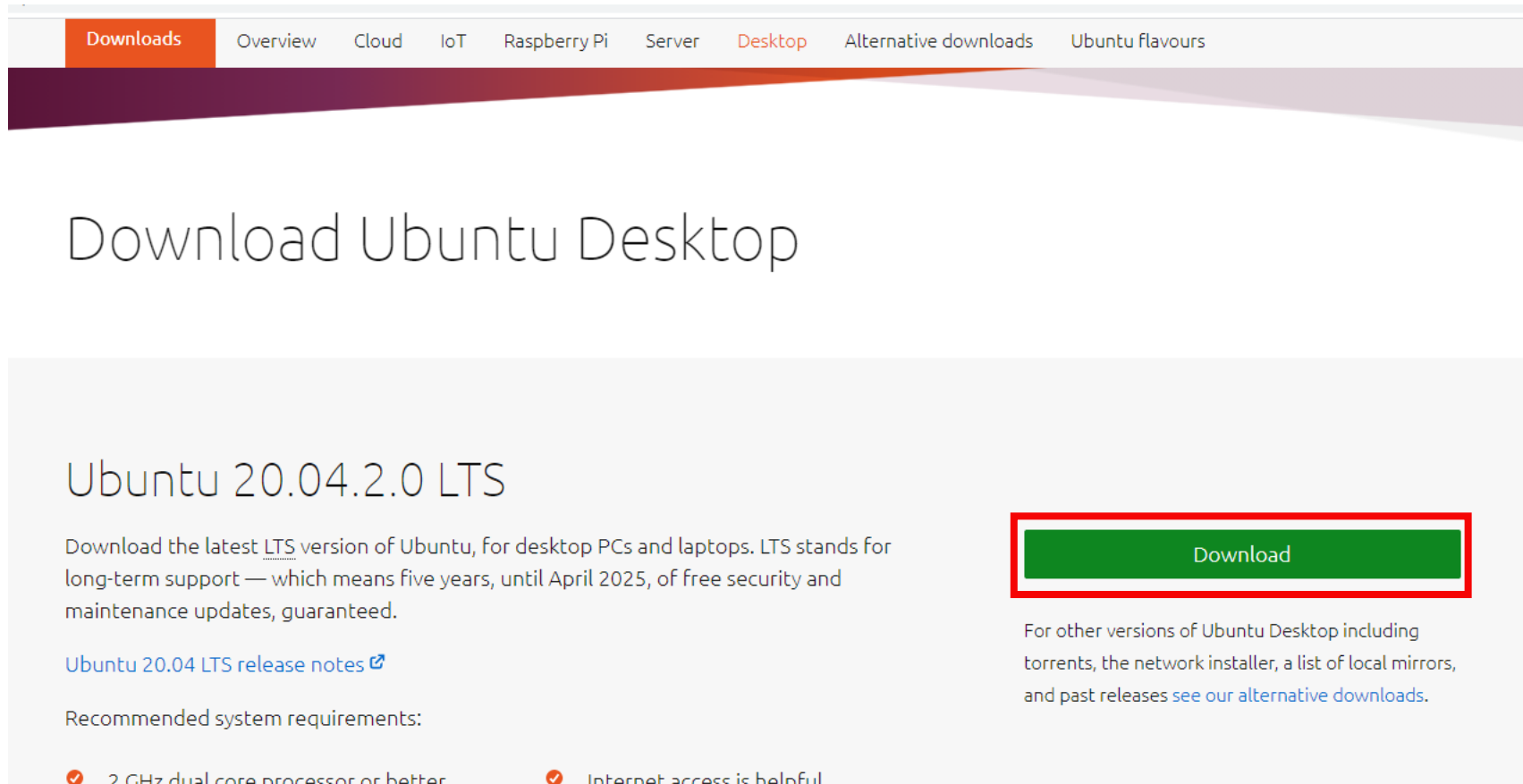
---- *image_gen_cstm (Convert elf and map file to image file)*

---- *HMX_FT4222H_GUI (Download image file to MCU)*

Download and Setup SDK

3. Download Ubuntu iso file

<https://ubuntu.com/download/desktop>



Download and Setup SDK

4. Open VM Oracle VirtualBox

5. New > Named your VM > select Linux, ubuntu (64-bit)



Name and operating system

Please choose a descriptive name and destination folder for the new virtual machine and select the type of operating system you intend to install on it. The name you choose will be used throughout VirtualBox to identify this machine.

Name:

Machine Folder:

Type:

Version:

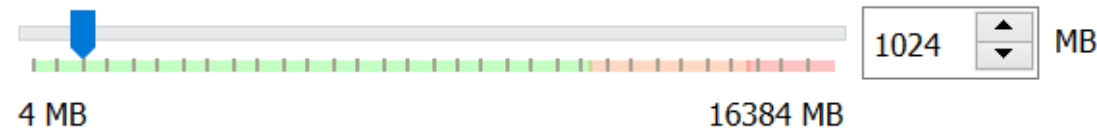
Download and Setup SDK

6. User can select your memory size according to your needs
Recommend more than 1024MB.

Memory size

Select the amount of memory (RAM) in megabytes to be allocated to the virtual machine.

The recommended memory size is **1024 MB**.



Next

Cancel

Download and Setup SDK

7. Click “Create” and “Next”

Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select one from the list or from another location using the folder icon.

If you need a more complex storage set-up you can skip this step and make the changes to the machine settings once the machine is created.

The recommended size of the hard disk is **10.00 GB**.

- ☐ Do not add a virtual hard disk
- ☒ Create a virtual hard disk now
- ☐ Use an existing virtual hard disk file

ub.vdi (Normal, 40.00 GB)



Create

Cancel

Hard disk file type

Please choose the type of file that you would like to use for the new virtual hard disk. If you do not need to use it with other virtualization software you can leave this setting unchanged.

- ☒ VDI (VirtualBox Disk Image)
- ☐ VHD (Virtual Hard Disk)
- ☐ VMDK (Virtual Machine Disk)

Expert Mode

Next

Cancel

Download and Setup SDK

7. User can select your hard disk size according to your needs Recommend 20~40 GB

Storage on physical hard disk

Please choose whether the new virtual hard disk file should grow as it is used (dynamically allocated) or if it should be created at its maximum size (fixed size).

A **dynamically allocated** hard disk file will only use space on your physical hard disk as it fills up (up to a maximum **fixed size**), although it will not shrink again automatically when space on it is freed.

A **fixed size** hard disk file may take longer to create on some systems but is often faster to use.

☒ Dynamically allocated

☐ Fixed size

Next

Cancel

File location and size

Please type the name of the new virtual hard disk file into the box below or click on the folder icon to select a different folder to create the file in.

C:\Users\williet\VirtualBox VMs\VM_ubuntu\VM_ubuntu.vdi



Select the size of the virtual hard disk in megabytes. This size is the limit on the amount of file data that a virtual machine will be able to store on the hard disk.

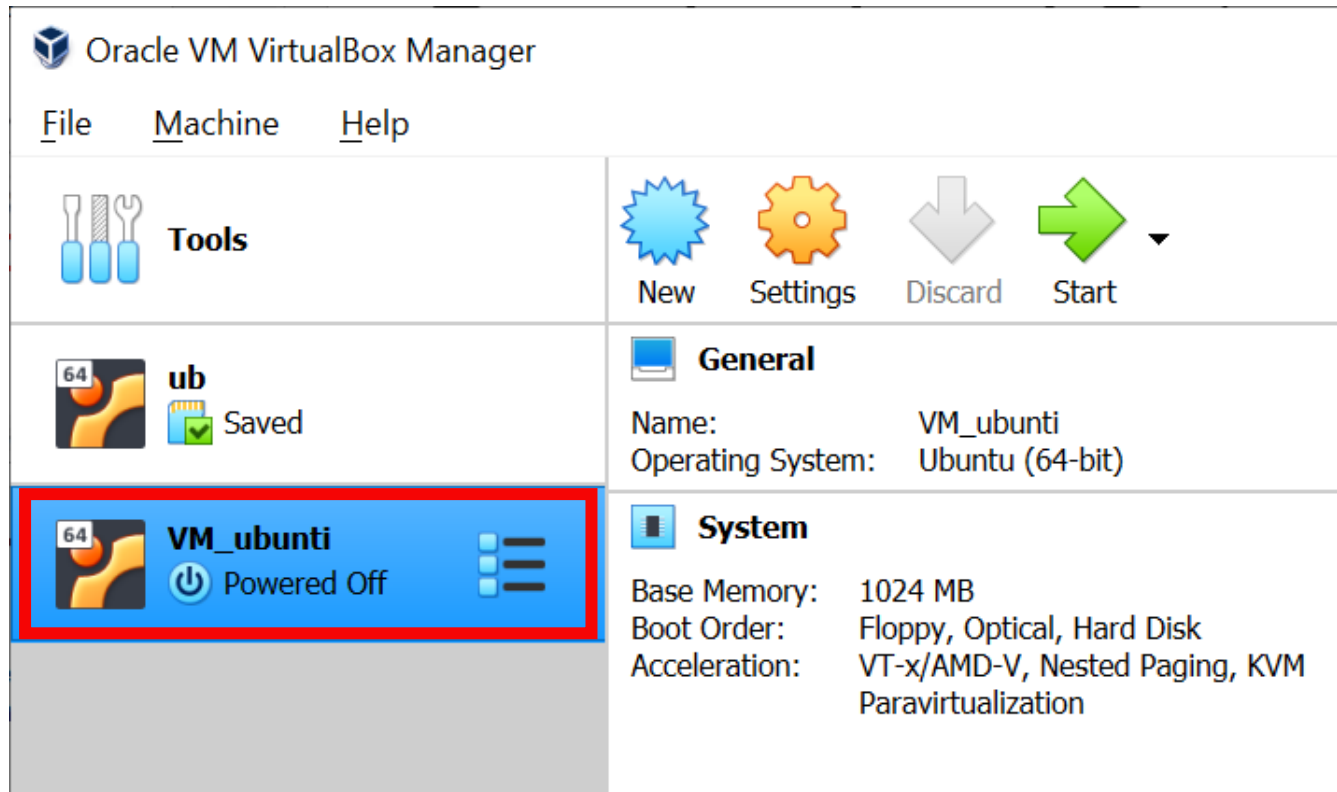


Create

Cancel

Download and Setup SDK

8. Start-up the VM and select the ISO file you download in step 3
(You need to set the user information in first time start-up in step 9)



← Select start-up disk

Please select a virtual optical disk file or a physical optical drive containing a disk to start your new virtual machine from.

The disk should be suitable for starting a computer from and should contain the operating system you wish to install on the virtual machine if you want to do that now. The disk will be ejected from the virtual drive automatically next time you switch the virtual machine off, but you can also do this yourself if needed using the Devices menu.

ubuntu-20.04.3-desktop-amd64.iso (2.86 GB)

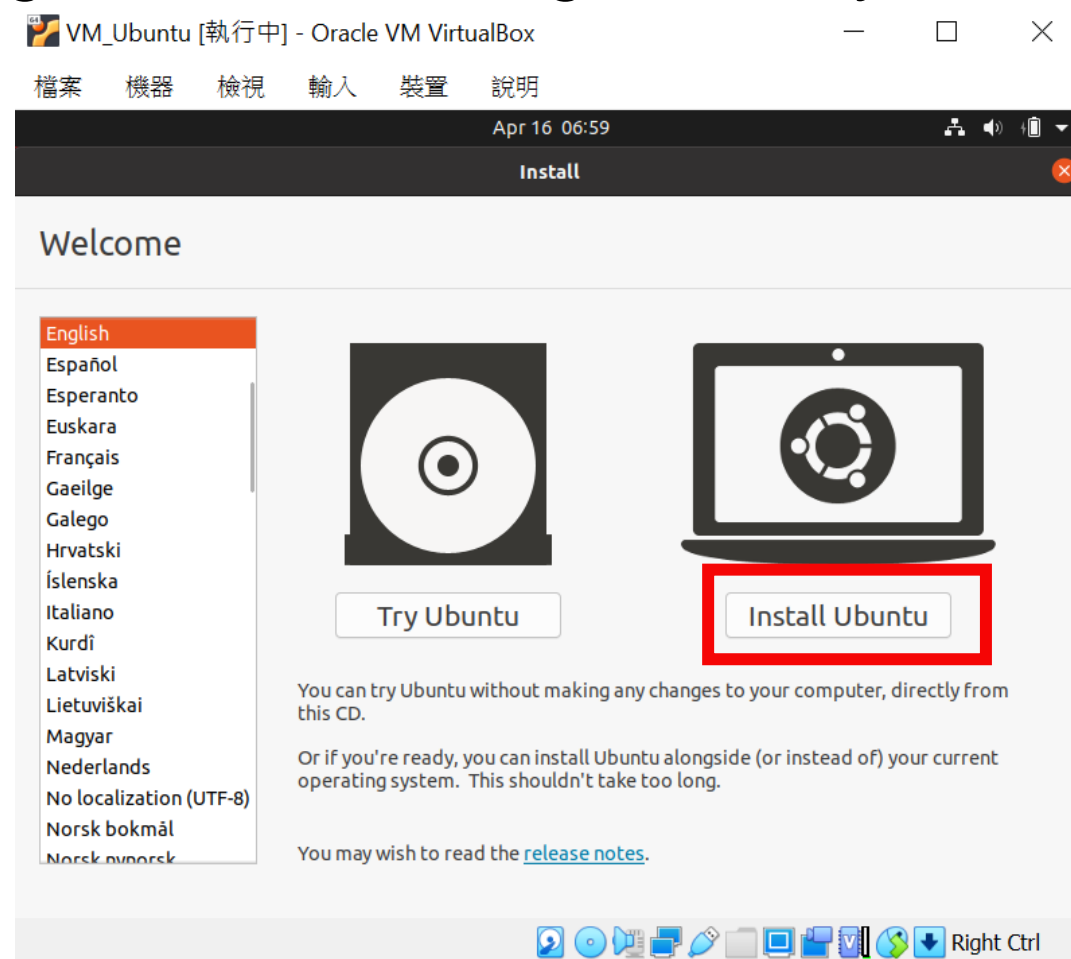
Start

Cancel

Download and Setup SDK

9. Select install Ubuntu

User can configure other setting accord your habit



Download and Setup SDK

10. After install Ubuntu, open Ubuntu and set the shared folder

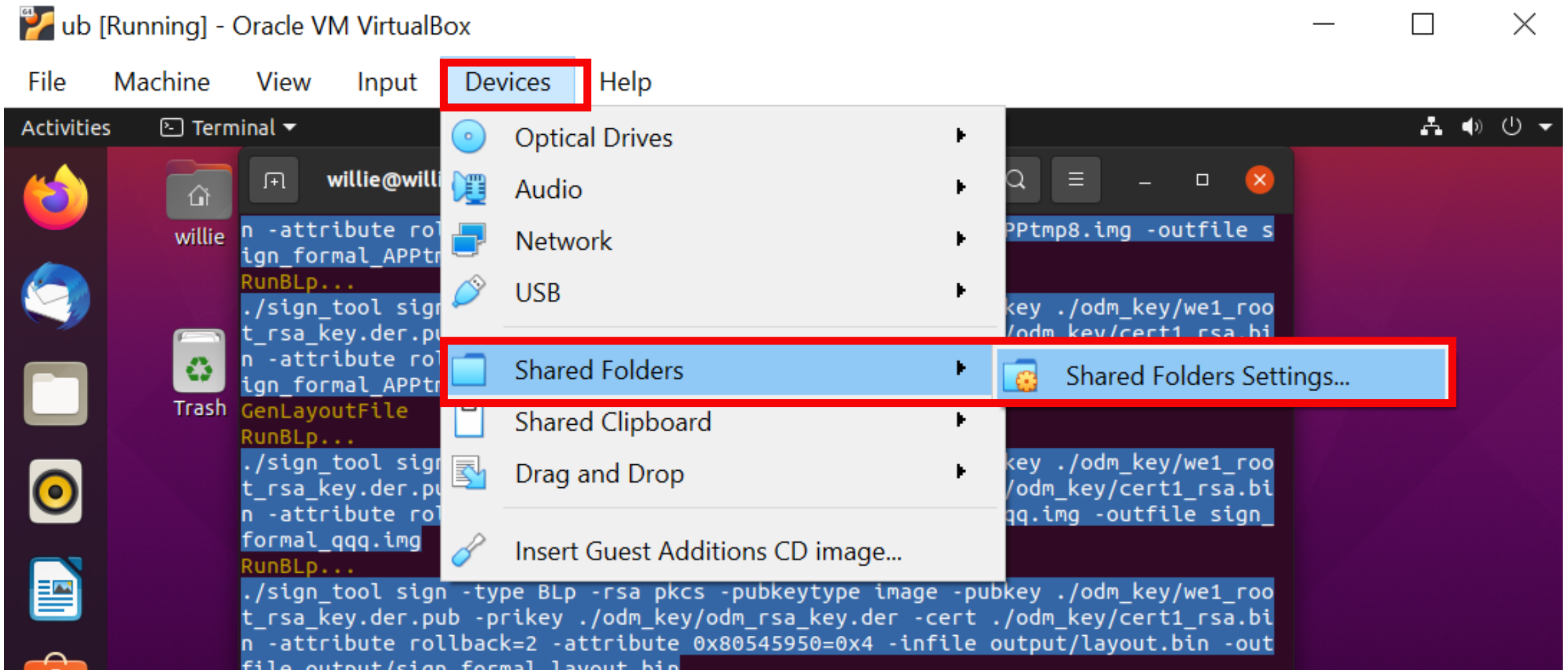
- User usually writes program and compiles in Windows but the image file for programming needs to convert in Ubuntu.

Therefore, you need to set the shared folder that two development environments can edit and use the same files.

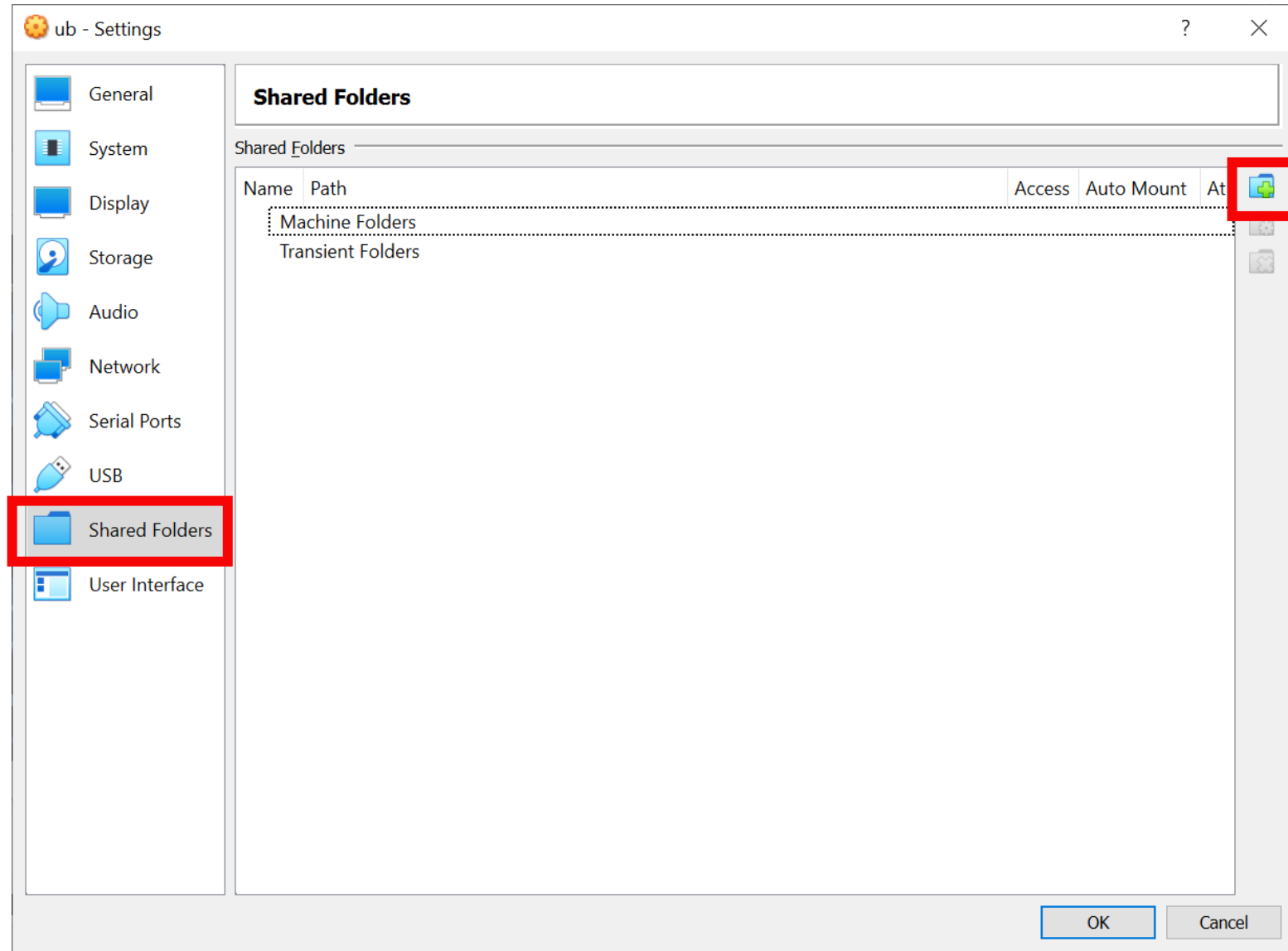
- User can also write program and compile in Ubuntu.

Download and Setup SDK

11. Set the folder to be shared




Download and Setup SDK




Download and Setup SDK

Add Share ? X

Folder Path: <not selected> 

Folder Name: <not selected>

 Other...

☐ Read-only

☐ Auto-mount

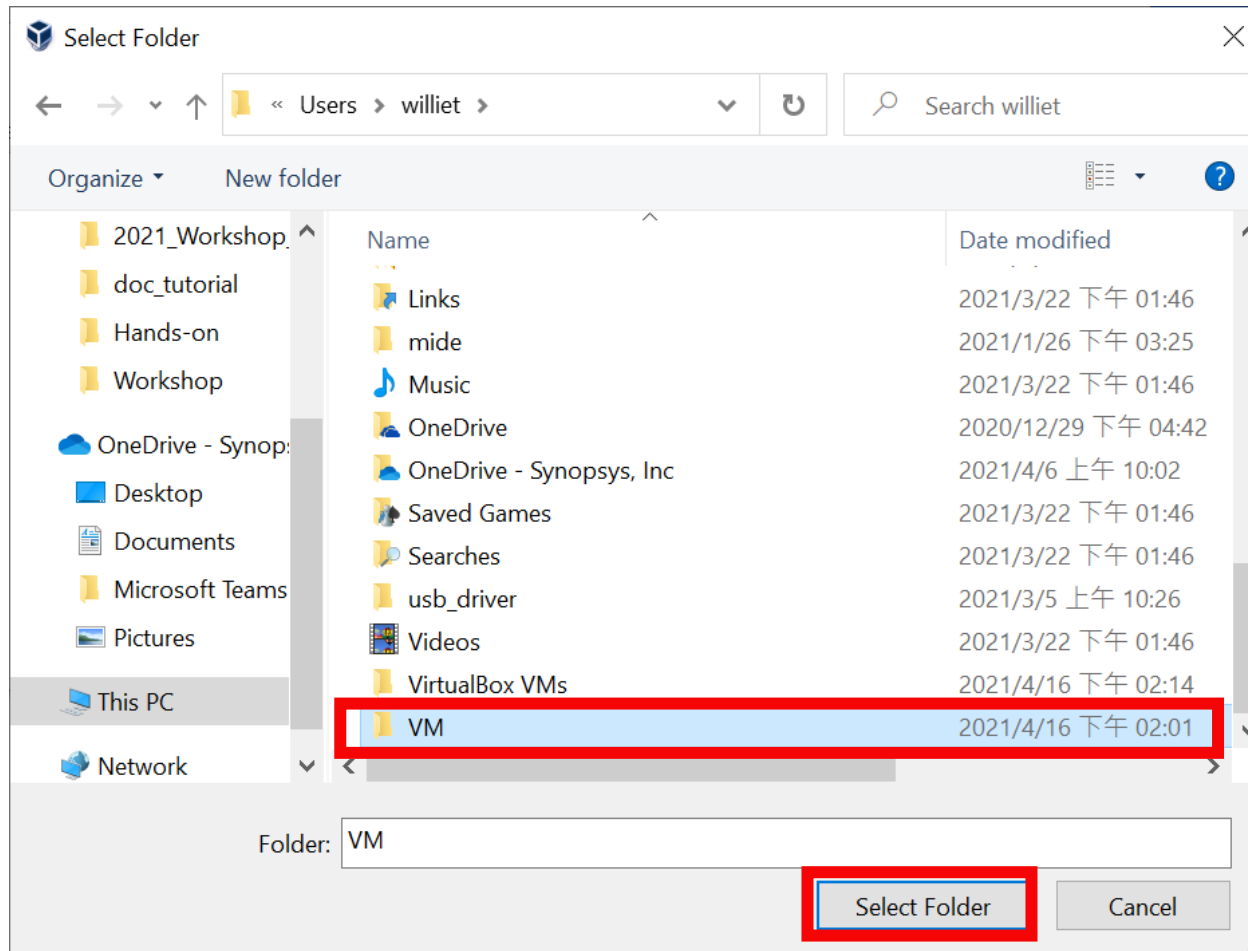
Mount point:

☐ Make Permanent

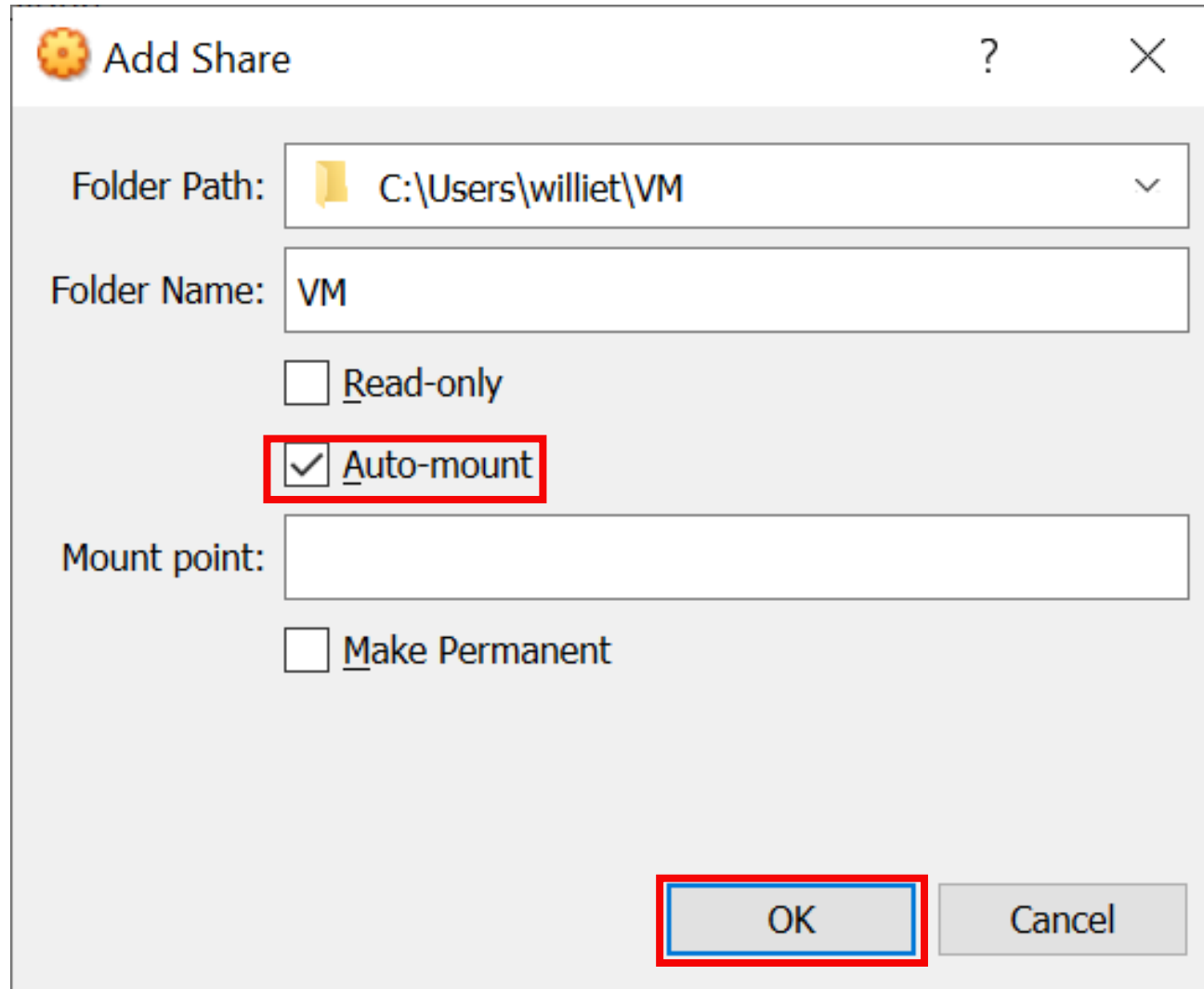
OK Cancel

Download and Setup SDK

Path “C:\Users\{username}\VM”



Download and Setup SDK

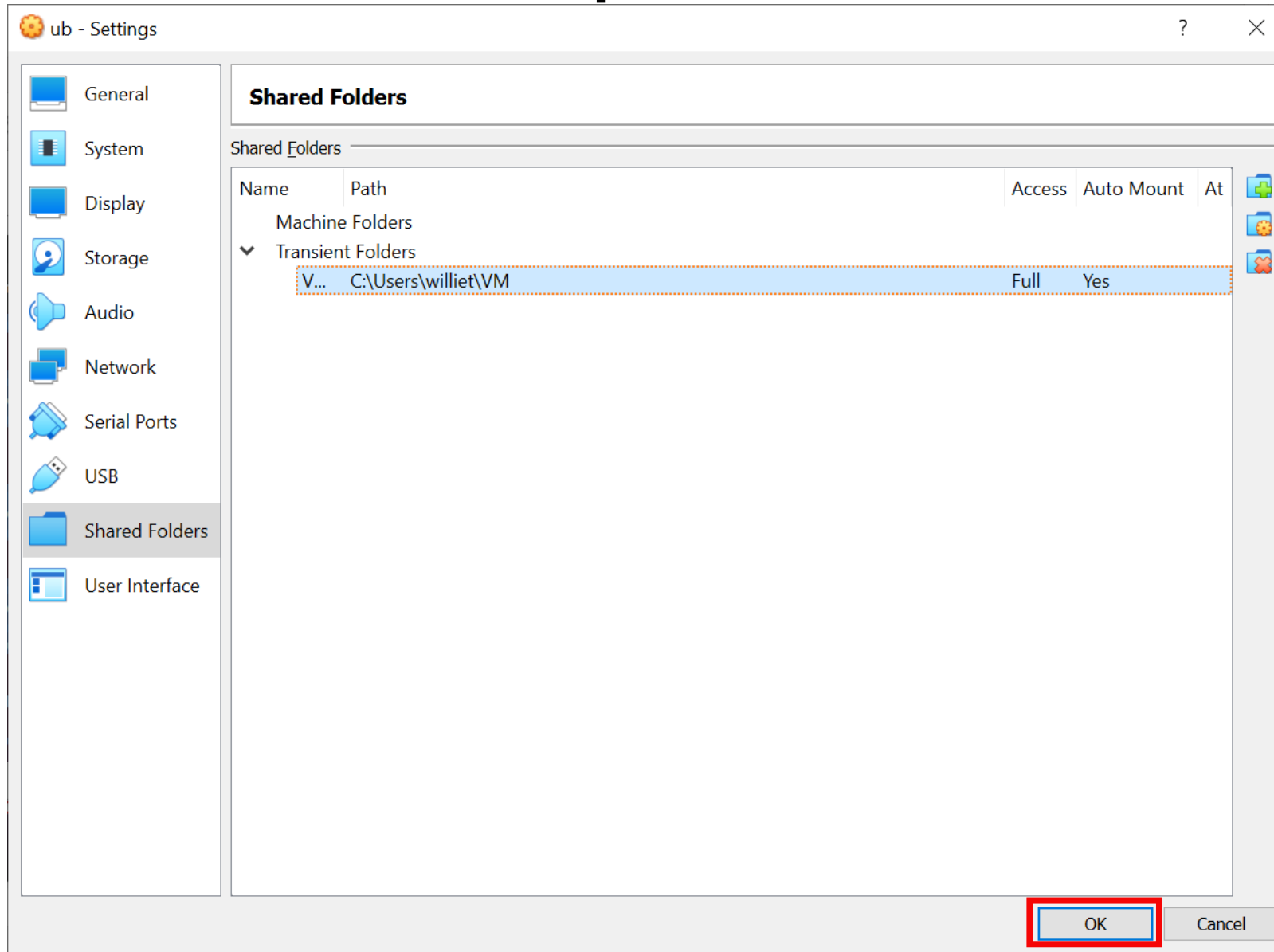


The screenshot shows a Windows 'Add Share' dialog box. The title bar includes a gear icon, the text 'Add Share', and standard window controls. The main area contains the following fields and options:

- Folder Path:** A dropdown menu showing 'C:\Users\williet\VM'.
- Folder Name:** A text box containing 'VM'.
- Read-only:** An unchecked checkbox.
- Auto-mount:** A checked checkbox, highlighted with a red rectangular box.
- Mount point:** An empty text box.
- Make Permanent:** An unchecked checkbox.

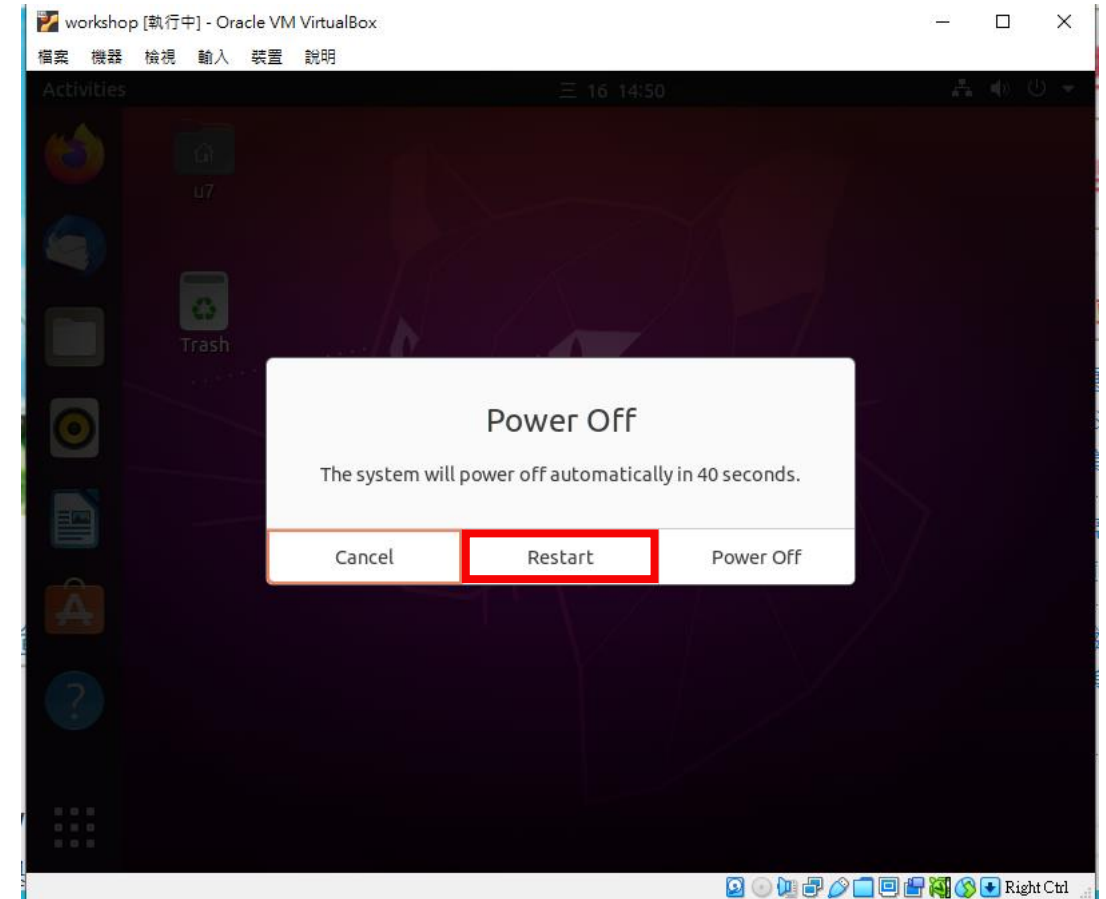
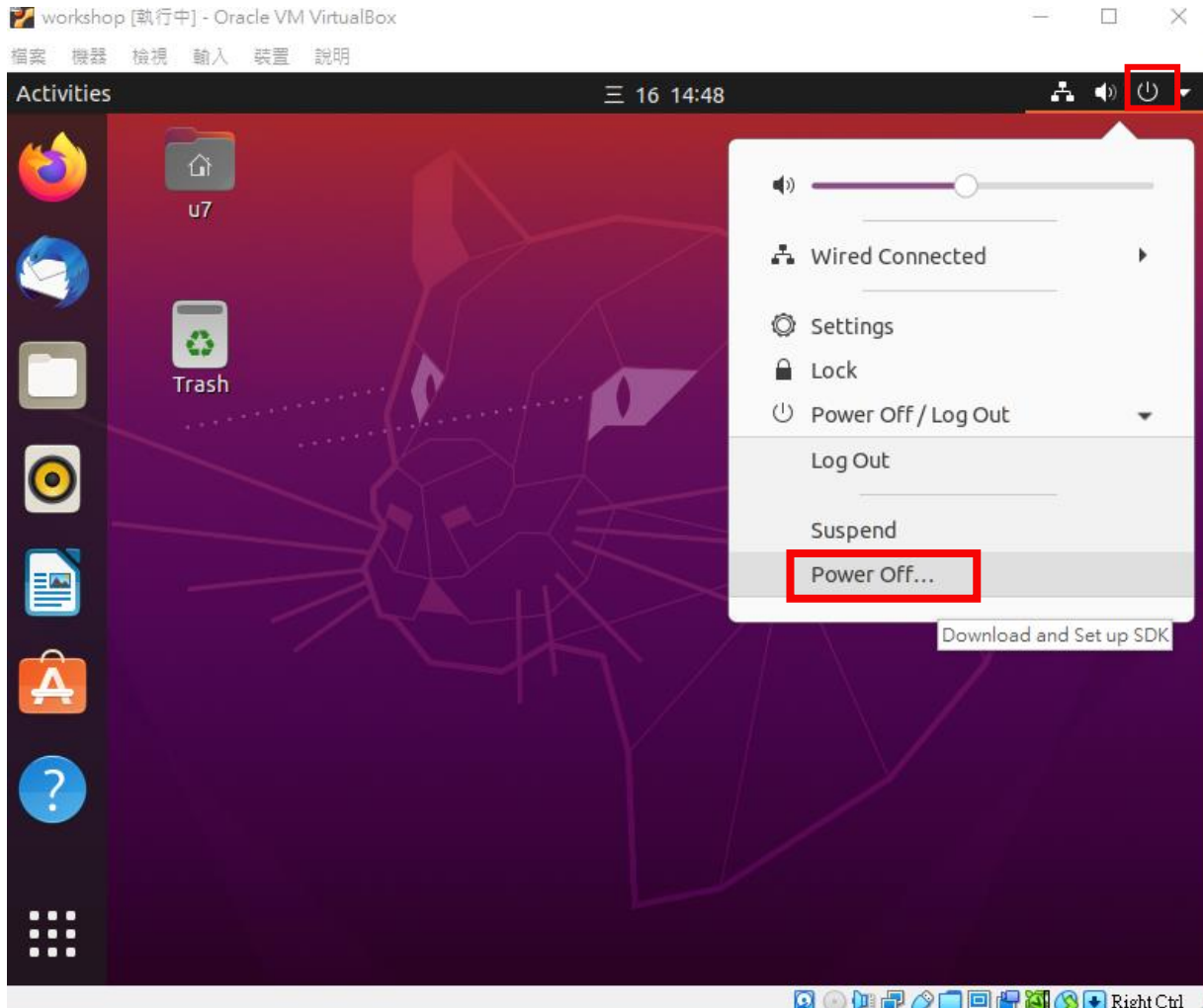
At the bottom right, there are two buttons: 'OK' and 'Cancel'. The 'OK' button is highlighted with a red rectangular box.

Download and Setup SDK



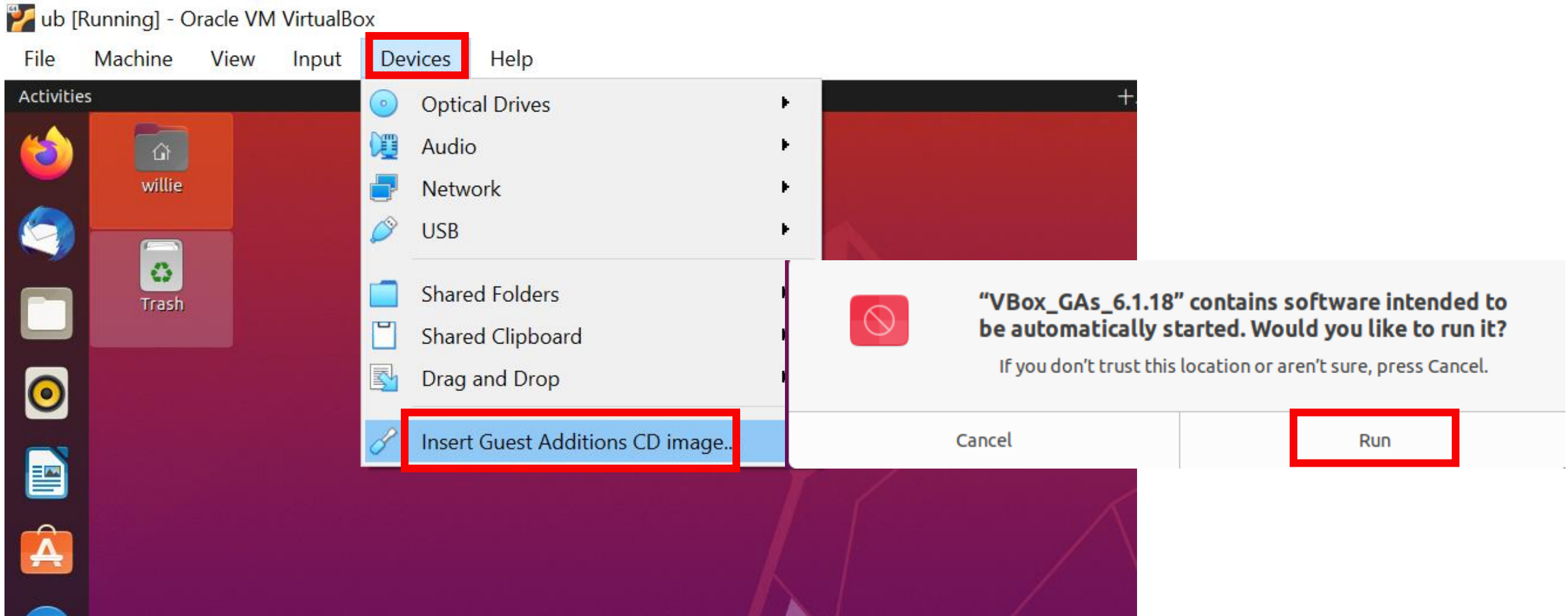
Download and Setup SDK

- Restart 



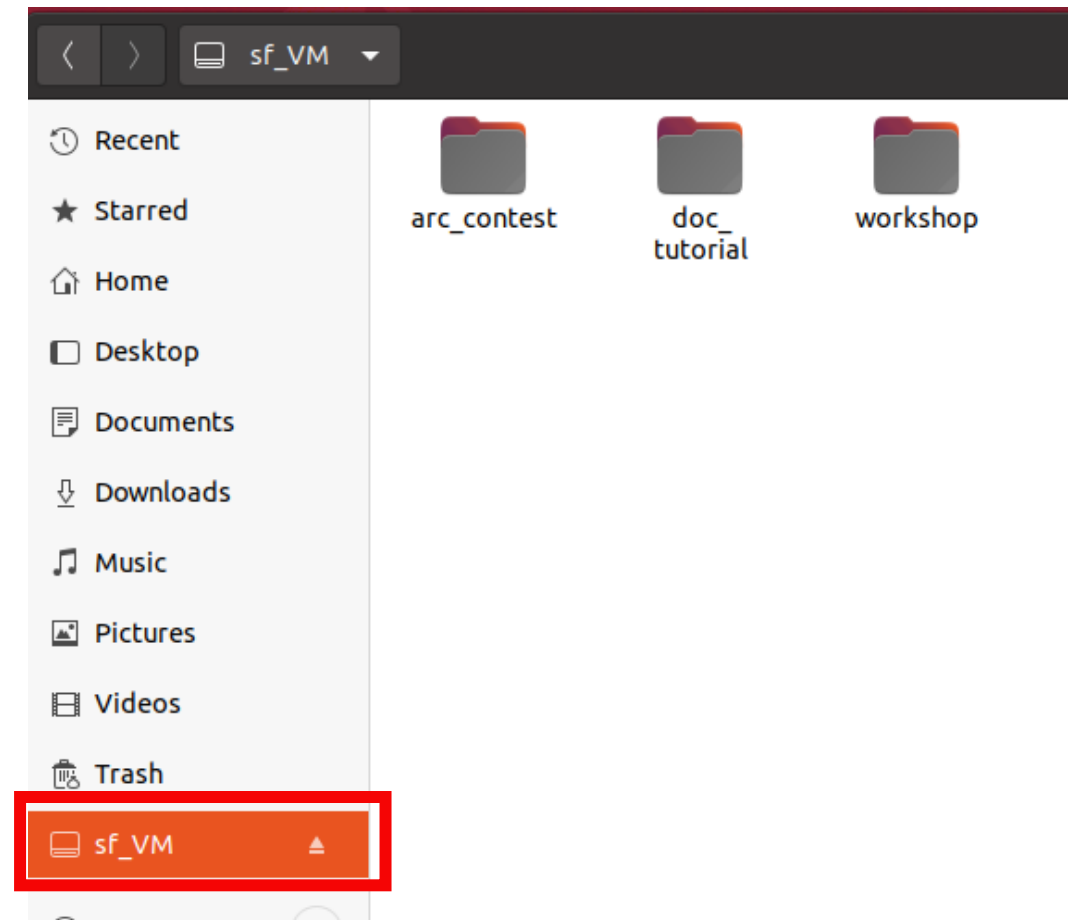
Download and Setup SDK

- Run Guest Addition CD



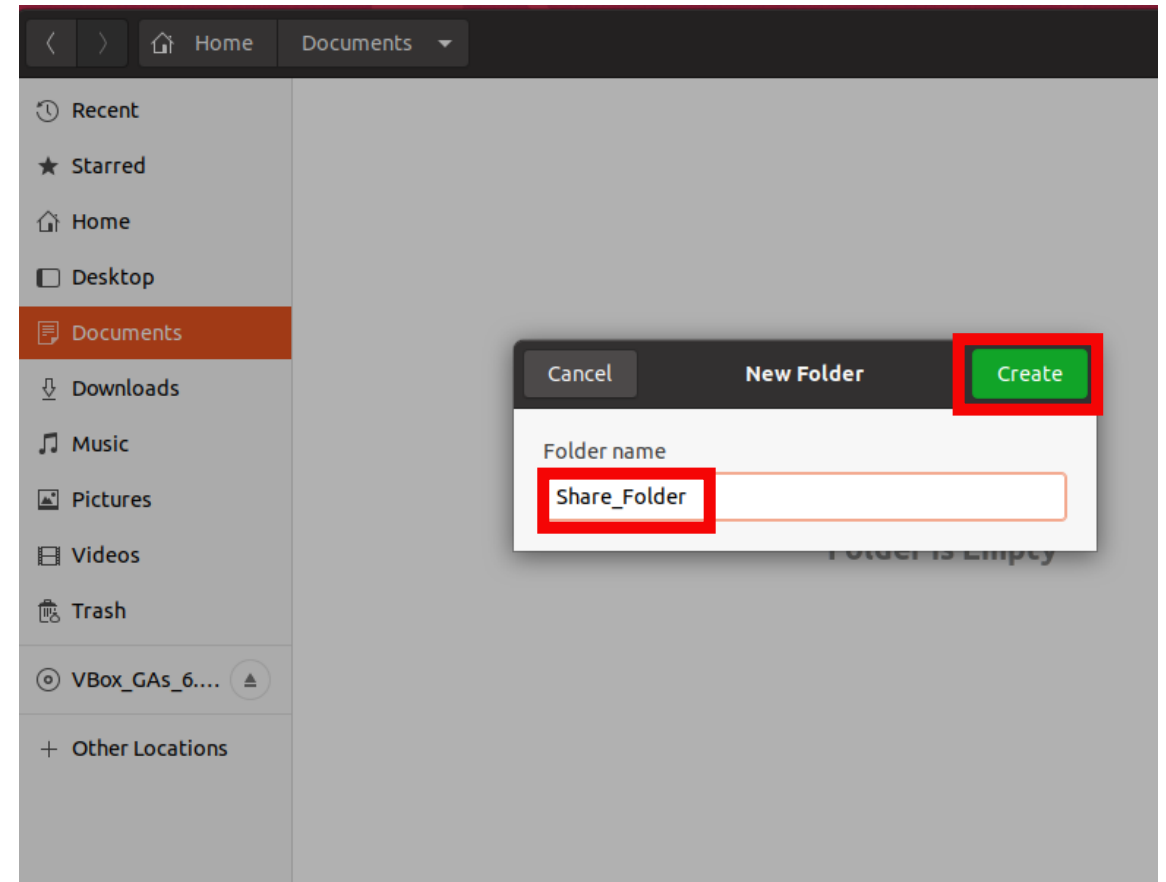
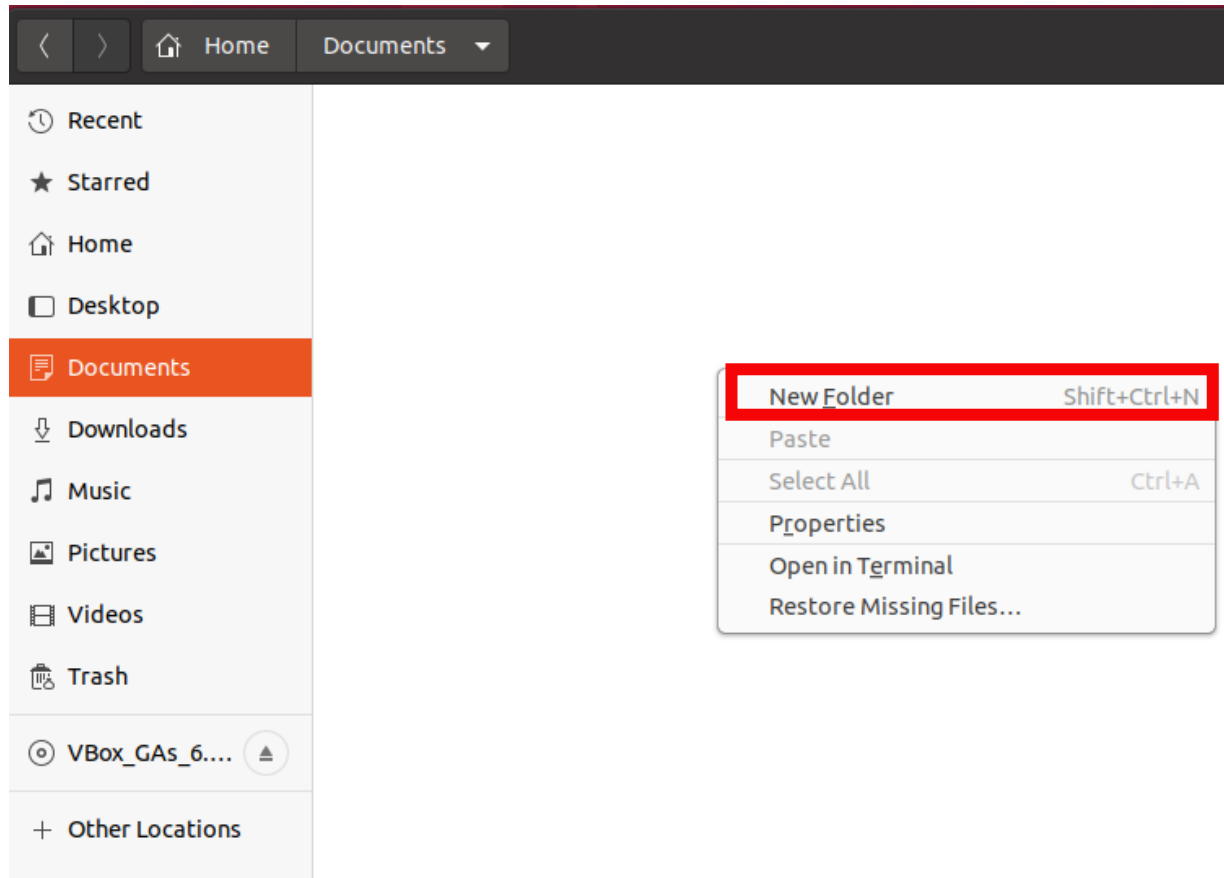
Download and Setup SDK

- After you mount, you can see files in Ubuntu “*sf_VM*” which are also in Windows “*C:\Users\{username}\VM*”
- If you mount well, go to step 13.
- If shared folder doesn't mount, please run step 12.



Download and Setup SDK

12. Create a new folder at “*Home/Documents/Share_Folder*”

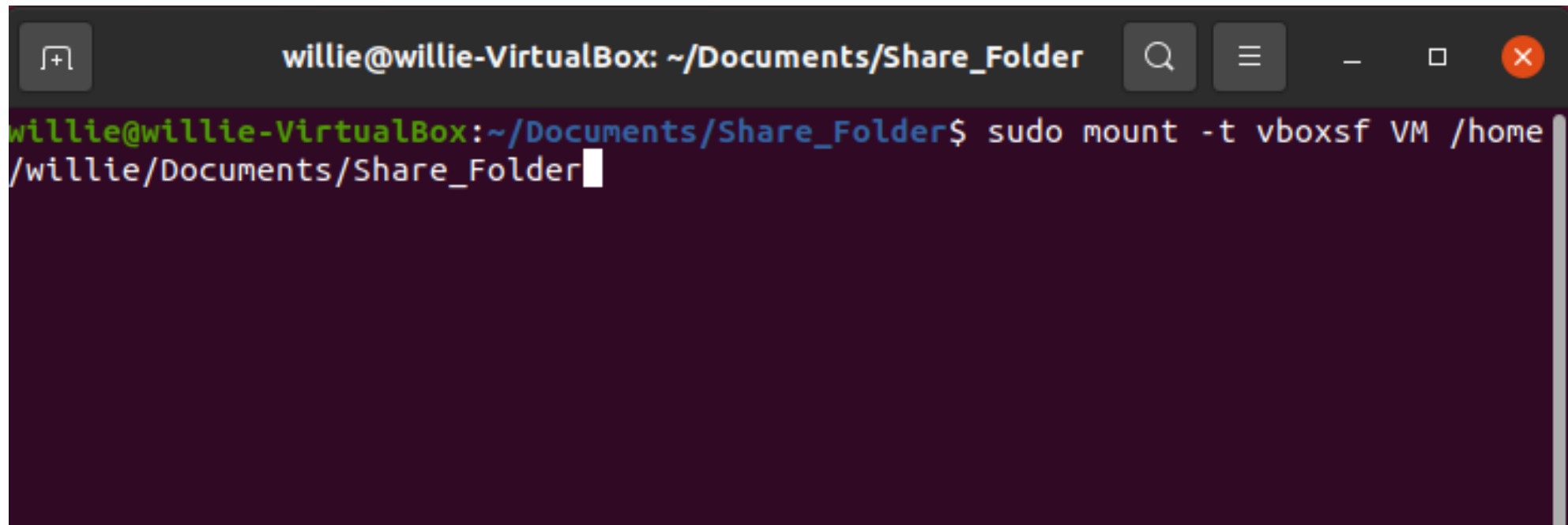


Download and Setup SDK

- Open Ubuntu terminal: Ctrl+Alt+t (or right click > Open in Terminal)

\$ `sudo mount -t vboxsf VM /home/{username}/Documents/Share_Folder`

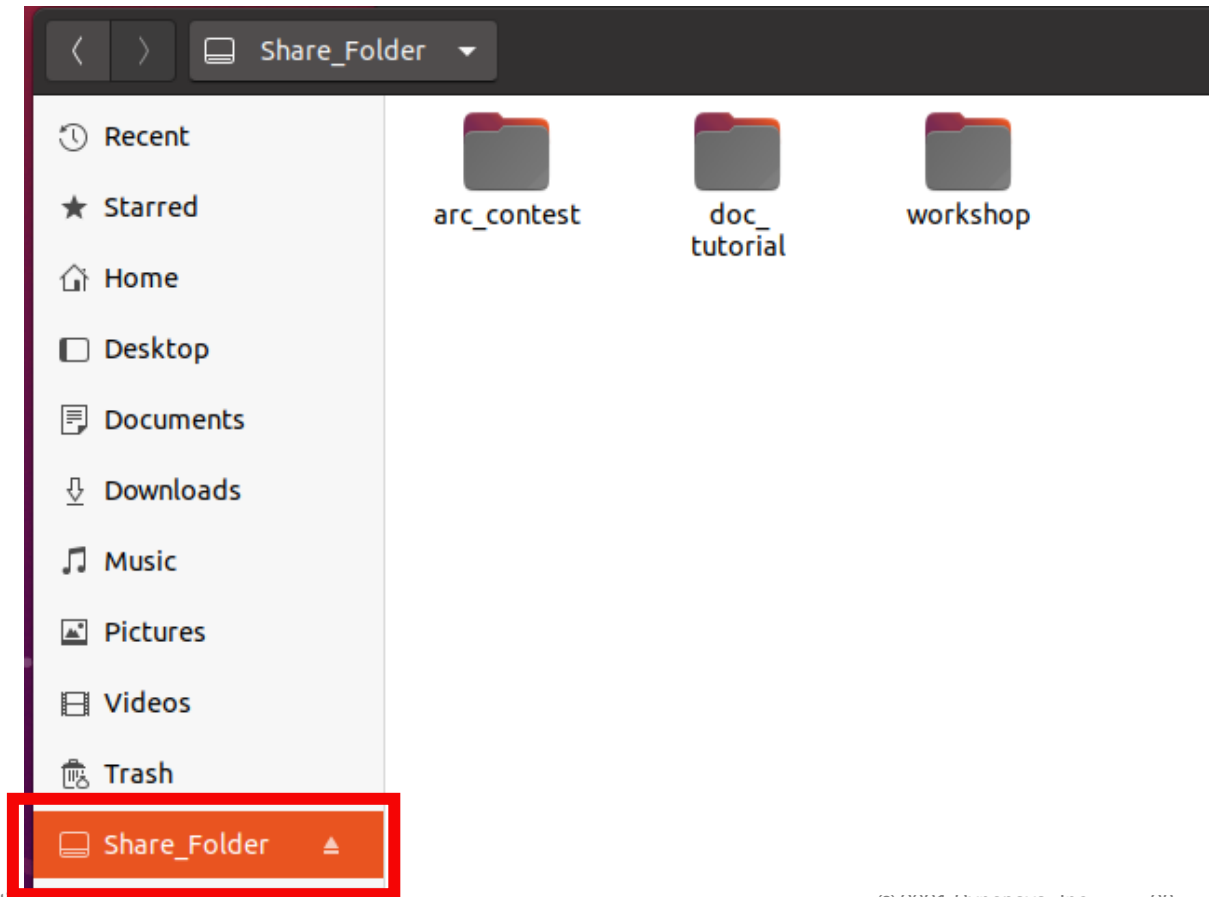
\$ Key-in your user password

A screenshot of a terminal window titled "willie@willie-VirtualBox: ~/Documents/Share_Folder". The terminal shows the command "sudo mount -t vboxsf VM /home/willie/Documents/Share_Folder" being entered. The prompt "willie@willie-VirtualBox:~/Documents/Share_Folder\$" is visible at the end of the command line.

```
willie@willie-VirtualBox: ~/Documents/Share_Folder
willie@willie-VirtualBox:~/Documents/Share_Folder$ sudo mount -t vboxsf VM /home
/willie/Documents/Share_Folder
```

Download and Setup SDK

- After you mount, you can see files in Ubuntu *“home/{username}/Documents/Share_Folder”* which are also in Windows *“C:\Users\{username}\VM”*

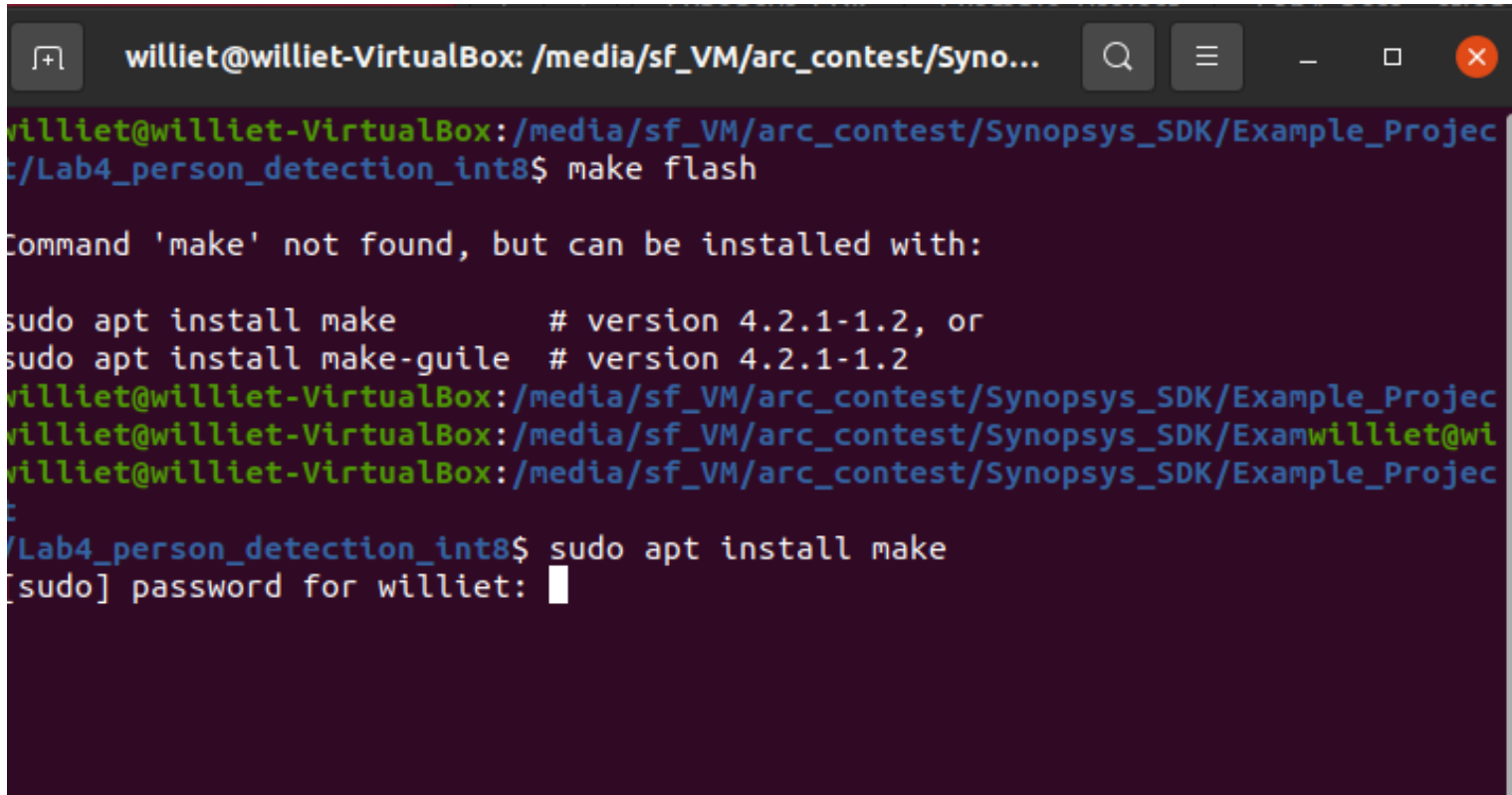


Download and Setup SDK

13. Install “make” command

\$ sudo apt install make

\$ Key-in your user password

A terminal window screenshot showing a user named 'williet' in a 'VirtualBox' environment. The user is in the directory '/media/sf_VM/arc_contest/Synopsys_SDK/Example_Project/Lab4_person_detection_int8'. They attempt to run 'make flash', but the command is not found. The terminal suggests installing 'make' with 'sudo apt install make'. The user then enters the command, and the terminal prompts for their password, which is masked with a black box.

```
williet@williet-VirtualBox: /media/sf_VM/arc_contest/Synopsys_SDK/Example_Project/Lab4_person_detection_int8$ make flash

Command 'make' not found, but can be installed with:

sudo apt install make          # version 4.2.1-1.2, or
sudo apt install make-guile    # version 4.2.1-1.2

williet@williet-VirtualBox: /media/sf_VM/arc_contest/Synopsys_SDK/Example_Project/Lab4_person_detection_int8$ sudo apt install make
[sudo] password for williet: 
```

Download and Setup SDK

14.Download and setup Linux ARC GNU file (Version: 2019.09)

<https://github.com/foss-for-synopsys-dwc-arc-processors/toolchain/releases/tag/arc-2019.09-release>

Other supported devices are `arc-hs` , `arc-600` and `arc-700` .

See section "3.2 Configuring the Ashling GDB Server" in the Ashling Opella-XD ARC User manual for more details.

4. Newlib's libgloss doesn't support RF16 configuration of ARC cores when building for nSIM, see [#231](#).

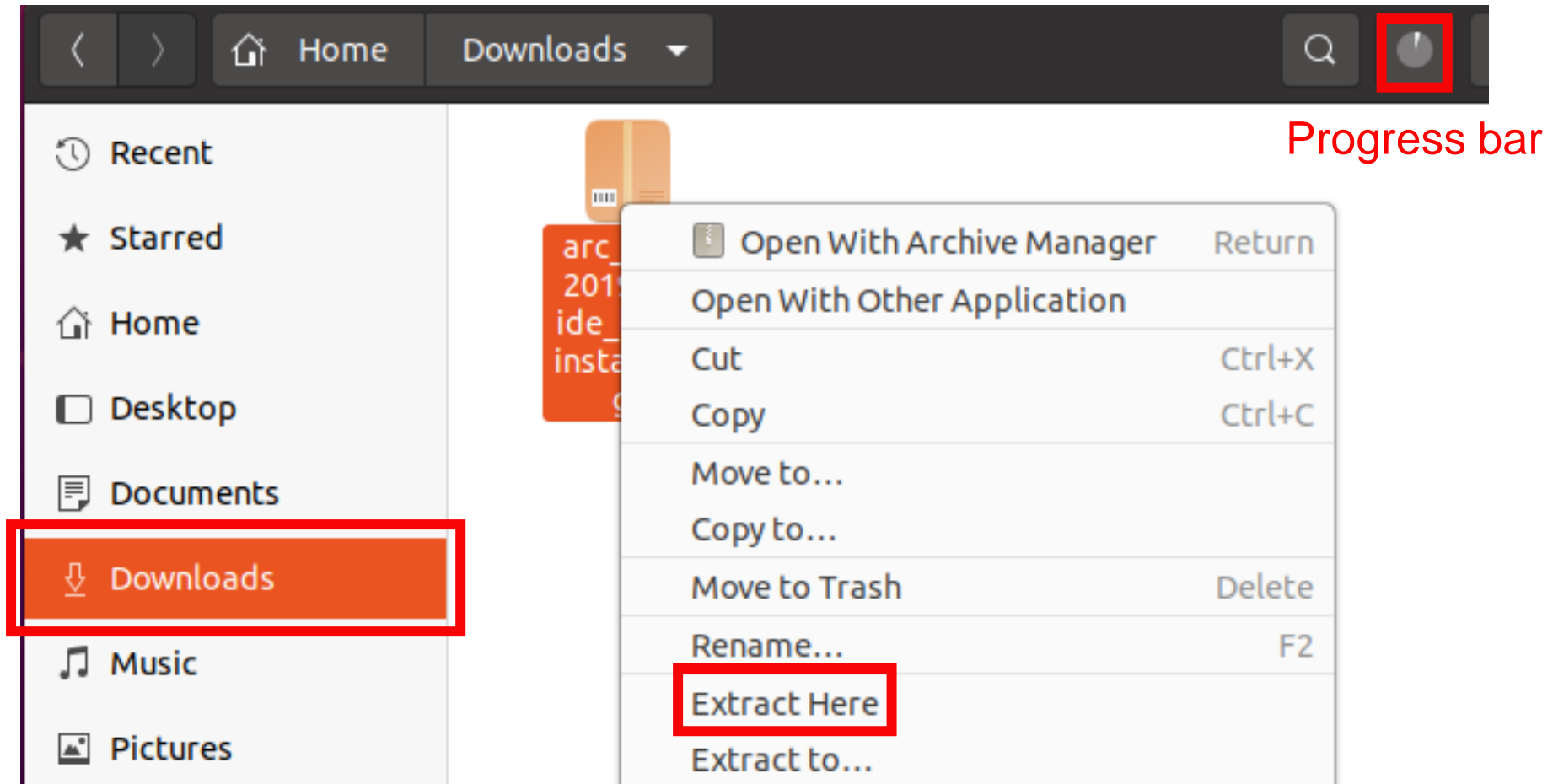
Please report any problems by filing an Issue in GitHub [here](#).

Note: The toolchain is only supported for 64-bit versions of Linux hosts. 32-bit Linux hosts are not supported.

	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

Download and Setup SDK

15. Go to “*Downloads*” and unzip install package
(Right Click > Extract Here)



Download and Setup SDK

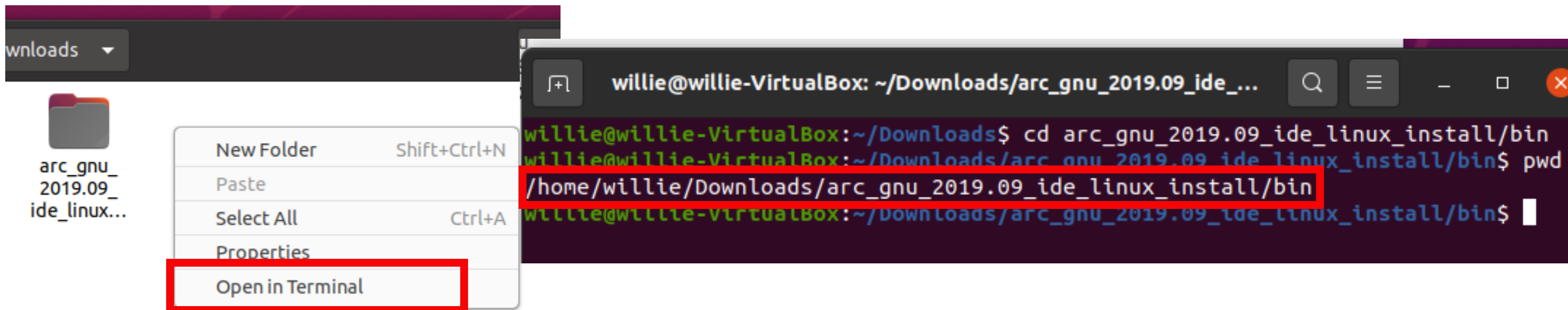
16. After unzip it, open the terminal.

(Right Click > Open in Terminal)

```
$ cd arc_gnu_2019.09_ide_linux_install/bin
```

```
$ pwd
```

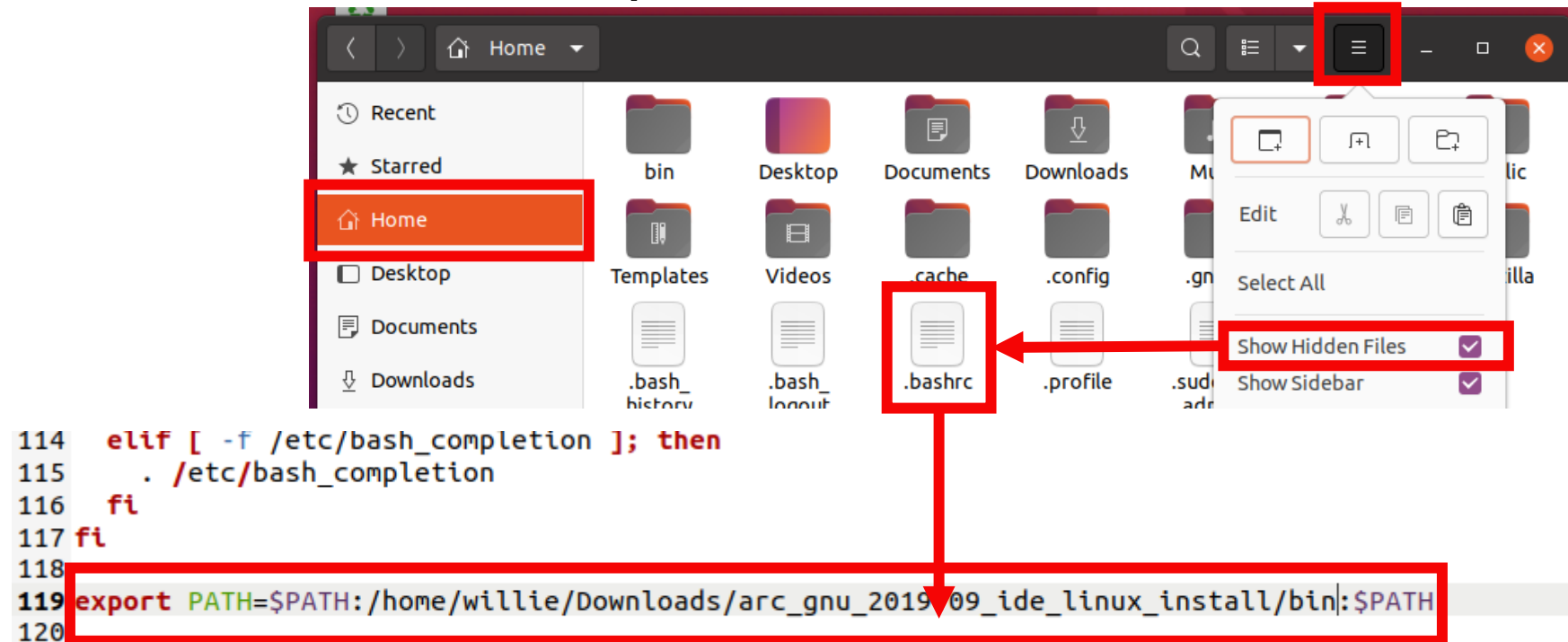
You will get the path and be used in next step.



Download and Setup SDK

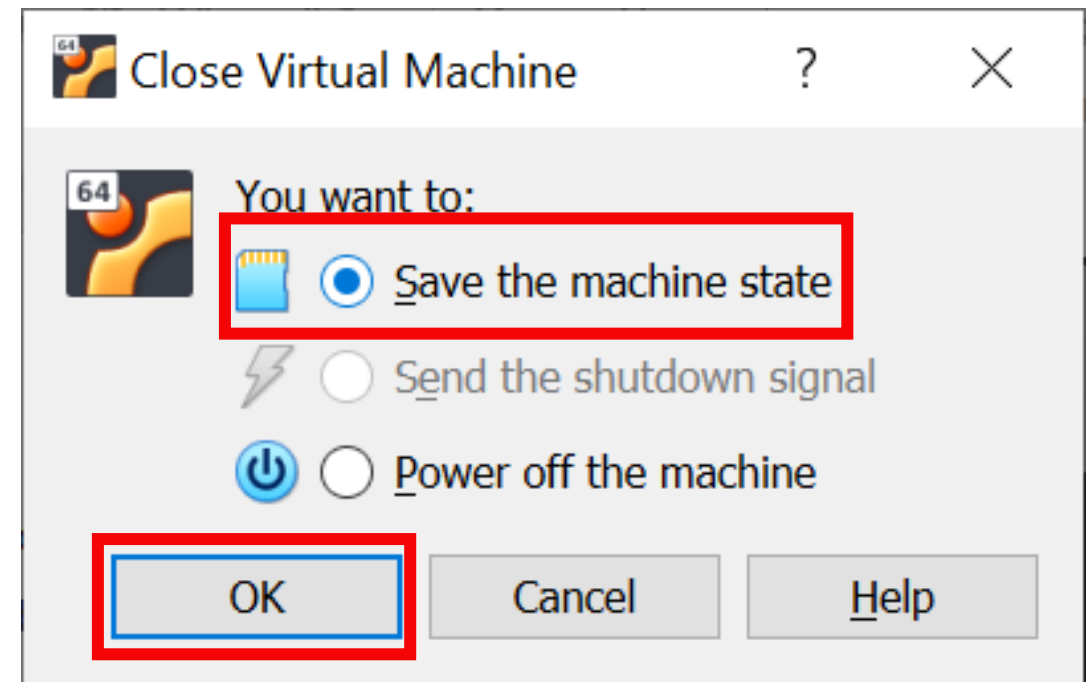
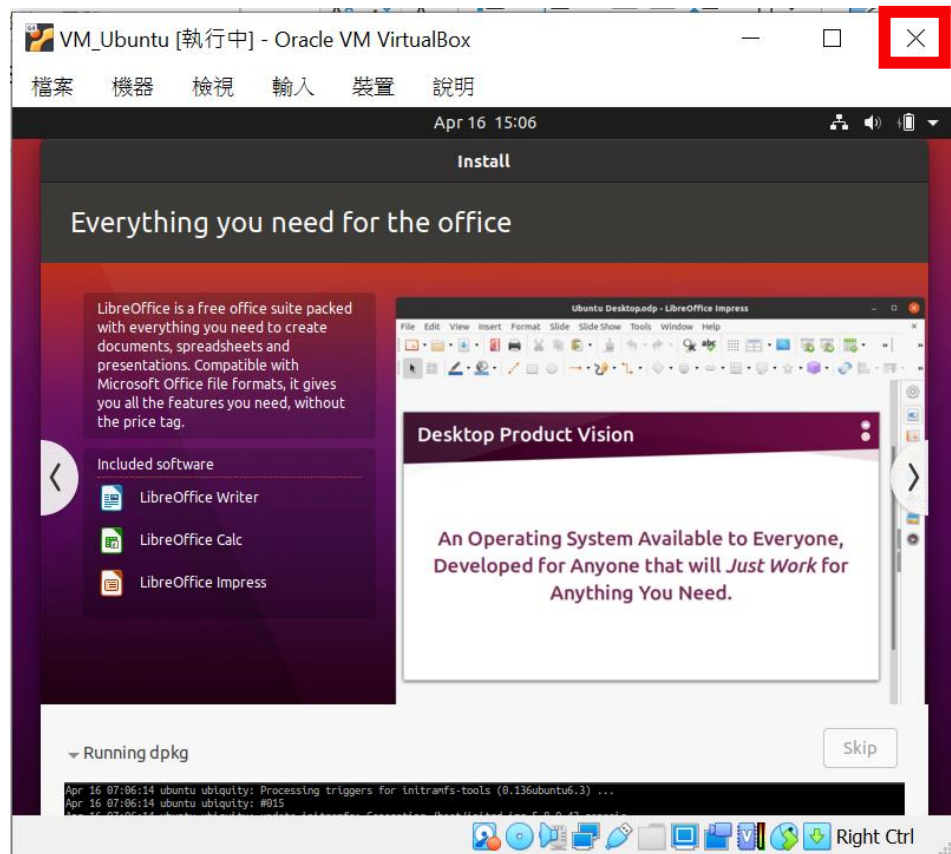
17. Edit “*/home/.bashrc*” for setting environment variable, add
“*export PATH=\$PATH:/home/{username}/Downloads/{ARC_GNU_ROOT}/bin:\$PATH*”
in the last line.

“*/home/{username}/Downloads/{ARC_GNU_ROOT}/bin*”
this path must be the same as last step.



Download and Setup SDK

18. Recommend to close your VirtualBox with “Store Computer Stage”
If not, you may need to remount shard folder every time.



Program Code & Make Project



Program Code

1. Go into folder

“Synopsys_SDK_Vxx/Example Project/Lab4_tflm_person_detect”

you will see folder “src” and “inc”

“src” folder: always keep your .c and .cpp file in here.

“inc” folder: always keep your .h file in here.

(c file: c language)

(cpp file: c++ language)

Make Project (By cygwin64)

2. Go to your project path in cygwin64 terminal

```
$ cd c:
```

```
$ cd Users/{username}/VM
```

```
$ cd Synopsys_SDK_Vxx/Example Project/Lab4_tflm_person_detect
```

```
$ make
```

```
williet@WILLIET-7490 /cygdrive/c/Users/williet/VM/Synopsys_SDK_Vxx/Example_Proje
ct/Lab4_tflm_person_detect_v2
$ cd c:

williet@WILLIET-7490 /cygdrive/c
$ cd Users/williet/VM/

williet@WILLIET-7490 /cygdrive/c/Users/williet/VM
$ cd Synopsys_SDK_Vxx/Example_Project/Lab4_tflm_person_detect_v2/

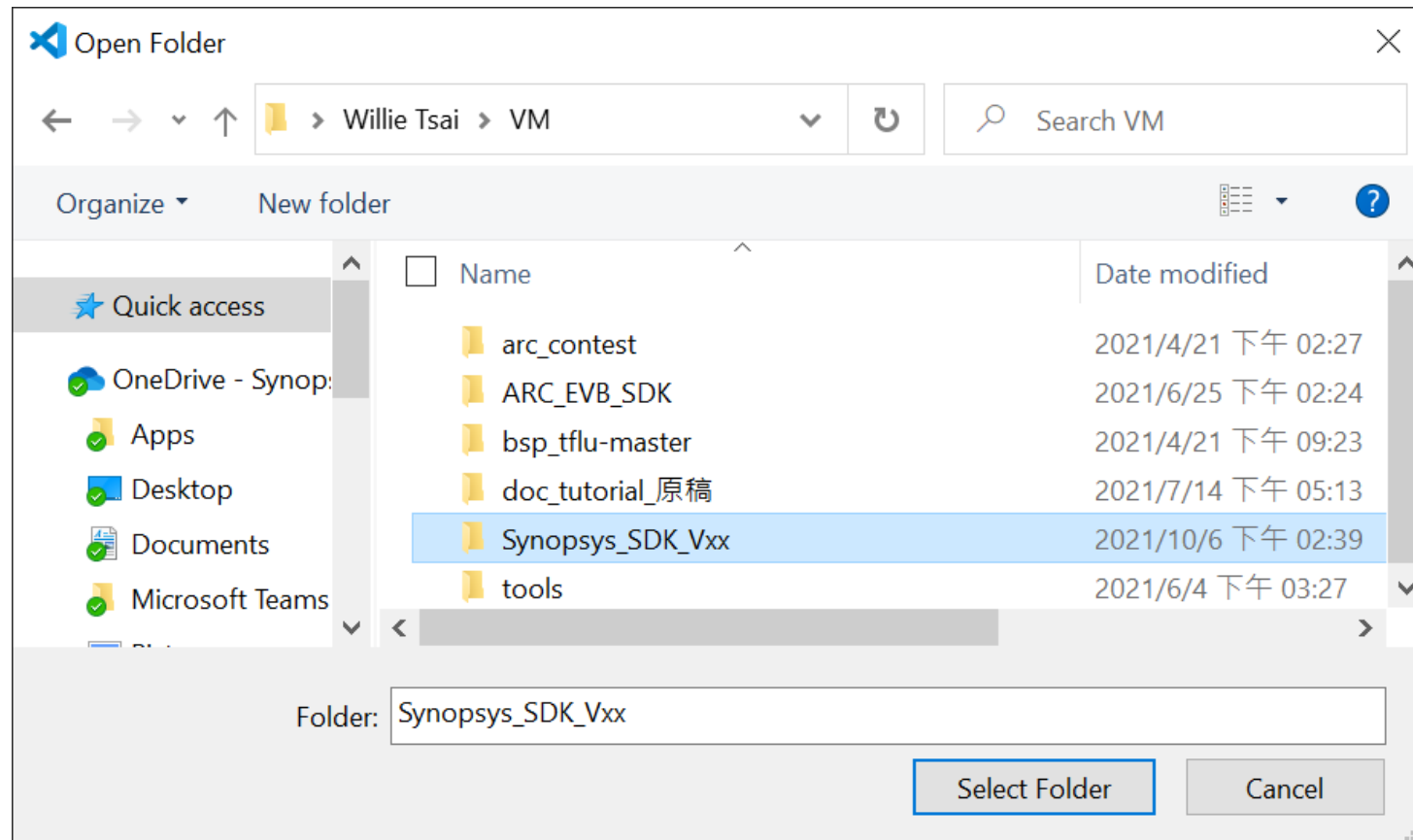
williet@WILLIET-7490 /cygdrive/c/Users/williet/VM/Synopsys_SDK_Vxx/Example_Proje
ct/Lab4_tflm_person_detect_v2
$ make
../../def_linker_script/def_linker_script.mk:27: ../../def_linker_script/linke
r_template_gnu_M.ld
Compiling          : src/app_cis_sensor_defcfg.c
arc-elf32-gcc -c -include ../../board/socket/configs/24/core_config.h -DAPDS_9
```

Make Project (By Visual Studio Code)

3. You can also use Visual Studio Code to make project.

Open Folder and Select Folder:

“...../Synopsys_SDK_Vxx”

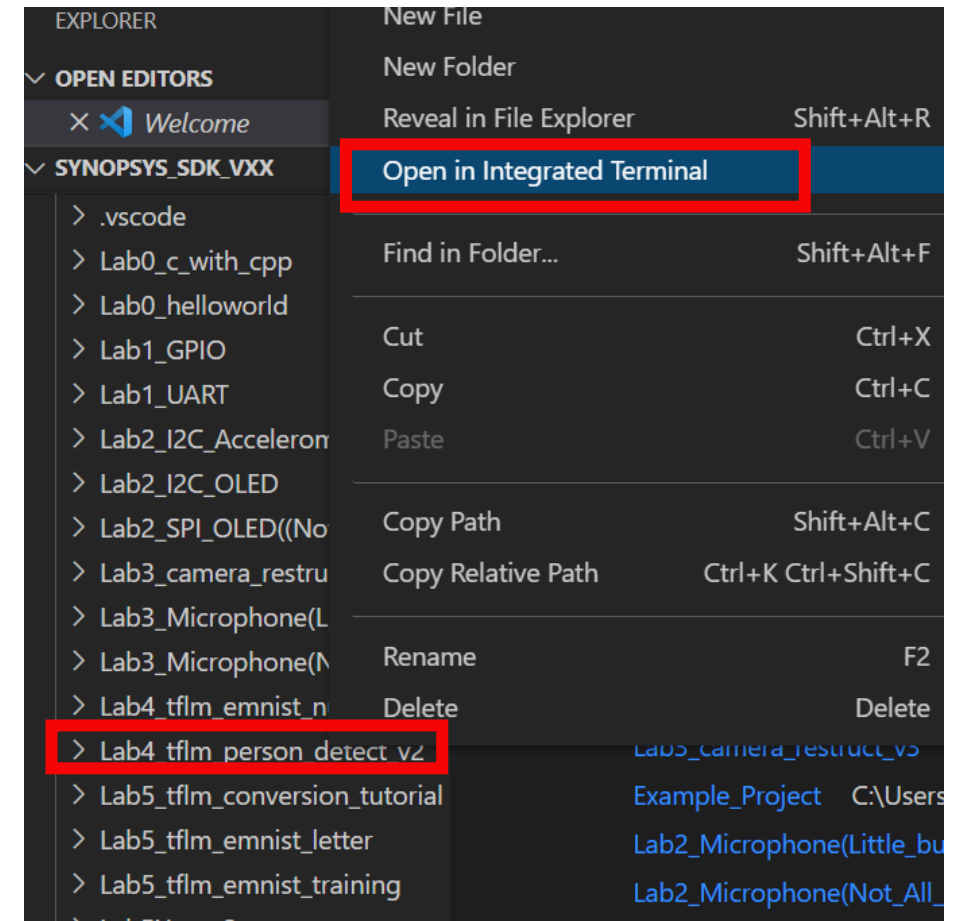


Make Project (By Visual Studio Code)

4. Navigate to

“Synopsys_SDK_Vxx/Example_Project/Lab4_tflm_person_detect”
in VS Code

Right Click “Lab4_tflm_person_detect”
> “Open in Integrated Terminal”

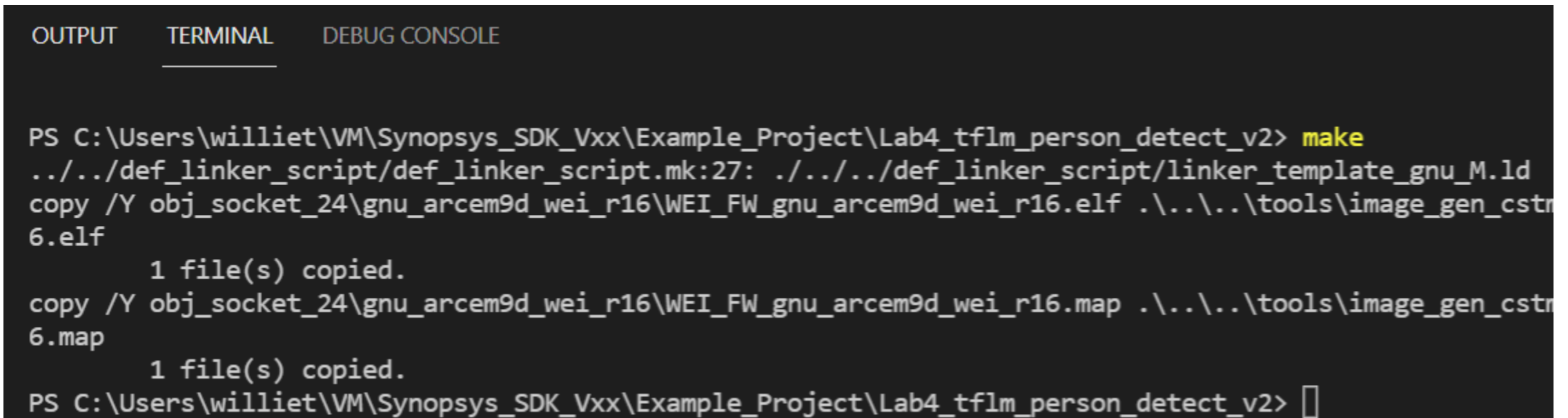


Make Project (By Visual Studio Code)

5. You will see terminal block

Type command:

\$ make



```
OUTPUT  TERMINAL  DEBUG CONSOLE

PS C:\Users\williet\VM\Synopsys_SDK_Vxx\Example_Project\Lab4_tflm_person_detect_v2> make
../../def_linker_script/def_linker_script.mk:27: ../../def_linker_script/linker_template_gnu_M.ld
copy /Y obj_socket_24\gnu_arcem9d_wei_r16\WEI_FW_gnu_arcem9d_wei_r16.elf ../../tools/image_gen_cstrn
6.elf
        1 file(s) copied.
copy /Y obj_socket_24\gnu_arcem9d_wei_r16\WEI_FW_gnu_arcem9d_wei_r16.map ../../tools/image_gen_cstrn
6.map
        1 file(s) copied.
PS C:\Users\williet\VM\Synopsys_SDK_Vxx\Example_Project\Lab4_tflm_person_detect_v2> 
```

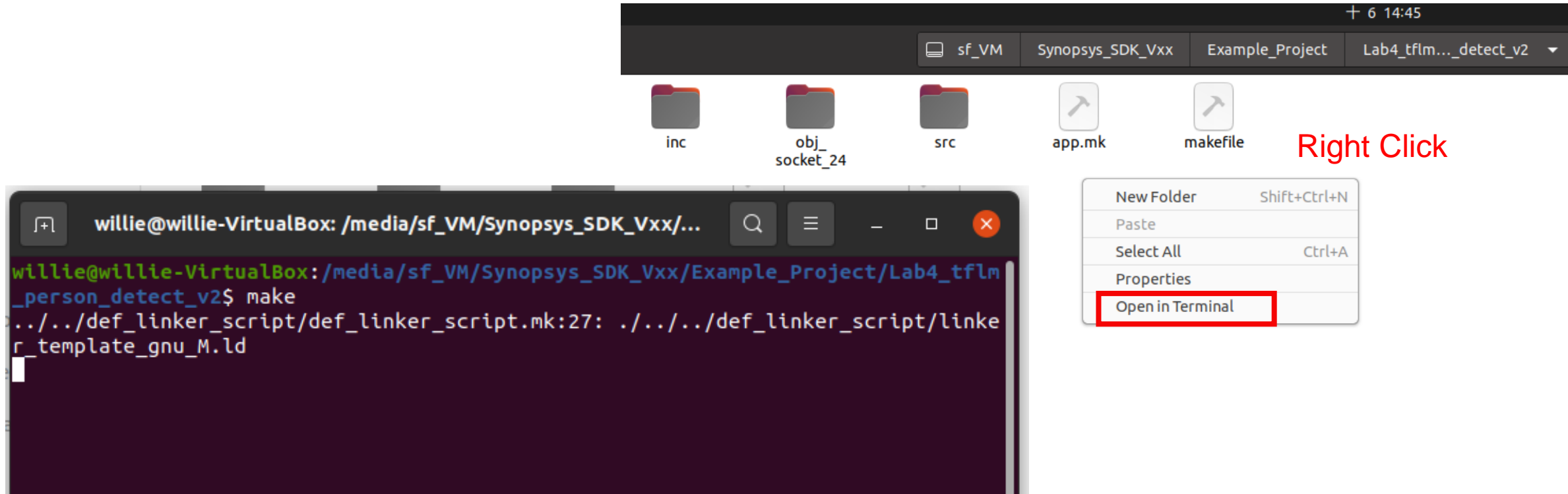

Make Project (By Ubuntu)

6. You can also use Ubuntu to make project

Open terminal in VirtualBox Ubuntu, and go to the same path:

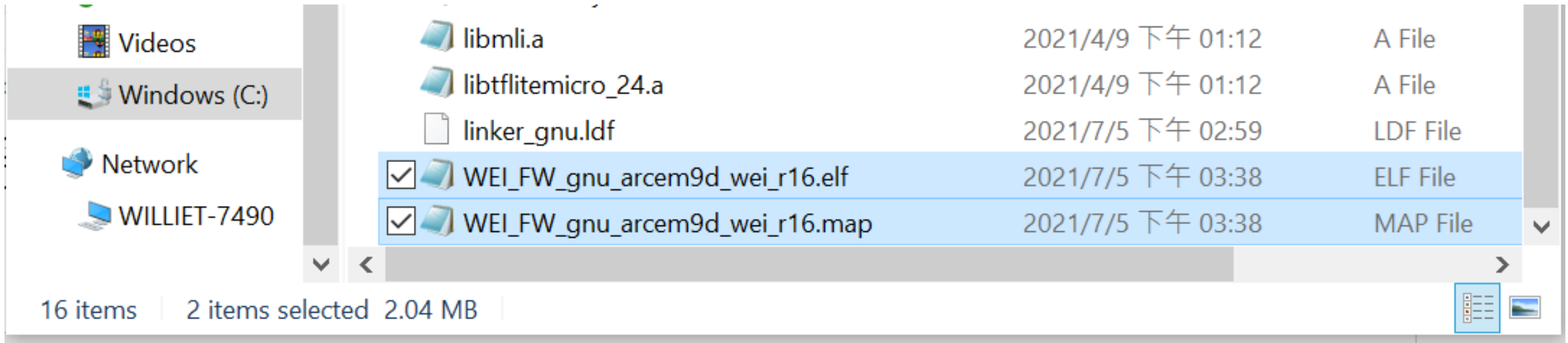
“/Synopsys_SDK_Vxx/Example_Project/Lab4_tflm_person_detect”

\$ make



Make Project

7. After you make project successfully, you will get “.elf” and “.map” file in “[Synopsys_SDK_Vxx/Example_Project/Lab4_tflm_person_detect/obj_socket_24/gnu_arcem9d_wei_r16/](#)”



WEI_FW_gnu_arcem9d_wei_r16: elf and map file are made by GNU toolchain
WEI_FW_mw_arcem9d_wei_r16: elf and map file are made by Metaware toolchain

Make Project

There are some commands can be used,

\$ **make** : compile and link your project with default toolchain.

\$ **make TOOLCHAIN=gnu**: compile and link your project with GNU.

\$ **make TOOLCHAIN=mw**: compile and link your project with Metaware.

\$ **make clean** : remove all .o file of default toolchain of this project

\$ **make boardclean** : remove all .o file of all toolchain of this project

You can add a command for changing toolchain

(default toolchain is gnu, define in makefile)

“TOOLCHAIN=mw”: compile with MetaWare toolchain

“TOOLCHAIN=gnu”: compile with ARC GNU toolchain

Please use \$ **make boardclean** after you change toolchain.

Make Project

\$ make TOOLCHAIN=gnu:

After compile successfully, create elf and map which are named bellow.

`"/obj_socket_24/gnu_arcem9d_wei_r16/WEI_FW_gnu_arcem9d_wei_r16.elf"`

`"/obj_socket_24/gnu_arcem9d_wei_r16/WEI_FW_gnu_arcem9d_wei_r16.map"`

\$ make TOOLCHAIN=mw:

After compile successfully, create elf and map which are named bellow.

`"/obj_socket_24/mw_arcem9d_wei_r16/WEI_FW_mw_arcem9d_wei_r16.elf"`

`"/obj_socket_24/mw_arcem9d_wei_r16/WEI_FW_mw_arcem9d_wei_r16.map"`

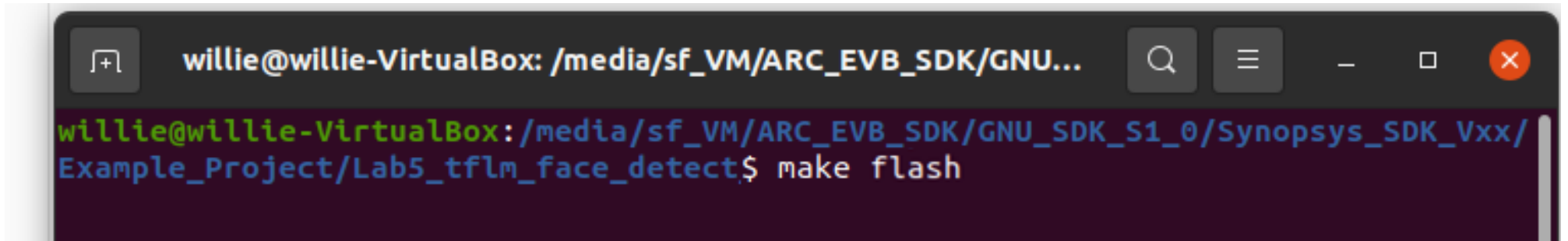
After compile, elf and map files will be copied automatically to
`"tools/image_gen_cstm/input"` and replace previous files automatically.

Make Flash File



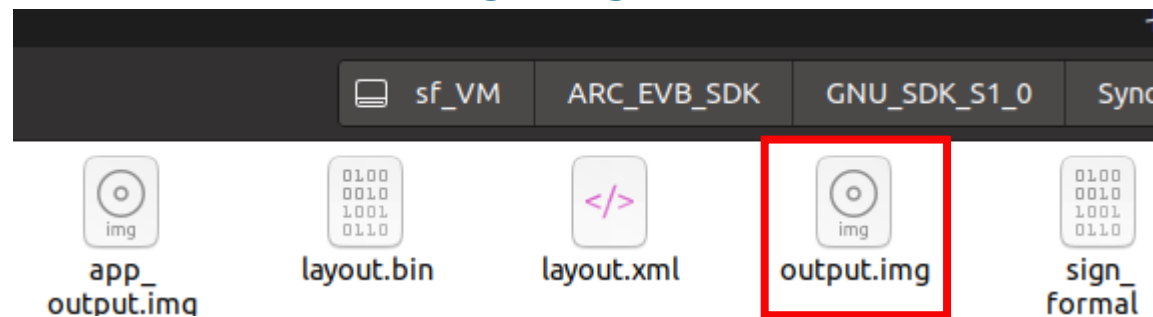
Make GNU Flash File (By image_gen_cstm_gnu)

1. Open terminal in VirtualBox Ubuntu, and go to the same path:
“/Synopsys_SDK_Vxx/Example_Project/Lab4_tflm_person_detect”
\$ make flash



```
willie@willie-VirtualBox: /media/sf_VM/ARC_EVB_SDK/GNU...  
willie@willie-VirtualBox:/media/sf_VM/ARC_EVB_SDK/GNU_SDK_S1_0/Synopsys_SDK_Vxx/  
Example_Project/Lab5_tflm_face_detect$ make flash
```

2. Check your “.img” file
“...../Synopsys_SDK_Vxx/tools/image_gen_cstm/output/output.img”



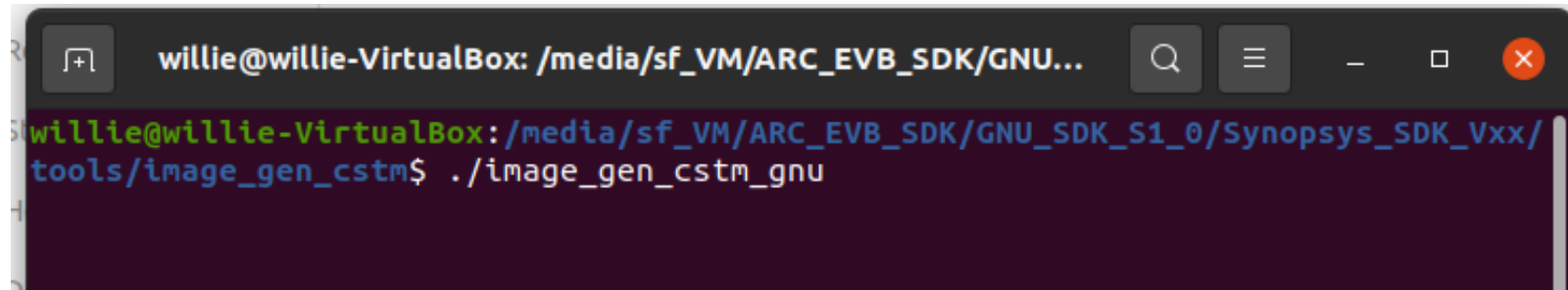
Make GNU Flash File (By image_gen_cstm_gnu)

1. You can also use “*image_gen_cstm_gnu*” manually.

Open terminal in VirtualBox Ubuntu, and go to the same path:

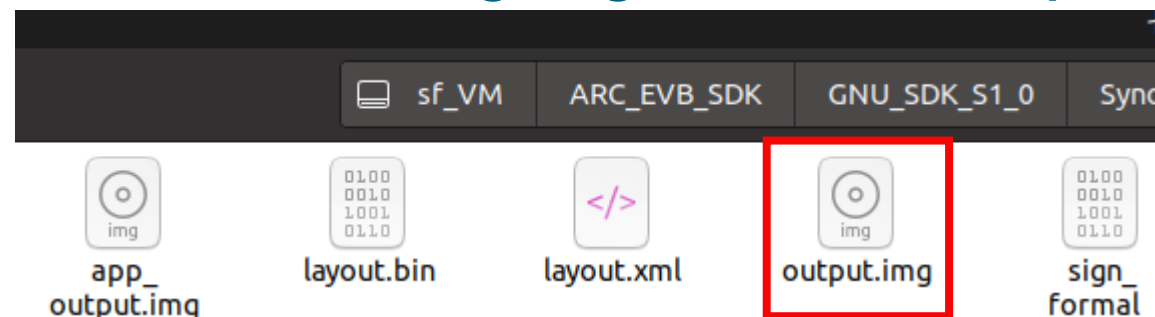
“...../Synopsys_SDK_Vxx/tools/image_gen_cstm”

\$./image_gen_cstm_gnu



2. Check your img file in

“...../Synopsys_SDK_Vxx/tools/image_gen_cstm/output/output.img”



Make Metaware Flash File (By image_gen_cstm)

1. If your toolchain is Metaware, you should use *“image_gen_cstm”* to convert elf and map file to image file.

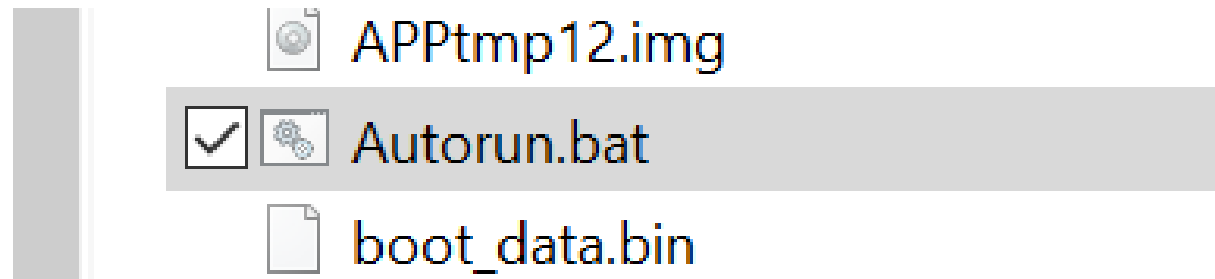
Use Windows and go to *“Synopsys_SDK_Vxx/tools/image_gen_cstm/”*
Make sure the files bellow are in your *“input”* folder.

“WEI_FW_mw_arcem9d_wei_r16.elf”

“WEI_FW_mw_arcem9d_wei_r16.map”

(They should be copied automatically after you compiled)

2. Execute *“Autorun.bat”*



Make Metaware Flash File (By image_gen_cstm)

3. Type command "2"

You will get "*output.img*" in "*output*" folder after convert successful.

C:\WINDOWS\system32\cmd.exe

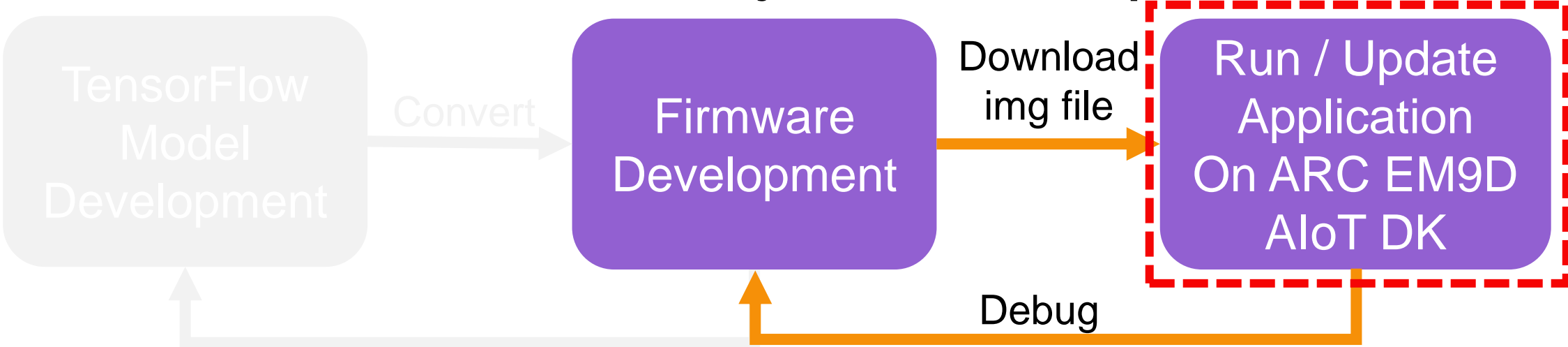
```
#####
Please choice one
1. Exit
2. LQFP128 r16 HX6539-A04TLDG-0001K Debug BLp
3. LQFP128 r16 HX6539-A04TLDG-1A11K Debug BLp
4. LQFP128 r16 HX6539-A04TLDG-1111K Debug BLp
5. QFN72 r16 HX6537-A09TDIG-1A11K Debug BLp
6. QFN72 r16 HX6537-A09TDIG-1111K Debug BLp
7. LQFP128 r16 HX6539-A04TLDG-1A11S Debug BLw
8. LQFP128 r16 HX6539-A04TLDG-1111S Debug BLw
9. QFN72 r16 HX6537-A09TDIG-1A11S Debug BLw
10. QFN72 r16 HX6537-A09TDIG-1111S Debug BLw
11. WLCSP38 r16 HX6540-A01TWA-1111S Debug BLw
12. None, Create new list(DOTO NOT Finish yet)
#####
Load setting.txt Success - 11 lines.
EnterVar:2
```

```
RunBLp...
signtool_odm.exe sign -type BLp -rsa pkcs
_rsa_key.der -cert odm_key¥cert1_rsa.bin -
ormal_qqq.img
RunBLp...
signtool_odm.exe sign -type BLp -rsa pkcs
_rsa_key.der -cert odm_key¥cert1_rsa.bin -
ile output¥sign_formal_layout.bin
ReorderXML
GenWholeImage
Total image size= 699 KB( 0xae30 )
Generate Image Done
output¥app_output.img
output¥output.img
2 file(s) copied.
output¥layout.bin
output¥sign_formal_layout.bin
2 file(s) copied.
Finish~~
```

Update and Run Application on ARC EM9D AIoT DK

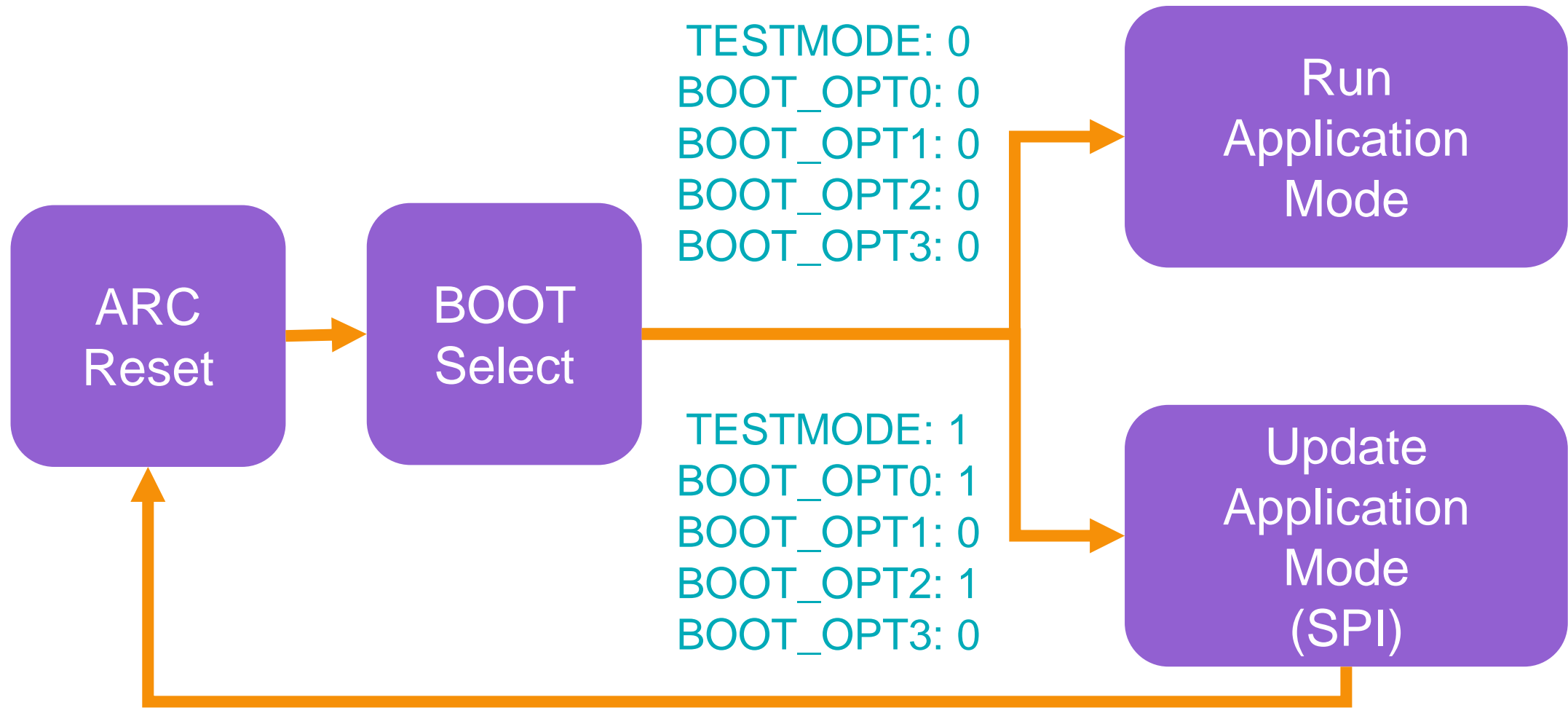


ARC EM9D AIoT DK Project Development Flow



Stage	TensorFlow Model Development	Firmware Development	Run / Update Application On ARC EM9D AIoT DK
Tool	Anaconda Cygwin	Cygwin Metaware or ARC GNU VirtualBox (Ubuntu 20.04)	JTAG Himax-FT4222-GUI USB Cable
Language	Python 3	C language C++ language	

Run / Update Application on ARC EM9D AIoT DK



Run / Update Application on ARC EM9D AIoT DK

1. Always modify SW1 as bellow
2. Short J20 and J11 for update application mode

TESTMODE: 1

BOOT_OPT0: 1

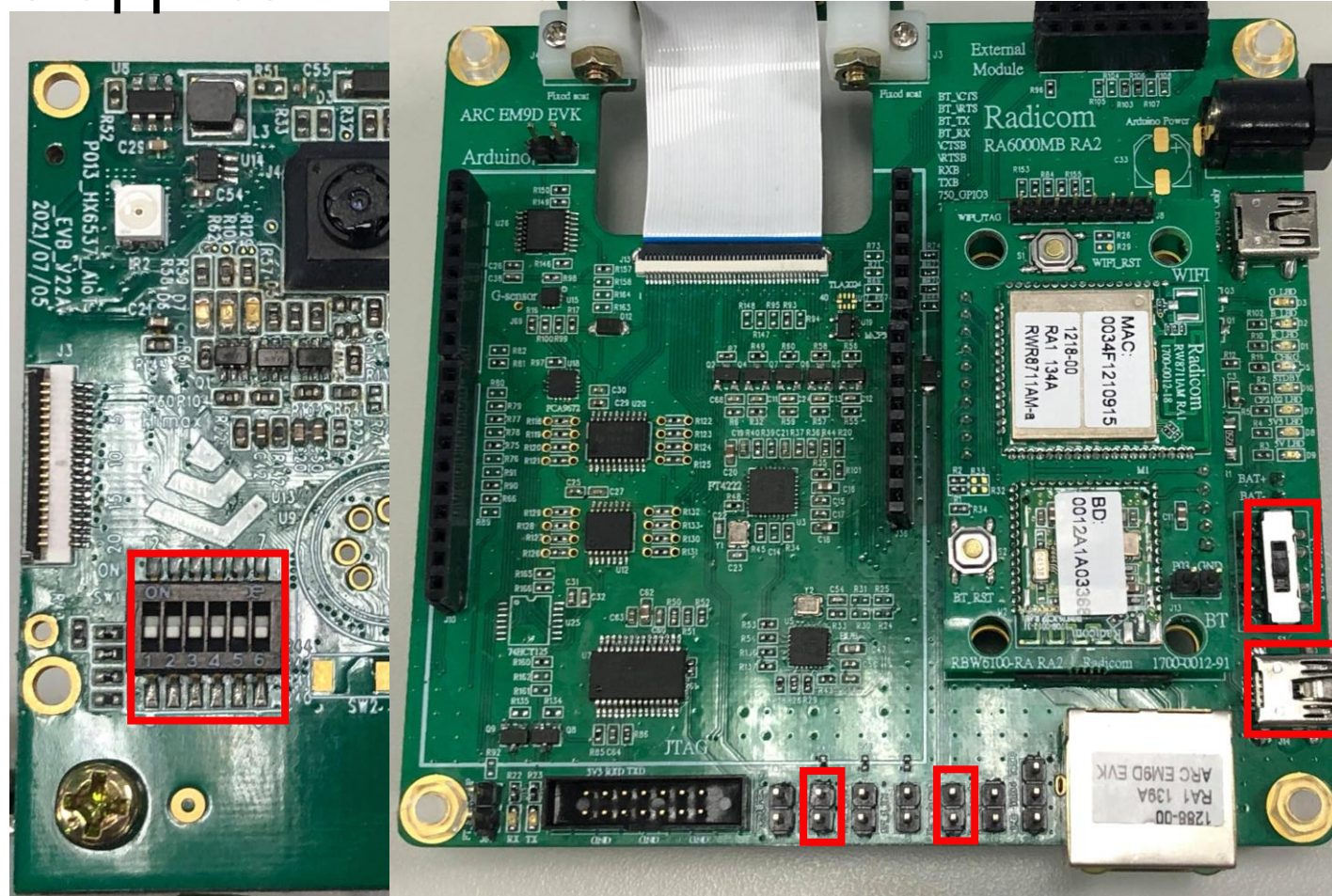
BOOT_OPT1: 0

BOOT_OPT2: 1

BOOT_OPT3: 0

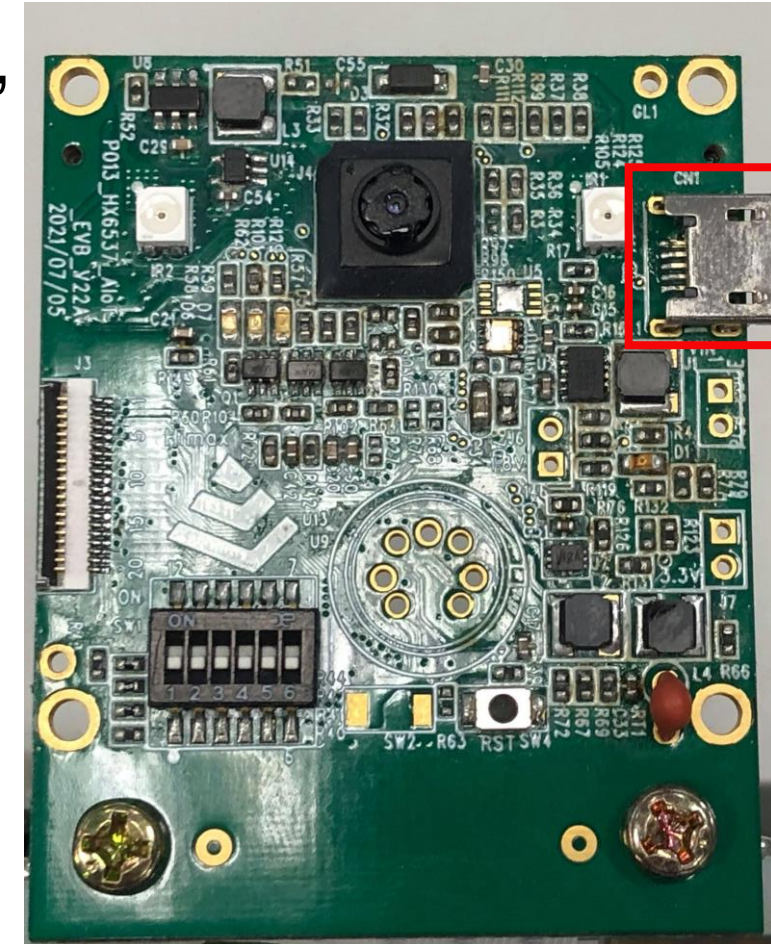
3. Turn power by S1

4. Connect USB Cable to J14



Update Application on ARC EM9D AIoT DK

5. Connect USB cable to CN1 for CPU board
 - This step is for some USB power source is lower than 5V.
 - If only connect USB cable to extension board, CPU may have fault during program flash.

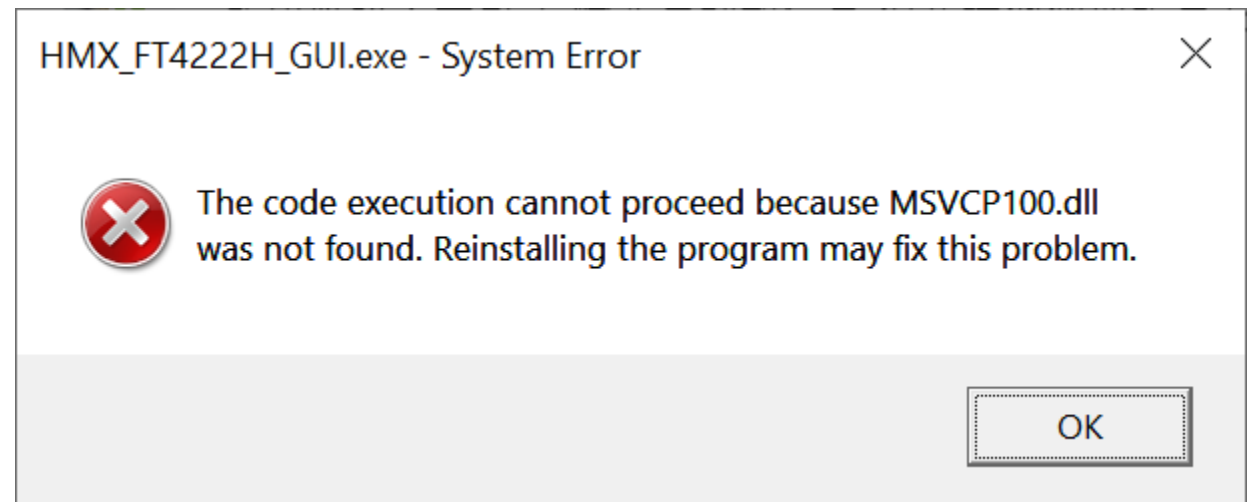


Update Application on ARC EM9D AIoT DK

6. Execute “*HMX_FT4222H_GUI.exe*” in
“*Synopsys_SDK_Vxx/tools/HMX_FT4222H_GUI/*”

If you have a warning message as left bellow, make sure the ARC EM9D AIoT DK has been connected to PC, and USB device in PC is mounted.

If USB device can't find correct driver, please refer to Appendix-2&3.
If your have DLL error as right bellow, please refer to Appendix-4.



Update Application on ARC EM9D AIoT DK

7. Select “*Flash download*” > “*Read ID*”
You will get “*ef/60/15*” if MCU is ready.
If the ID is incorrect or shows “*unknown Flash type!*”, make sure your boot-signals are right.

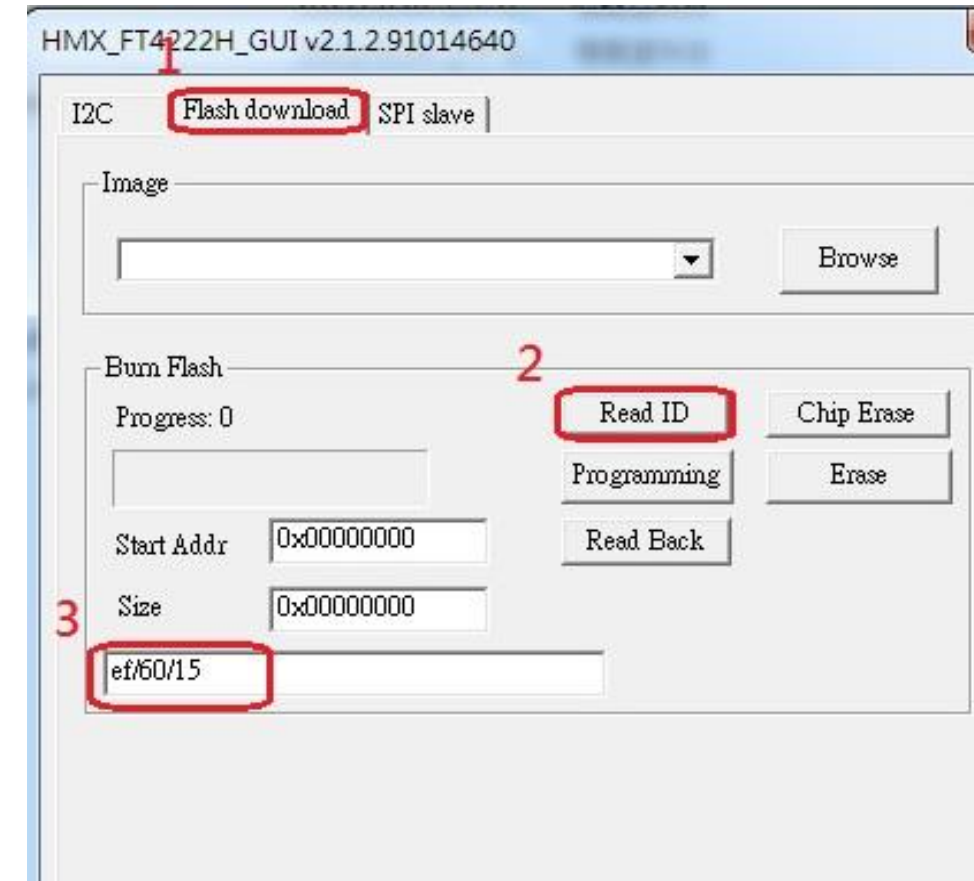
After that, you can power-up and press reset switch, then close and execute “*HMX_FT4222H_GUI.exe*” again.

HMX_FT4222H_GUI



unknown Flash type!

確定



Update Application on ARC EM9D AIoT DK

8. Click *“Browse”* to select your image file and *“Programming”*.
9. After programming finish, open J20 for running application.

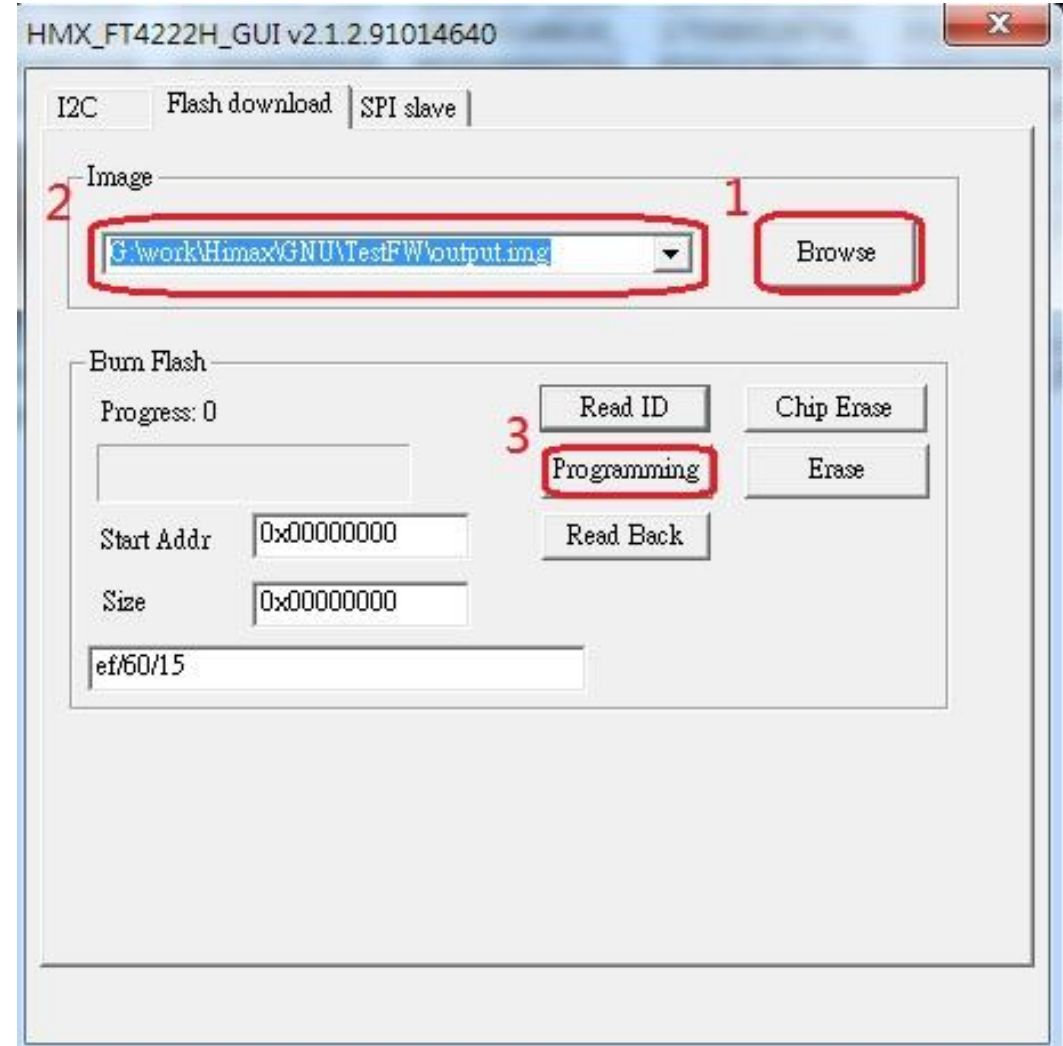
TESTMODE: 0

BOOT_OPT0: 0

BOOT_OPT1: 0

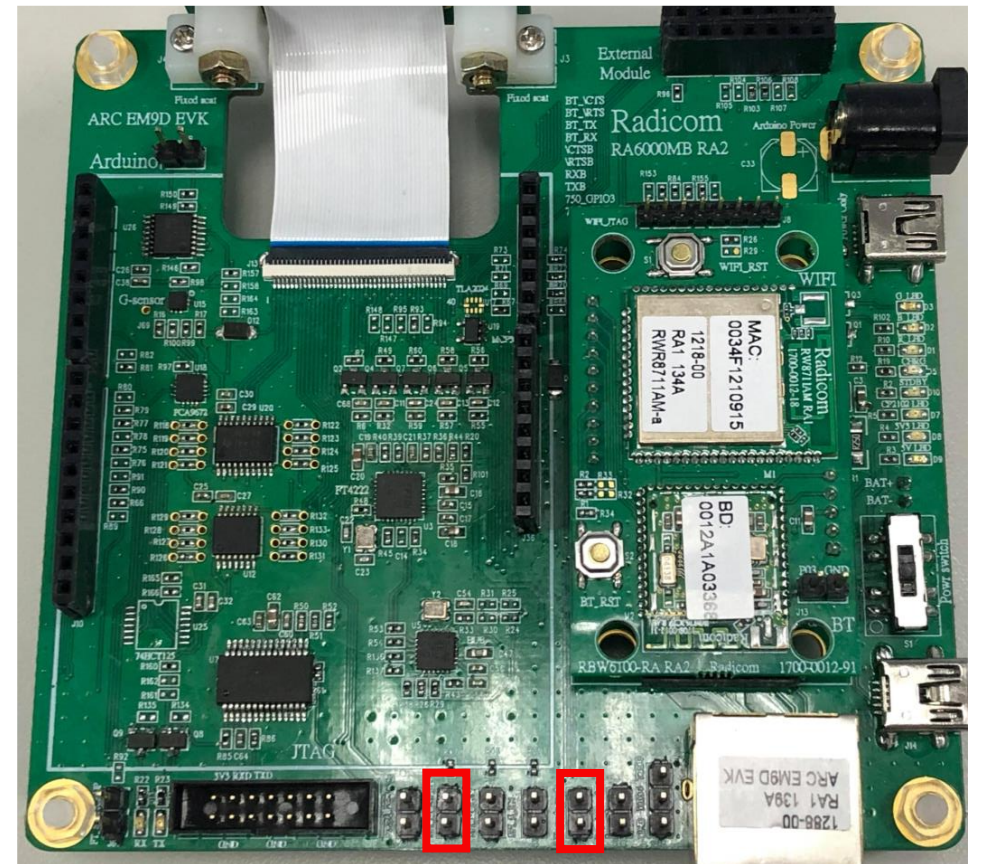
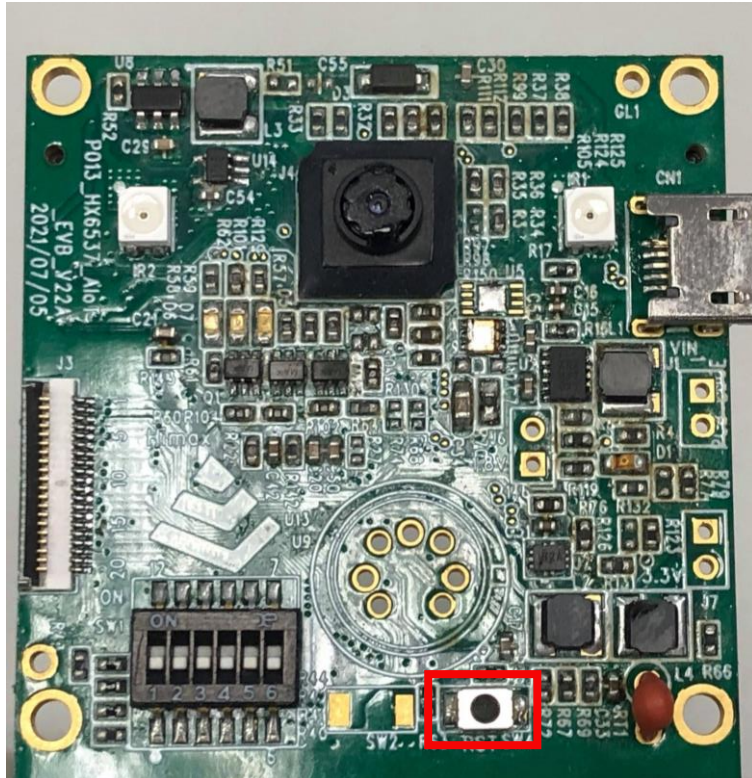
BOOT_OPT2: 0

BOOT_OPT3: 0



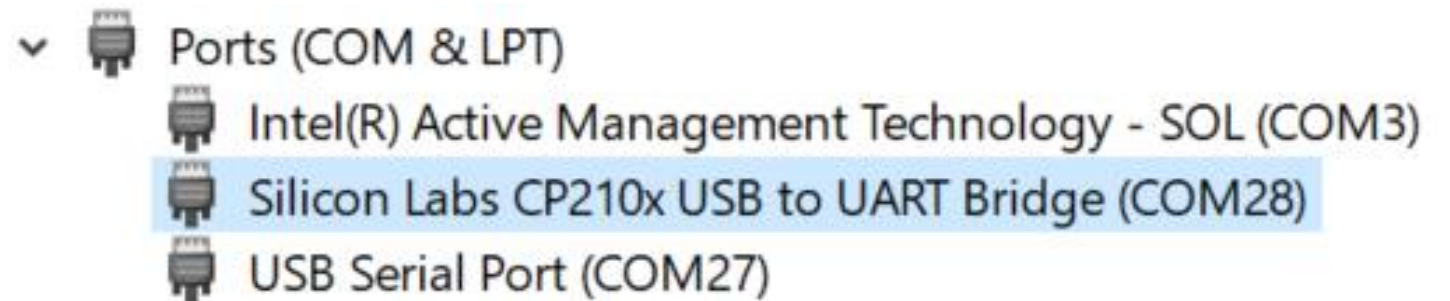
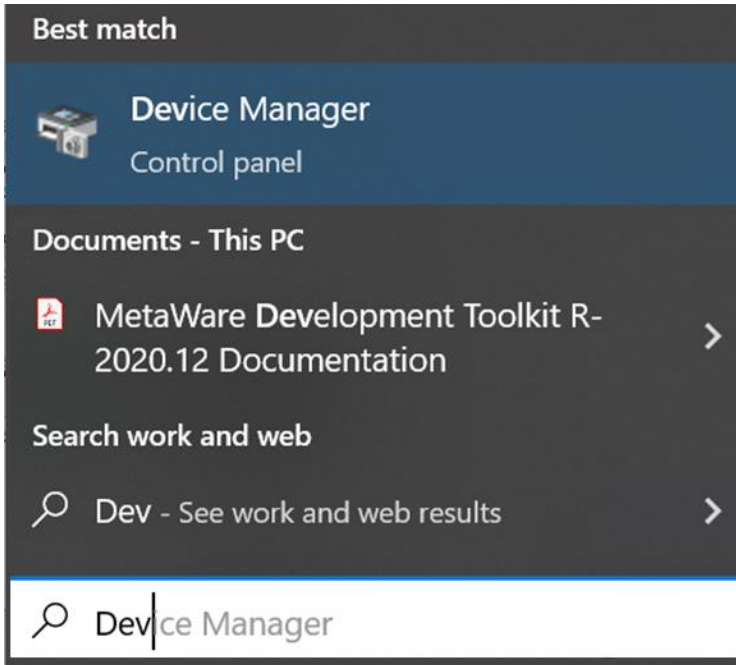
Run Application on ARC EM9D AIoT DK

1. Short J20 and J11 for update mode, open J20 for run mode
2. Press reset button SW4. MCU will start to run the application.
If MCU doesn't boot-up, make sure your boot-signals are set right, or you can try to power-up again.



Run Application on ARC EM9D AIoT DK

3. You can also use USB VCP to receive package from ARC EM9D AIoT DK.
4. Connect ARC EM9D AIoT DK and PC by USB Cable



Run Application on ARC EM9D AIoT DK

5. Check your ARC EM9D AIoT DK USB port number

We have 2 USB ports on ARC EM9D AIoT DK .

Uart0_CP2102: Silicon Labs CP210x USB to UART Bridge (COMx)

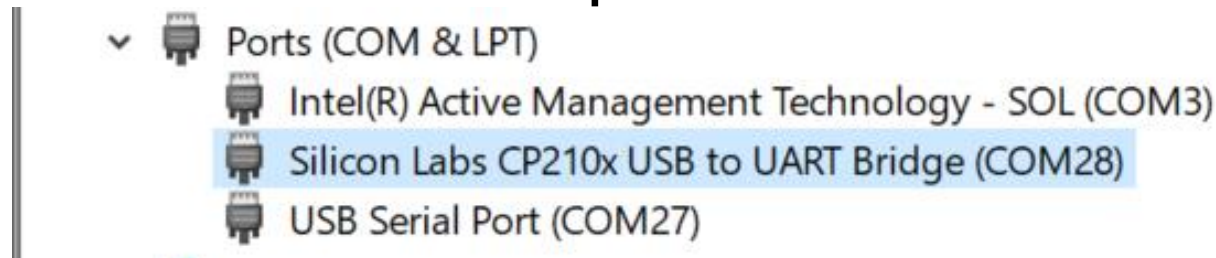
Uart1_FT232: USB Serial Port (COMx)

(If USB Serial Port is not shown here, please refer to Appendix-2 and Appendix-3)

6. Device Manager> Ports(COM & LPT) > CP210x USB to UART (COMx)

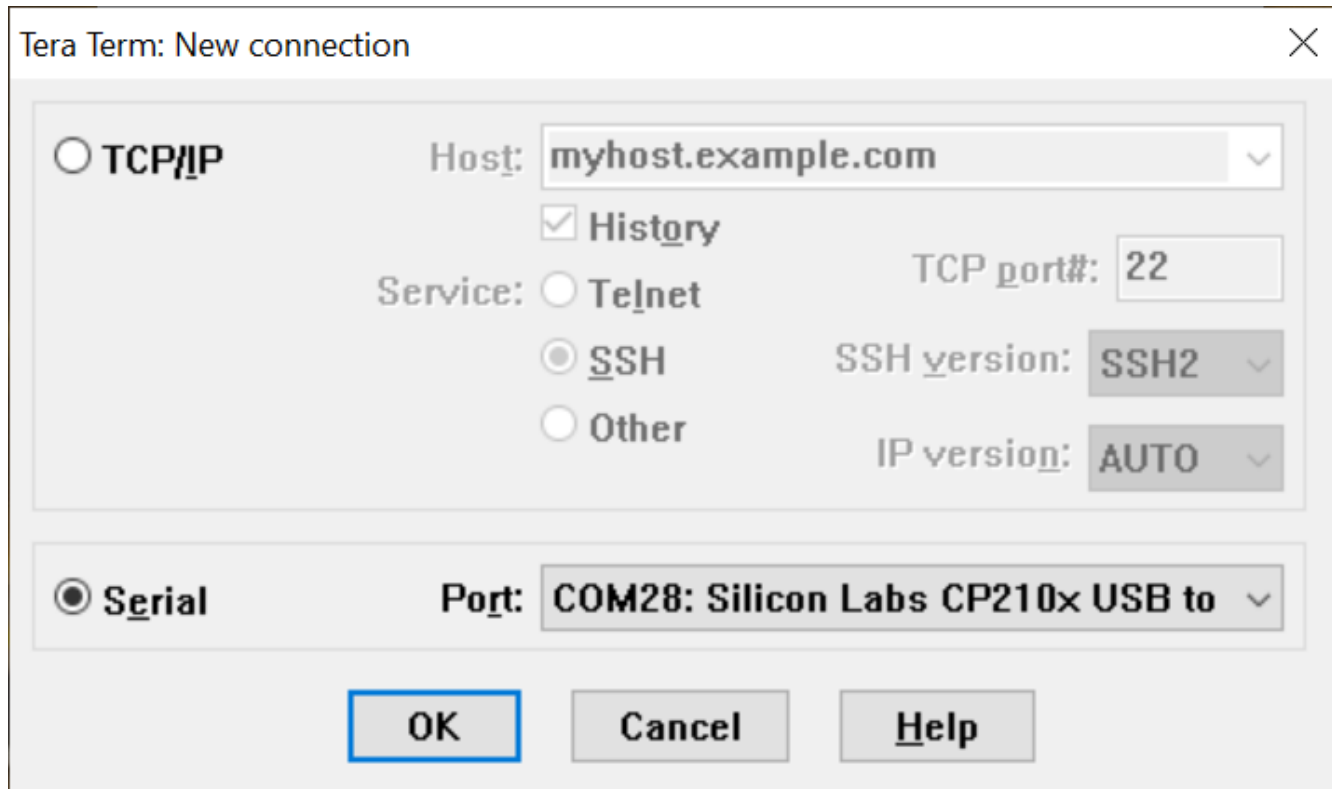
x: This is your ARC EM9D AIoT DK USB port number

Please choose the USB serial port with CP2102 by default.



Run Application on ARC EM9D AIoT DK

7. Open tera term and select “COM~~x~~: CP210x USB to UART (COM~~x~~)
 8. Tera term “*Setup*” > “*Serial Port*” > *Change Baud to 115200*
- No need to change other settings.



Tera Term: New connection

☐ TCP/IP

Host: myhost.example.com

☒ History

Service: ☐ Telnet ☒ SSH ☐ Other

TCP port#: 22

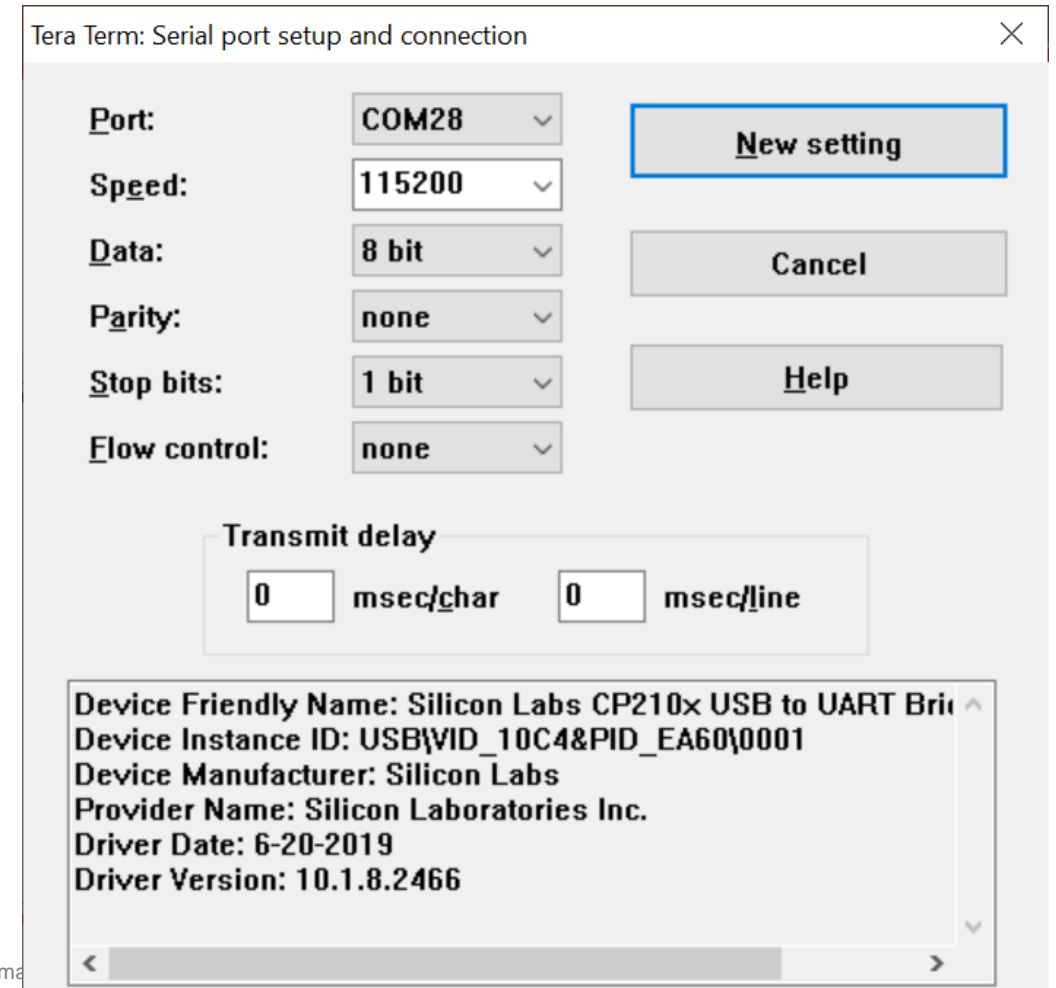
SSH version: SSH2

IP version: AUTO

☒ Serial

Port: COM28: Silicon Labs CP210x USB to

OK Cancel Help



Tera Term: Serial port setup and connection

Port: COM28

Speed: 115200

Data: 8 bit

Parity: none

Stop bits: 1 bit

Flow control: none

New setting

Cancel

Help

Transmit delay

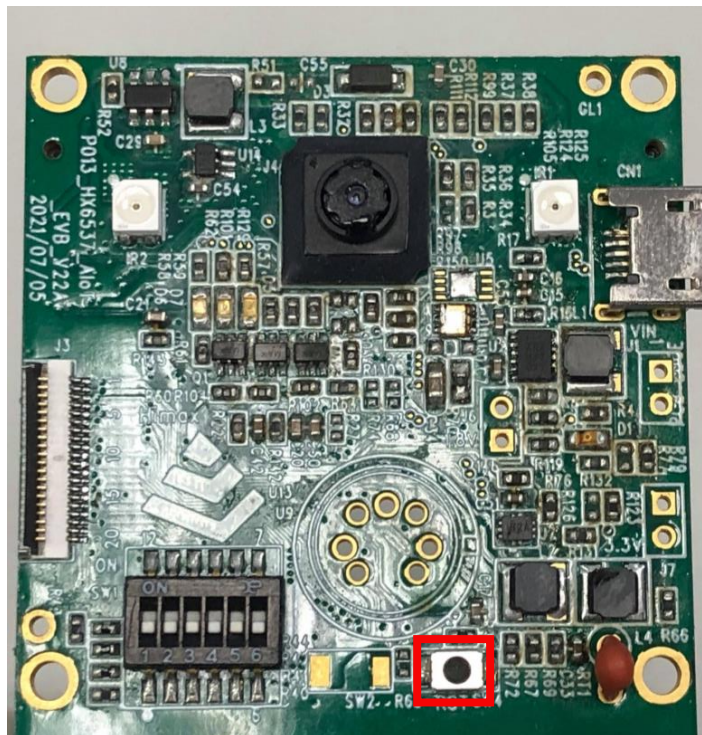
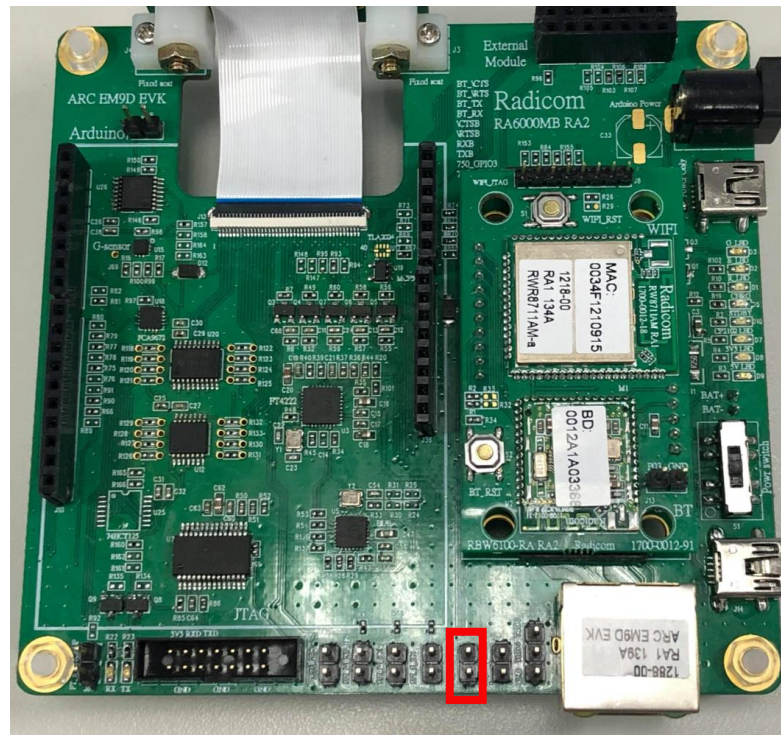
0 msec/char 0 msec/line

Device Friendly Name: Silicon Labs CP210x USB to UART Bri
Device Instance ID: USB\VID_10C4&PID_EA60\0001
Device Manufacturer: Silicon Labs
Provider Name: Silicon Laboratories Inc.
Driver Date: 6-20-2019
Driver Version: 10.1.8.2466

Run Application on ARC EM9D AIoT DK

9. Open J20 for run mode

10. Press reset button SW4. MCU will reset and run the application



```
Himax HEI Boot loader

-----
enbARC Build Time: Jan  4 2021, 13:44:14
Compiler Version: MetaWare, 4.2.1 Compatible Clang 8.0.1
Boot loader Version : 1.4.4 (Date:Jan  4 2021)
chip version : 0x8535a1
cpu speed : 400000000 hz
spi speed : 50000000 hz
wake up evt:4
...secure lib version = 352380df9a347b1187d2361bfcd4455178a1ebcb
1st APPLICATION addr[31]=21000 (main-1966)
Bootloader Done !!!!!
jump to app FW : 0x10000004
12 bytes lost due to alignment. To avoid this loss, please make s
HM036D RevB,C,D Config
person score:-2 no person score 2
person score:-6 no person score 6
```

Appendix-2: Troubleshooting - FTDI VCP Driver



Troubleshooting – FTDI VCP Driver

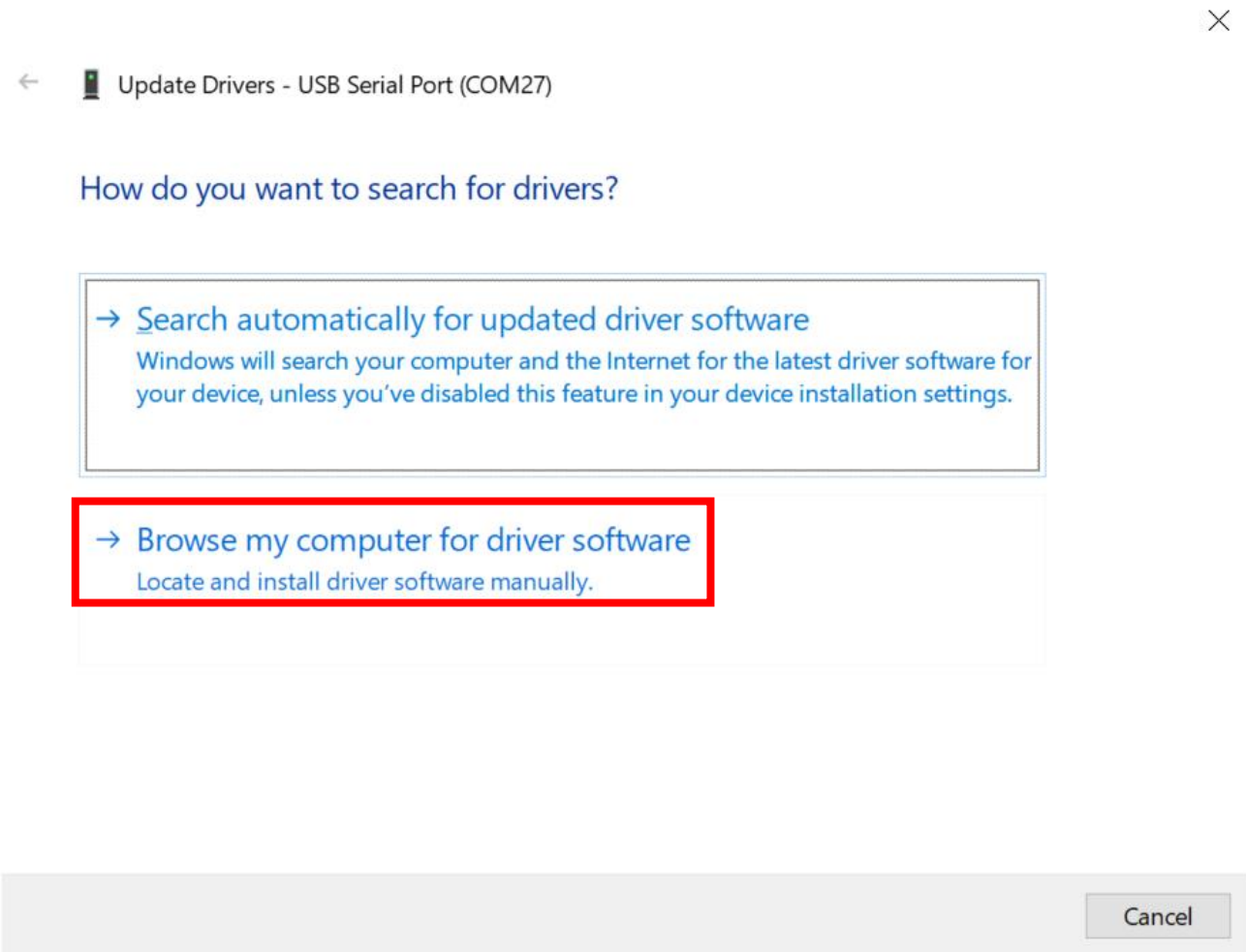
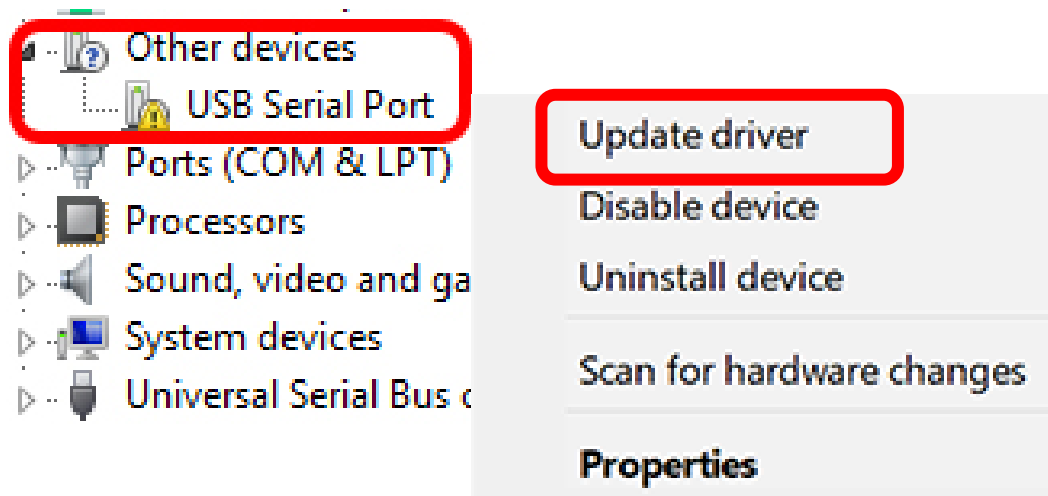
If the USB serial port is not shown in Ports (COM & LPT):

1. Download VCP driver: <https://ftdichip.com/drivers/vcp-drivers/>
Select Windows/X64 version
2. Unzip the downloaded file (CDM v2.XX.XX WHQL Certified)


Operating System	Release Date	Processor Architecture				
		X86 (32-Bit)	X64 (64-Bit)	PPC	ARM	MIPSII
Windows*	2017-08-30	2.12.28	2.12.28	-	-	-
Linux	-	-	-	-	-	-

Troubleshooting – FTDI VCP Driver

3. Click Device Manager > Other devices > USB Serial Port > Update driver
4. Choose “Browse my computer for driver software”



Troubleshooting – FTDI VCP Driver

←  Update Drivers - USB Serial Port (COM27) ×

[Browse for drivers on your computer](#) 5. Choose the downloaded folder (CDM v2.XX.XX WHQL Certified)

Search for drivers in this location:

☒ Include subfolders

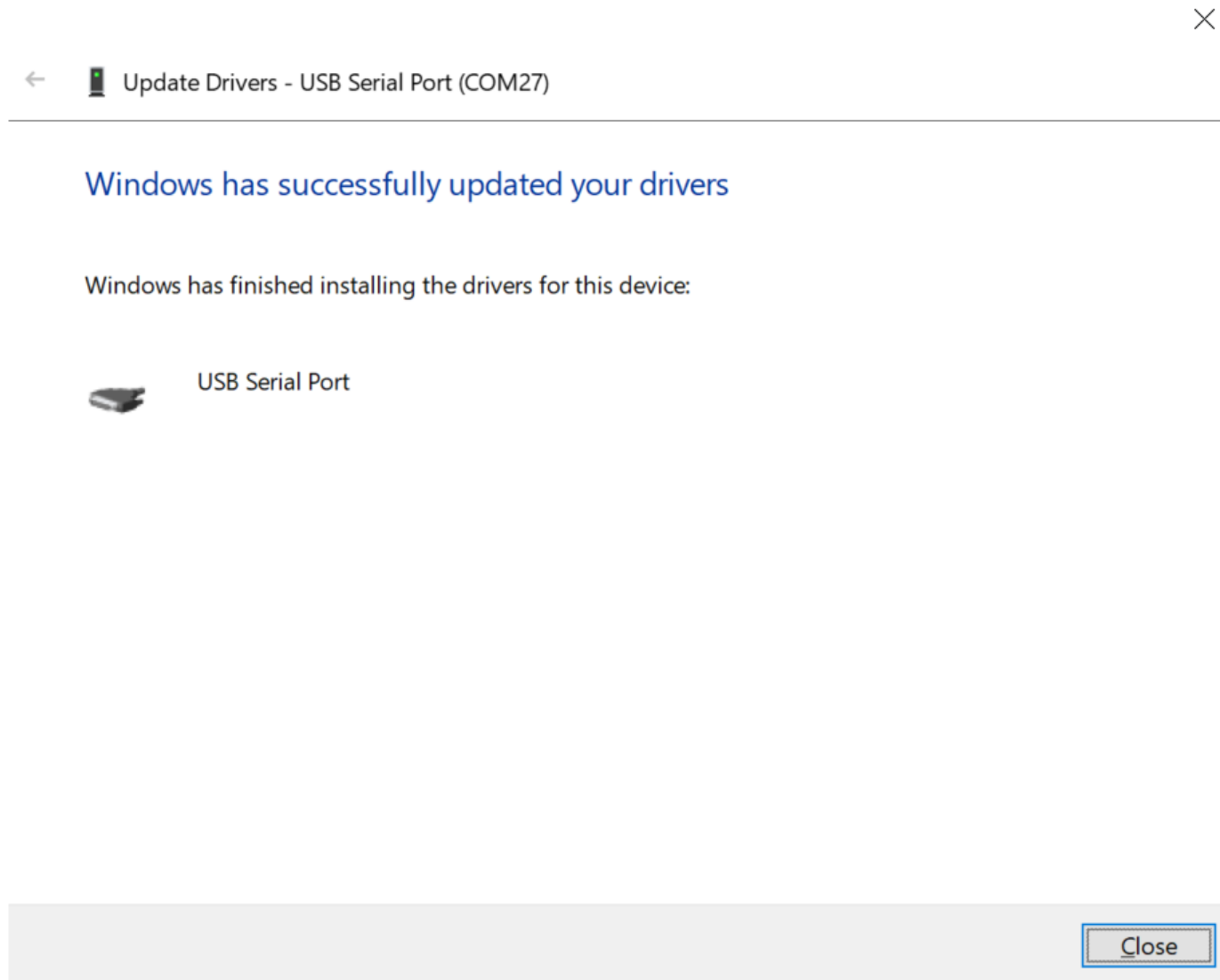
6. Select

→ [Let me pick from a list of available drivers on my computer](#)
This list will show available drivers compatible with the device, and all drivers in the same category as the device.

7.

Troubleshooting – FTDI VCP Driver

8. Finish



Appendix-3: Troubleshooting - CP2102 VCP Driver



Troubleshooting – CP2102 VCP Driver

If the USB serial port is not shown in Ports (COM & LPT):

1. Download VCP driver:

<https://www.silabs.com/developers/usb-to-uart-bridge-vcp-drivers>

Select CP210x VCP Windows version

2. Unzip the downloaded file (CP210x_VCP_Windows)

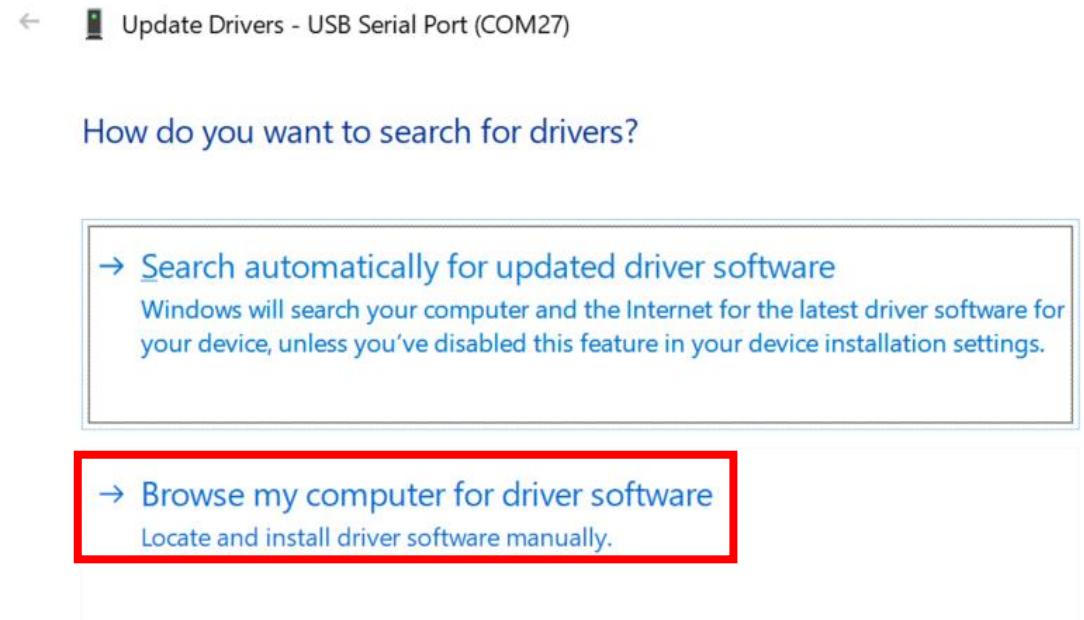
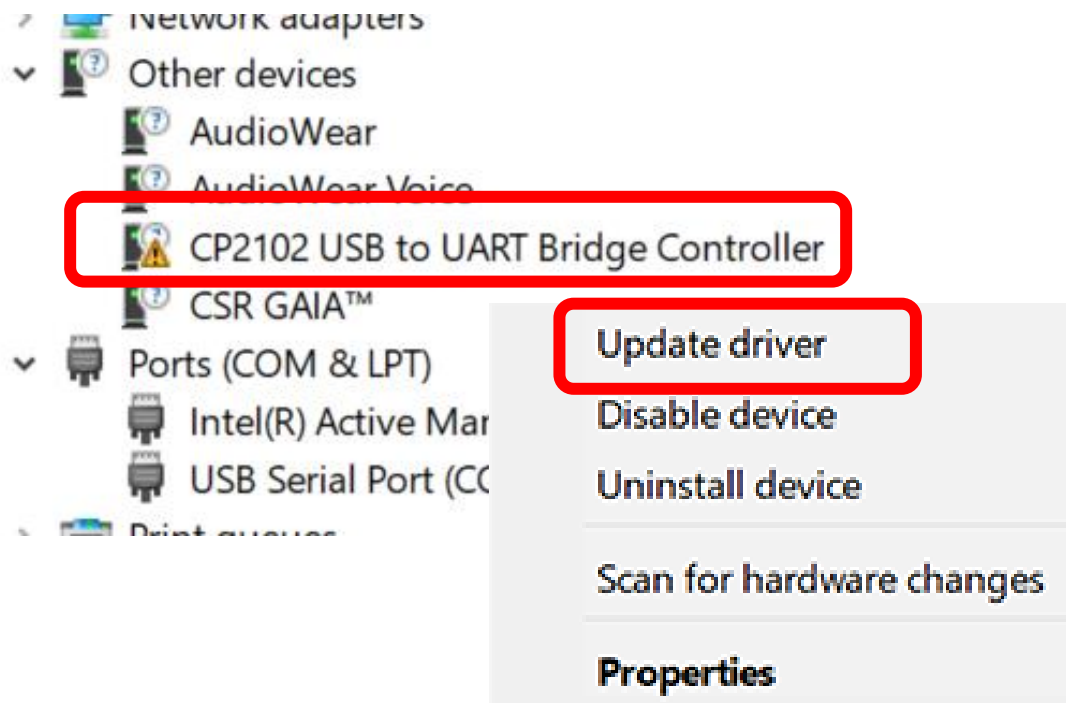
Software • 10

CP210x Universal Windows Driver	v11.1.0 3/22/2022
CP210x VCP Mac OSX Driver	v6.0.2 10/27/2021
CP210x Windows Drivers	v6.7.6 9/4/2020
CP210x Windows Drivers with Serial Enumerator	v6.7.6 9/4/2020
CP210x_5x_AppNote_Archive	9/4/2020

[Show 5 more Software](#)

Troubleshooting – CP2102 VCP Driver

3. Click Other devices > USB Serial Port > Update driver
4. Choose “Browse my computer for driver software”



Troubleshooting – CP2102 VCP Driver



← Update Drivers - CP2102 USB to UART Bridge Controller

[Browse for drivers on your computer](#)

5. Choose the downloaded folder
(CP210x_VCP_Windows)

Search for drivers in this location:

C:\Users\williet\Downloads\CP210x_VCP_Windows

☒ Include subfolders

6. Select

→ [Let me pick from a list of available drivers on my computer](#)

This list will show available drivers compatible with the device, and all drivers in the same category as the device.

7.

Troubleshooting – CP2102 VCP Driver



8. Finish



Update Drivers - Silicon Labs CP210x USB to UART Bridge (COM28)

The best drivers for your device are already installed

Windows has determined that the best driver for this device is already installed. There may be better drivers on Windows Update or on the device manufacturer's website.



Silicon Labs CP210x USB to UART Bridge

→ [Search for updated drivers on Windows Update](#)

Close

Appendix-4: Troubleshooting – HMX_FT4222H_GUI.exe DLL File Missing



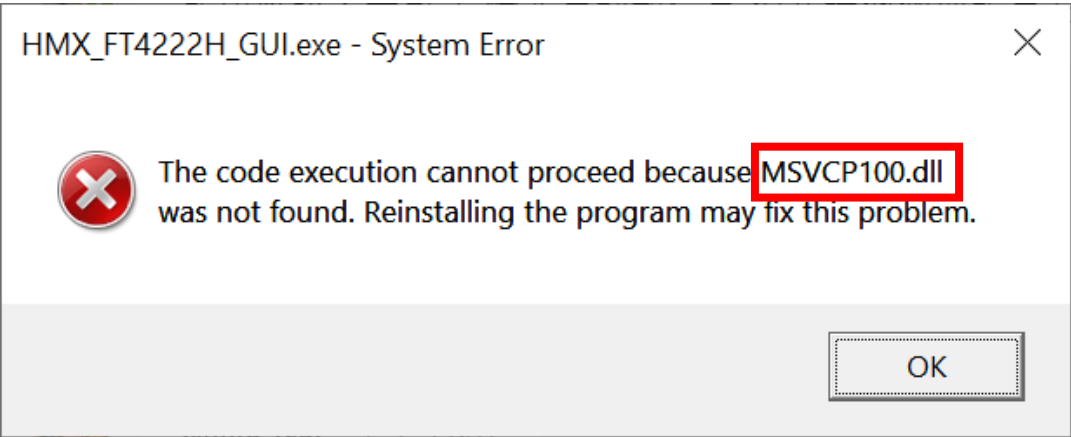
Troubleshooting – DLL File Missing

If the DLL file missing is ftd2xx.dll

- Download FTDI D2XX Drivers install it.
<https://ftdichip.com/drivers/d2xx-drivers/>

Currently Supported D2XX Drivers:

Subscribe to Our Driver Updates



		Processor Architecture					
Operating System	Release Date	X86 (32-Bit)	X64 (64-Bit)	ARM	MIPS	SH4	Comments
Windows (Desktop)*	2021-07-15	2.12.36.4	2.12.36.4	2.12.36.4A *****	–	–	WHQL Certified. Includes VCP and D2XX. Available as a setup executable . Please see the Release Notes and Installation Guides .
Windows (Universal)****	2021-11-12	2.12.36.4U	2.12.36.4U	–	–	–	WHQL Certified. Includes VCP and D2XX.

Troubleshooting – DLL File Missing

If the DLL file missing is MSVCP100.dll or MSCR100.dll

- Download Microsoft Visual C++ 2010 and install it.
<https://www.microsoft.com/en-us/download/details.aspx?id=26999>

If the DLL file missing is MFC140.dll

- Download Microsoft Visual C++ 2015 and install it.
<https://www.microsoft.com/en-us/download/details.aspx?id=48145>

If missing other DLL files, please search on the internet

