

# **CSC 456/591**

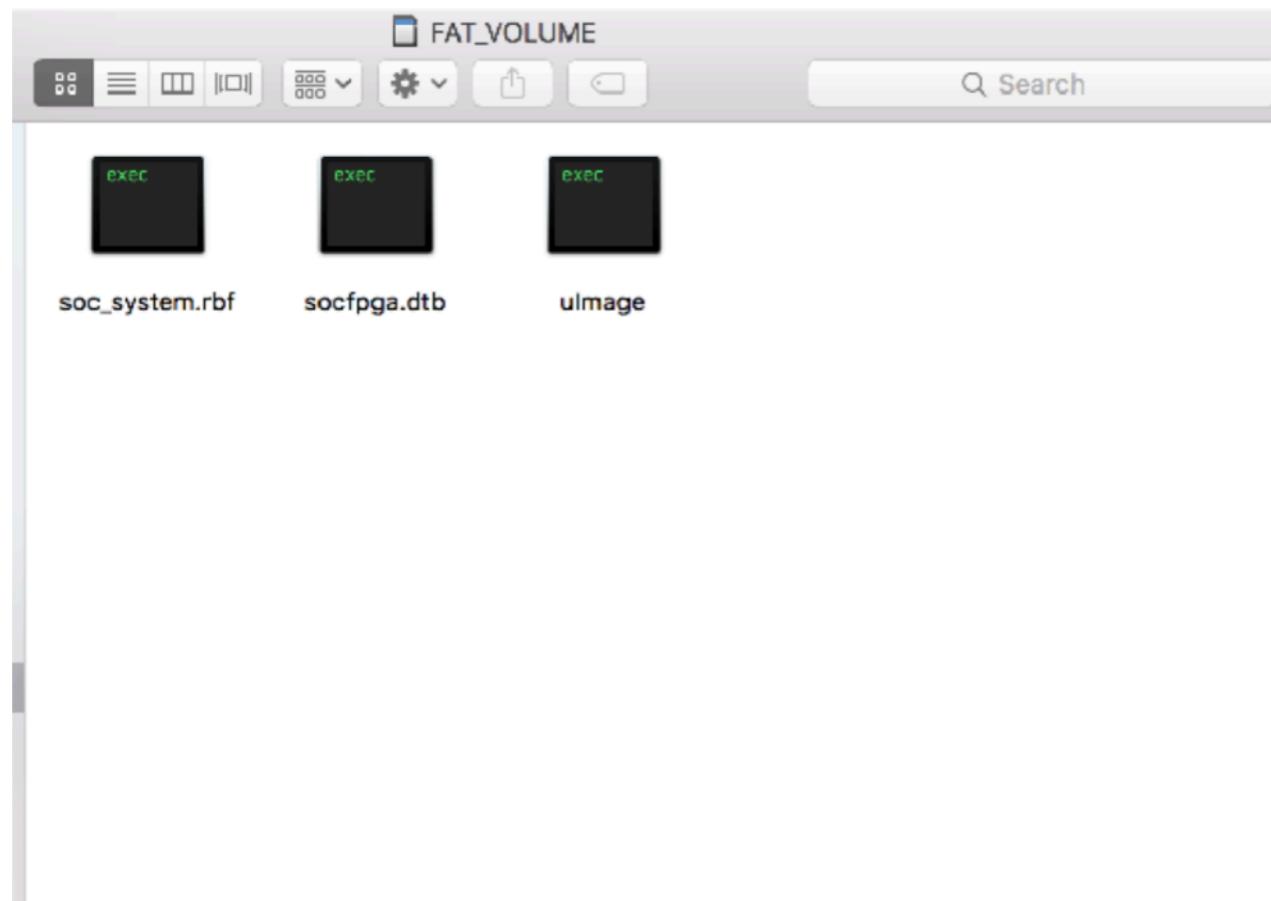
# **Tutorial of DE1-SoC**

**Zhengyi Qiu**

# Outline

- 1. Create MicroSD card image**
- 2. Connect to DE1-SoC**
- 3. Transfer files to De1-SoC**
- 4. Using VCL**
- 5. Demo**

# Create SD card Image



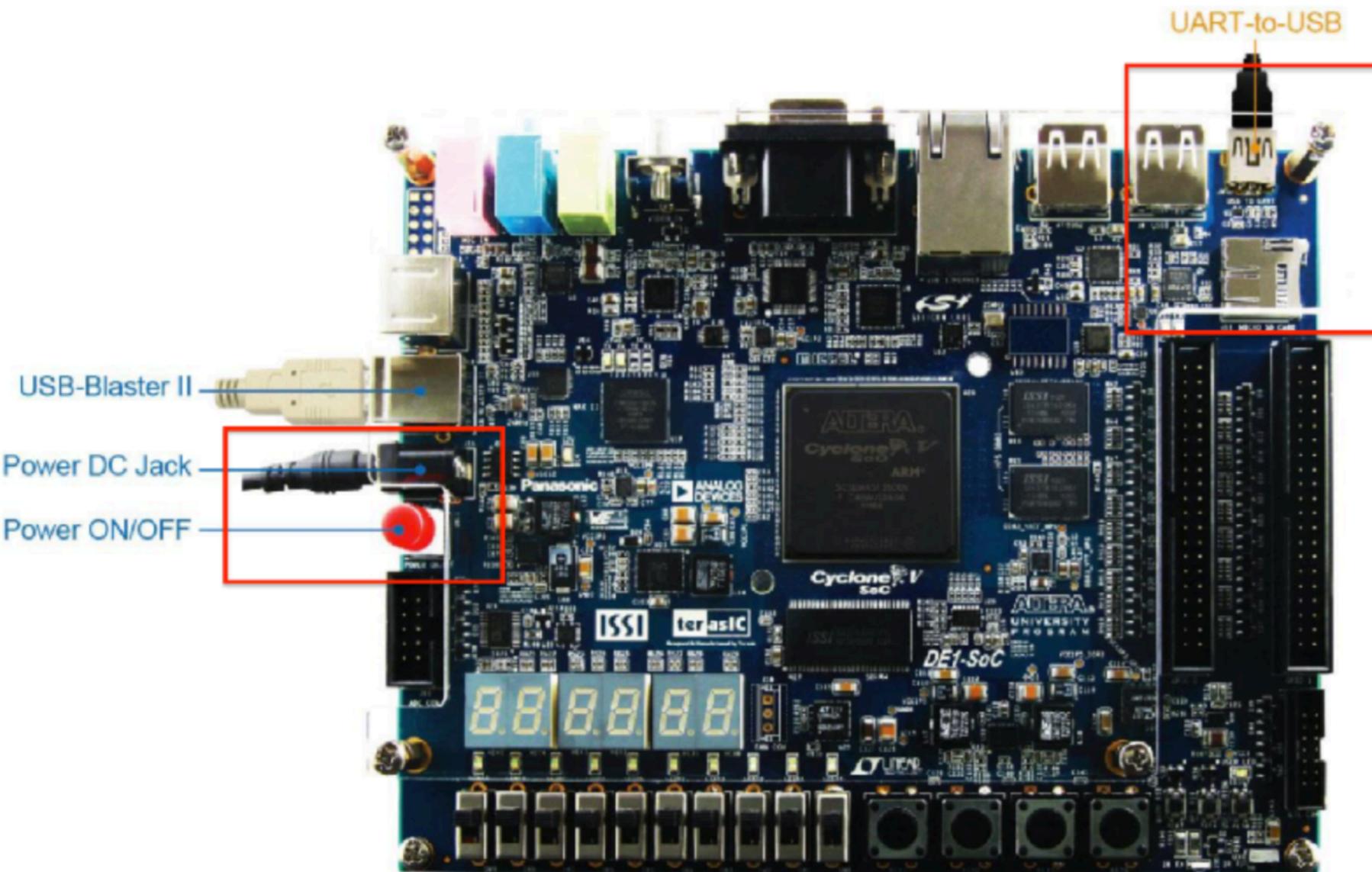
**Download the Linux image from**

**[https://drive.google.com/file/d/1zYaghec\\_3\\_7DdQODqQ3aINN\\_7tWnHuKz/view](https://drive.google.com/file/d/1zYaghec_3_7DdQODqQ3aINN_7tWnHuKz/view)**

**Choose a tool to write the image to the SD card**

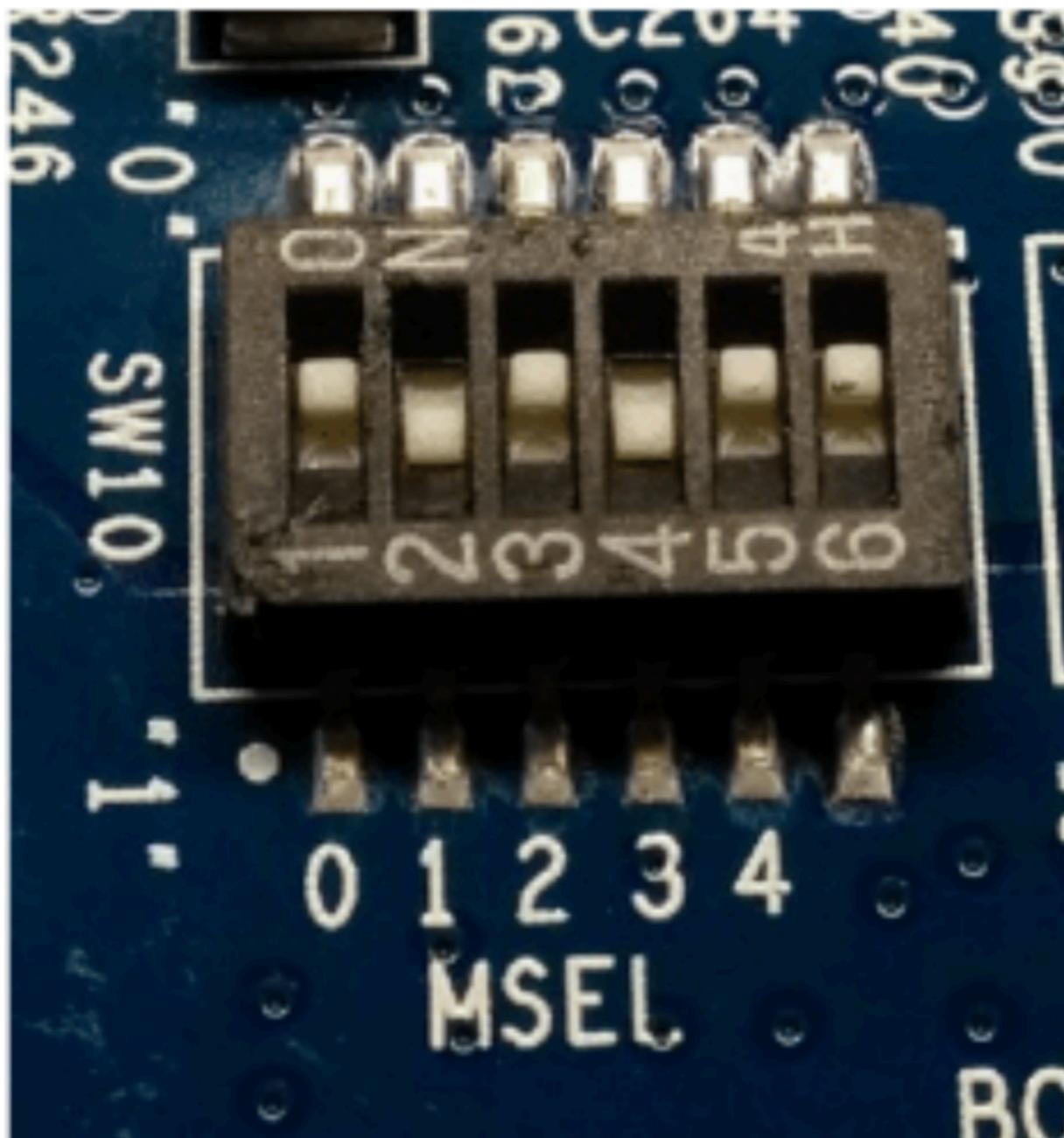
**When the write finishes, you should see these three files in the SD card**

# Connect to DE1-SoC



Plug in the power jack  
Plug in the USB port  
Put sd card in the slot

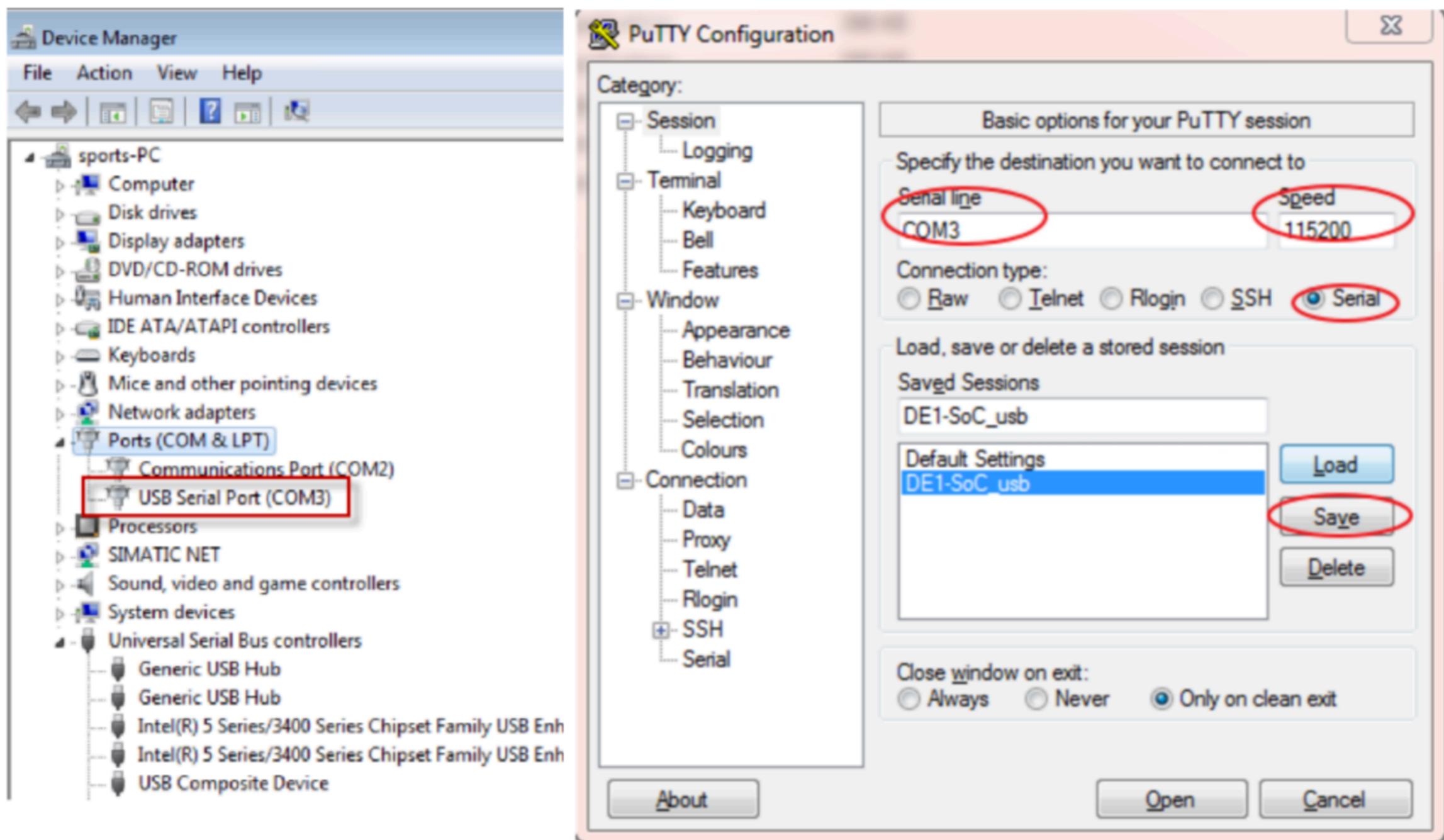
# Connect to DE1-SoC



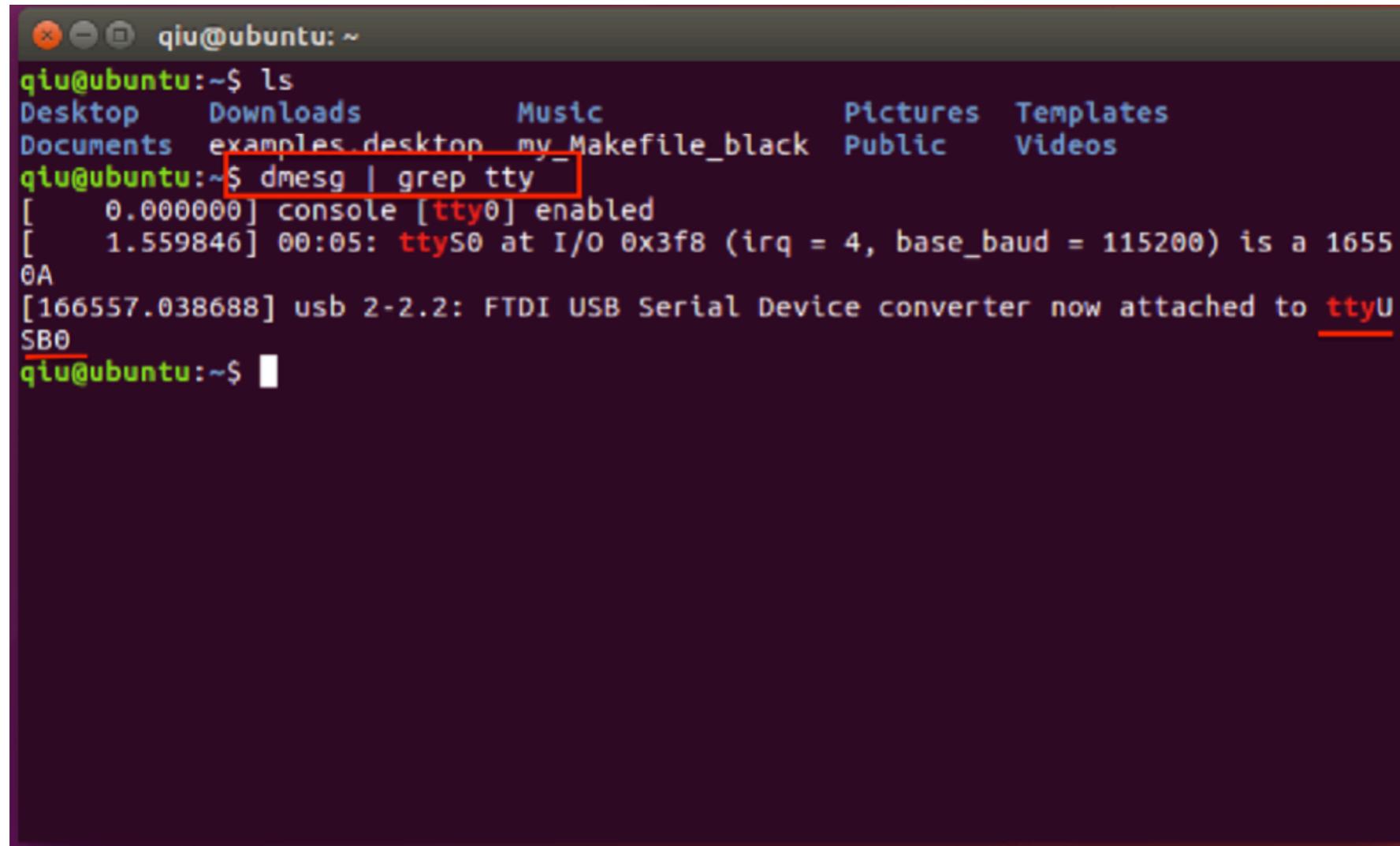
configure the **MSEL** switch at the back of the board before booting

only 2 and 4 are turned to 1

# Connect to DE1-SoC in Windows



# Connect to DE1-SoC in Linux



```
qi@ubuntu:~$ ls
Desktop  Downloads  Music  Pictures  Templates
Documents examples.desktop my_Makefile_black Public  Videos
qi@ubuntu:~$ dmesg | grep tty
[    0.000000] console [tty0] enabled
[ 1.559846] 00:05: ttyS0 at I/O 0x3f8 (irq = 4, base_baud = 115200) is a 1655
0A
[166557.038688] usb 2-2.2: FTDI USB Serial Device converter now attached to ttyU
SB0
qi@ubuntu:~$
```

If you are using mac, you can create a linux vm.

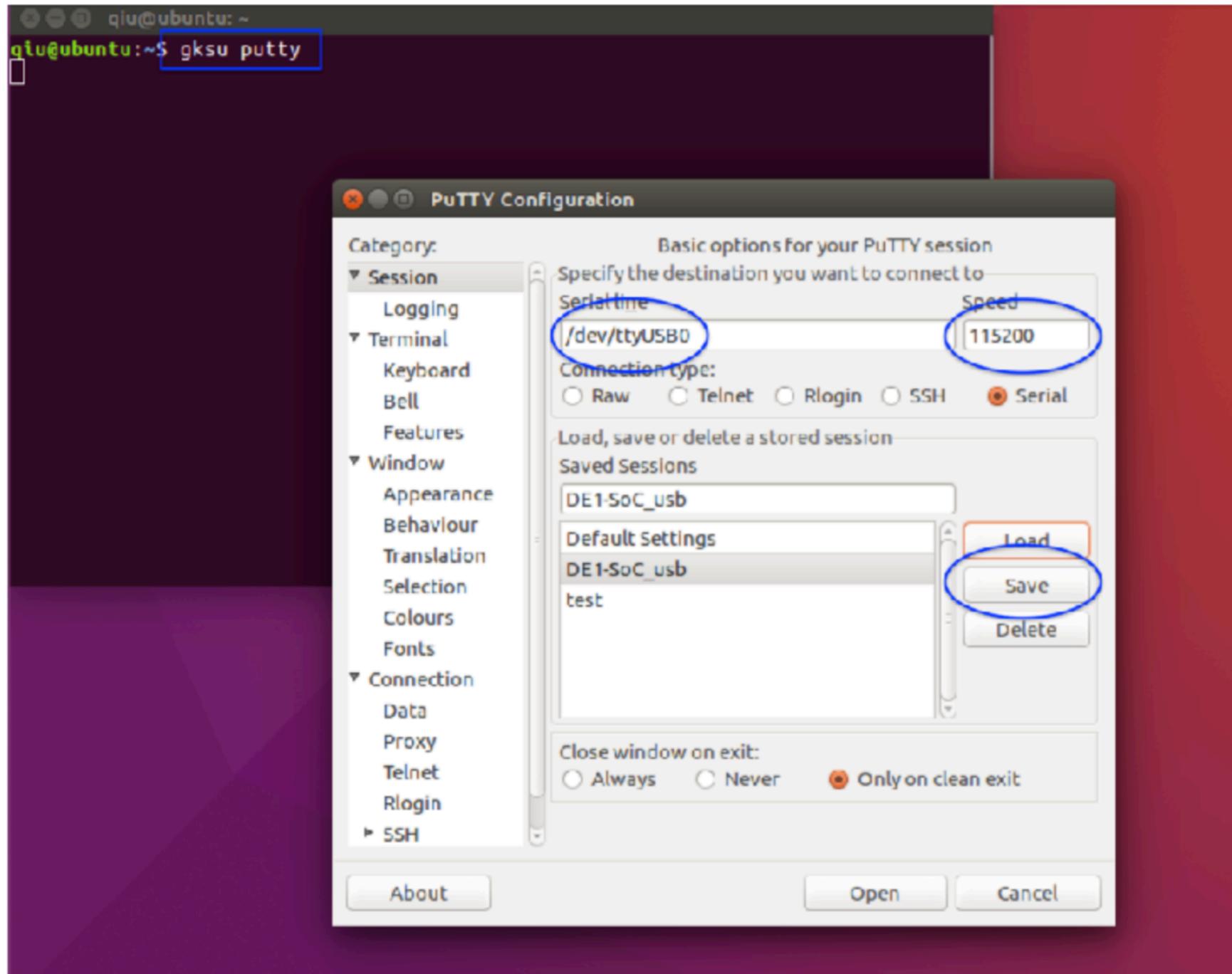
Get the device ID

use command:

**dmesg | grep tty**

The device ID in the figure is **ttyUSB0**

# Connect to DE1-SoC in Linux



Connect to board

use command:

**putty (might need sudo)**

# Connect to DE1-SoC



**When boot successfully,  
you will see this window**

# Transfer files to DE1-SoC using SD card

You can drag your file to the SD card directly.

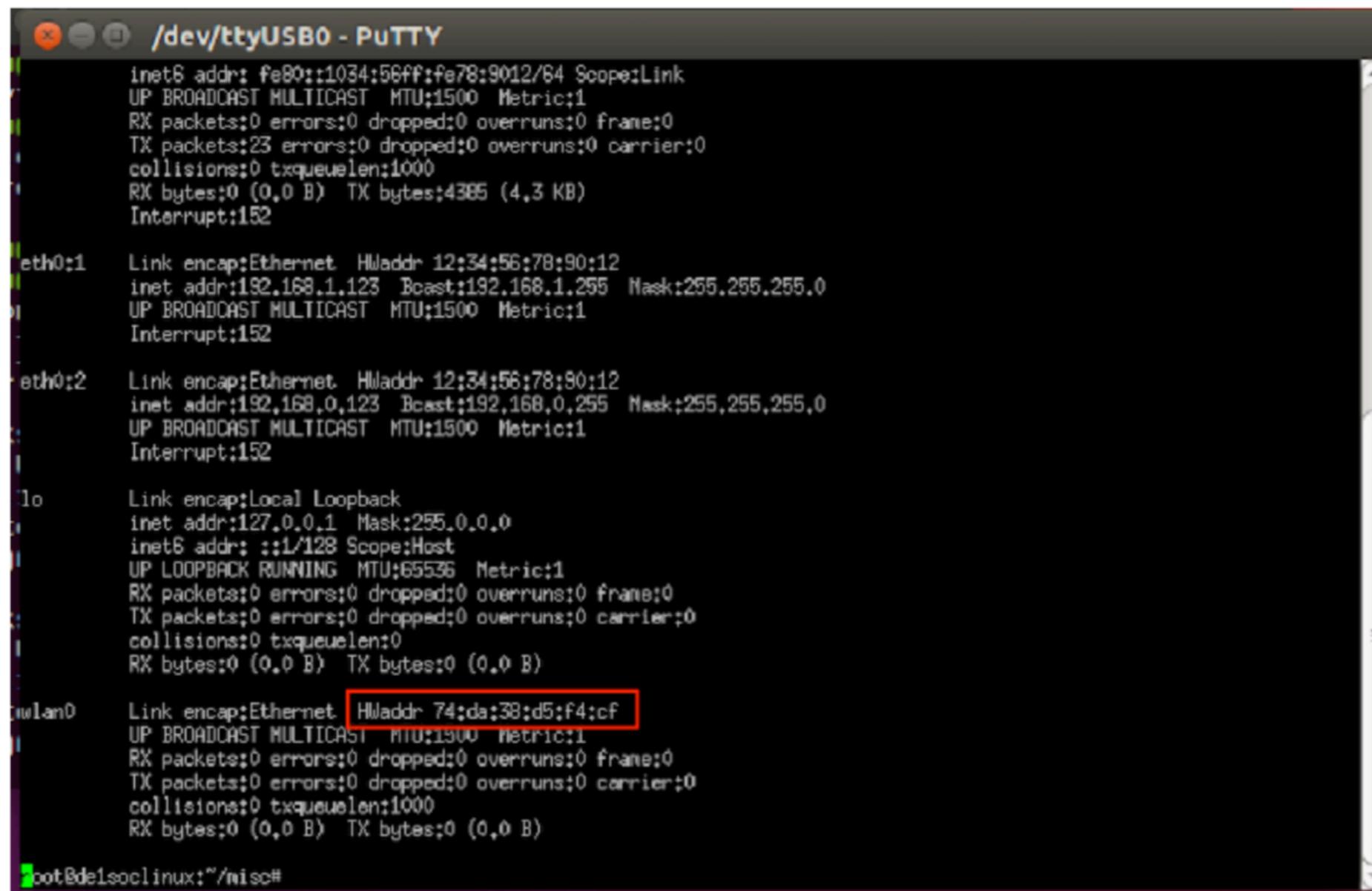
After put SD card in the slot on the board and correctly boot the board, the files can be accessed in `/media/fat_partition` directory.

use command:

`cd /media/fat_partition`

and copy files to your desired position

# Transfer files to DE1-SoC using SSH



The screenshot shows a PuTTY terminal window titled '/dev/ttyUSB0 - PuTTY'. The window displays the output of the 'ifconfig' command. The output lists several network interfaces:

- inet6 addr: fe80::1034:56ff:fe78:9012/64 Scope:Link  
UP BROADCAST MULTICAST MTU:1500 Metric:1  
RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
TX packets:23 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1000  
RX bytes:0 (0.0 B) TX bytes:4385 (4.3 KB)  
Interrupt:152
- eth0:1 Link encap:Ethernet HWaddr 12:34:56:78:90:12  
inet addr:192.168.1.123 Bcast:192.168.1.255 Mask:255.255.255.0  
UP BROADCAST MULTICAST MTU:1500 Metric:1  
Interrupt:152
- eth0:2 Link encap:Ethernet HWaddr 12:34:56:78:80:12  
inet addr:192.168.0.123 Bcast:192.168.0.255 Mask:255.255.255.0  
UP BROADCAST MULTICAST MTU:1500 Metric:1  
Interrupt:152
- lo Link encap:Local Loopback  
inet addr:127.0.0.1 Mask:255.0.0.0  
inet6 addr: ::1/128 Scope:Host  
UP LOOPBACK RUNNING MTU:65536 Metric:1  
RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:0  
RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)
- wlan0 Link encap:Ethernet HWaddr 74:da:38:d5:f4:cf  
UP BROADCAST MULTICAST MTU:1500 Metric:1  
RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1000  
RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

The line 'wlan0 Link encap:Ethernet HWaddr 74:da:38:d5:f4:cf' is highlighted with a red box.

root@de1soclinux:/misc#

**type command:  
ifconfig**

# Transfer files to DE1-SoC using SSH

Introduction

Policy Agreement

Devices

Add Device

## Adding new device to container user: zqiu2

The device you register is uniquely identified by a MAC address. The MAC address is sometimes referred to as a physical address or a hardware address.

The pre-populated MAC address below reflects the device you are currently on.

MAC Address\*

A4:5E:60:E8:F8:5B

Ex: 00:01:02:03:A3:DE

Description\*

Ex: My Laptop

Policy Agreement\*

By adding this device, you are agreeing to be bound by the terms of the registration system **policy** agreement.

Add Device

Fields marked with a \* are required.

[https://nomad.ncsu.edu/nmr/dynamic/device\\_add.py](https://nomad.ncsu.edu/nmr/dynamic/device_add.py)

# Transfer files to DE1-SoC using SSH

```
root@de1soclinux:/misc# ifconfig  
  
lo      Link encap:Local Loopback  
        inet addr:127.0.0.1  Mask:255.0.0.0  
        inet6 addr: ::1/128 Scope:Host  
           UP LOOPBACK RUNNING MTU:65536 Metric:1  
           RX packets:4 errors:0 dropped:0 overruns:0 frame:0  
           TX packets:4 errors:0 dropped:0 overruns:0 carrier:0  
           collisions:0 txqueuelen:0  
           RX bytes:360 (360.0 B)  TX bytes:360 (360.0 B)  
  
wlan0    Link encap:Ethernet  HWaddr 74:da:38:d5:f4:cf  
       inet addr:10.139.70.5  Bcast:10.139.71.255  Mask:255.255.248.0  
        inet6 addr: fe80::76da:38ff:fed5:f4cf/64 Scope:Link  
           UP BROADCAST RUNNING MULTICAST  MTU:1500 Metric:1  
           RX packets:15060 errors:0 dropped:0 overruns:0 frame:0  
           TX packets:98 errors:0 dropped:0 overruns:0 carrier:0  
           collisions:0 txqueuelen:1000  
           RX bytes:2915631 (2.9 MB)  TX bytes:20015 (20.0 KB)  
  
root@de1soclinux:/misc#
```

Make sure connect to the network **ncsu**

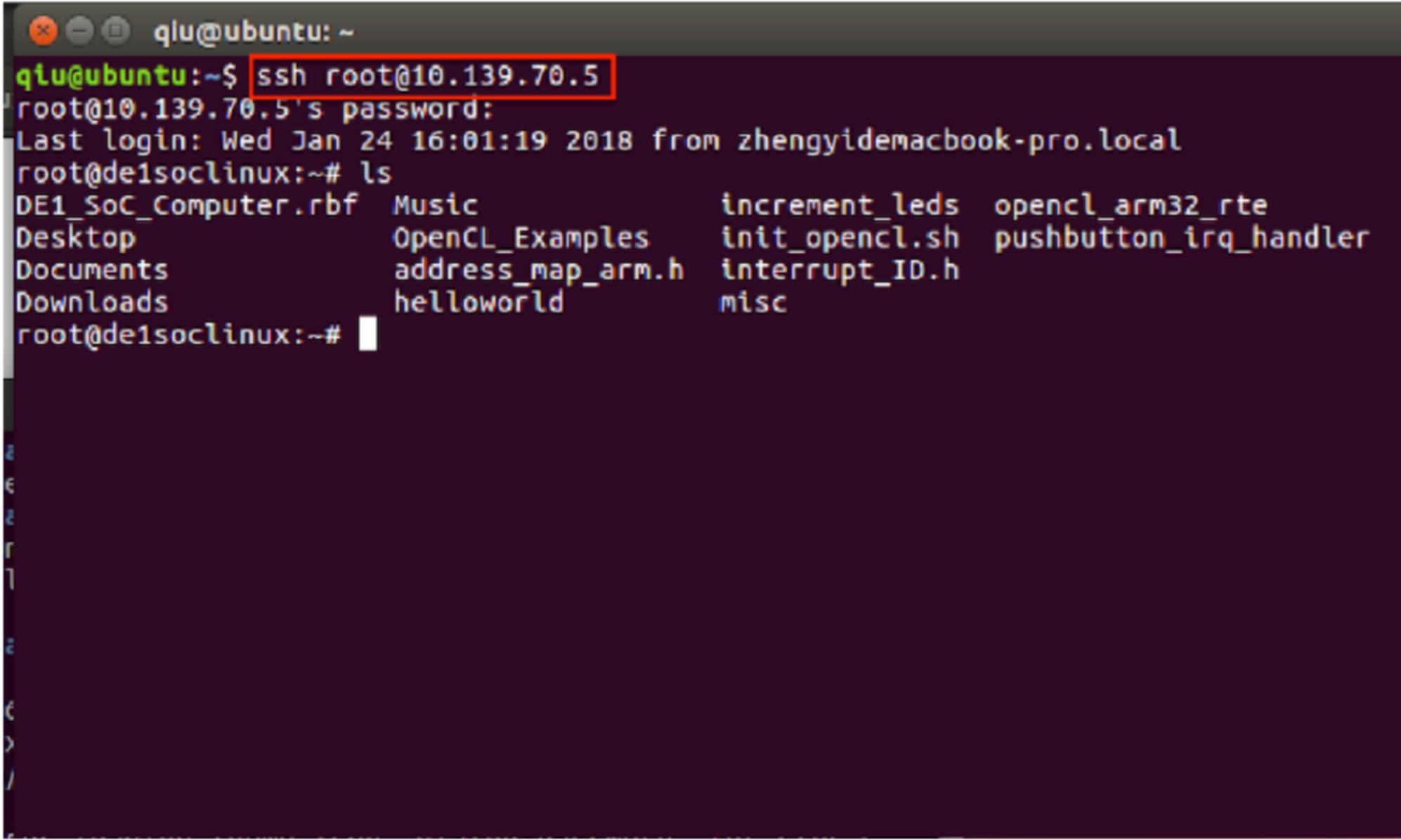
**type command**

**cd misc**

**./connect\_wap <ssid> <password> if you are at home**

**./connect\_ncsu if you are at school**

# Transfer files to DE1-SoC using SSH



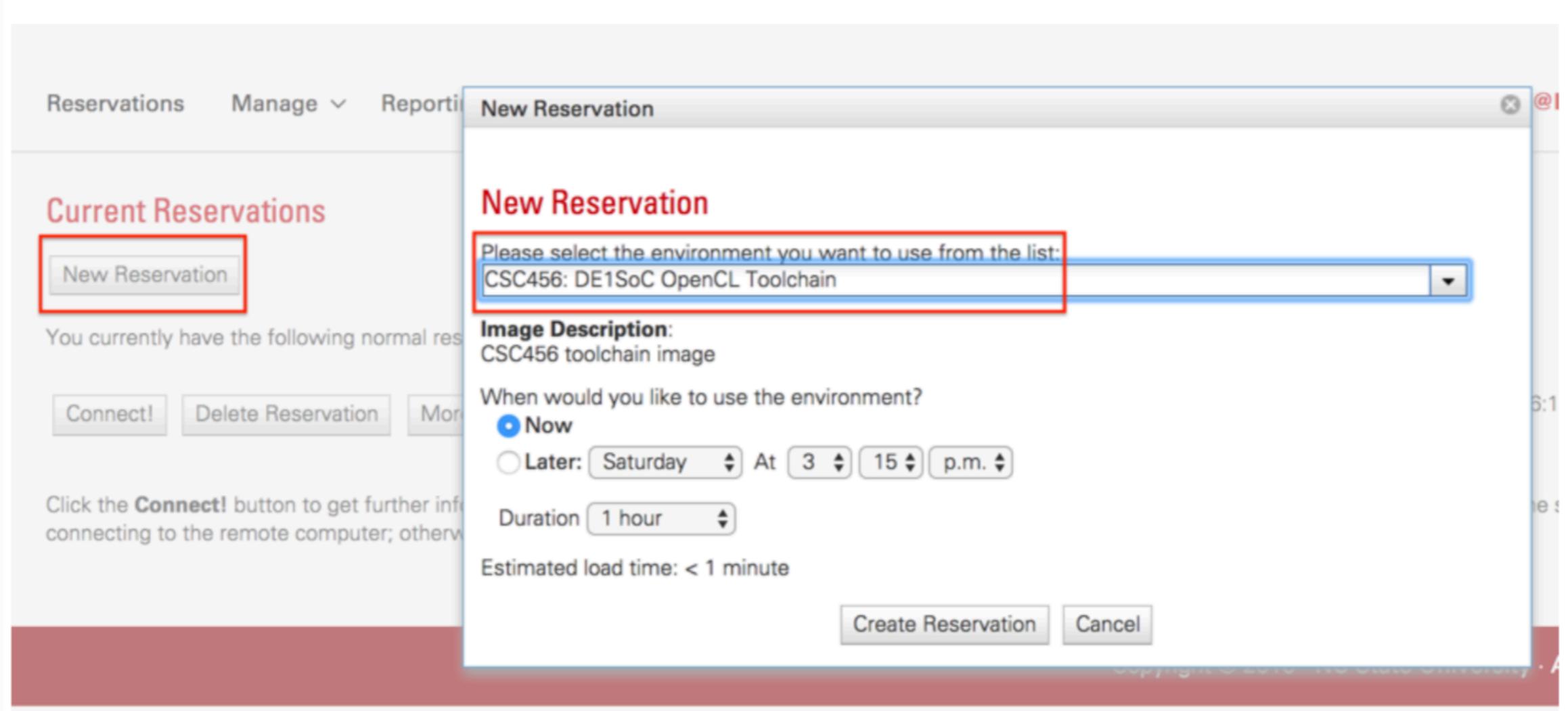
The screenshot shows a terminal window with the following session:

```
qiu@ubuntu:~$ ssh root@10.139.70.5
root@10.139.70.5's password:
Last login: Wed Jan 24 16:01:19 2018 from zhengyidemacbook-pro.local
root@de1soclinux:~# ls
DE1_SoC_Computer.rbf  Music          increment_leds  opencl_arm32_rte
Desktop                OpenCL_Examples  init_opencl.sh  pushbutton_irq_handler
Documents              address_map_arm.h  interrupt_ID.h
Downloads              helloworld      misc
root@de1soclinux:~#
```

The password to log in is: **password**

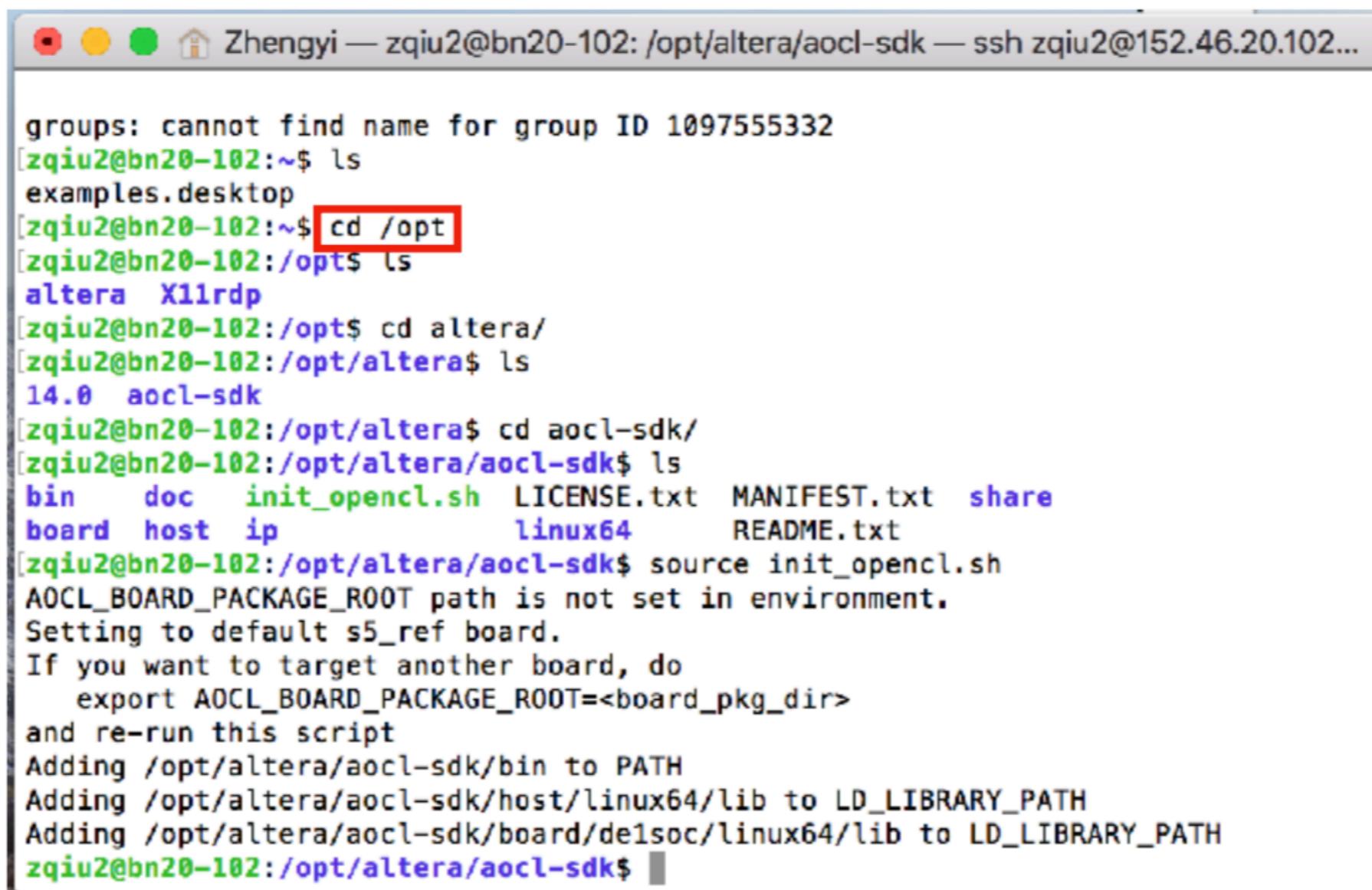
**scp /path/to/file root@host: /path/to/destination**

# Using VCL



Go to the VCL website and choose the image  
for **CSC456**

# Using VCL



The screenshot shows a terminal window titled "Zhengyi — zqiu2@bn20-102: /opt/altera/aocl-sdk — ssh zqiu2@152.46.20.102...". The terminal output is as follows:

```
groups: cannot find name for group ID 1097555332
[zqiu2@bn20-102:~$ ls
examples.desktop
[zqiu2@bn20-102:~$ cd /opt
[zqiu2@bn20-102:/opt$ ls
altera X11rdp
[zqiu2@bn20-102:/opt$ cd altera/
[zqiu2@bn20-102:/opt/altera$ ls
14.0 aocl-sdk
[zqiu2@bn20-102:/opt/altera$ cd aocl-sdk/
[zqiu2@bn20-102:/opt/altera/aocl-sdk$ ls
bin doc init_opencl.sh LICENSE.txt MANIFEST.txt share
board host ip linux64 README.txt
[zqiu2@bn20-102:/opt/altera/aocl-sdk$ source init_opencl.sh
AOCL_BOARD_PACKAGE_ROOT path is not set in environment.
Setting to default s5_ref board.
If you want to target another board, do
  export AOCL_BOARD_PACKAGE_ROOT=<board_pkg_dir>
and re-run this script
Adding /opt/altera/aocl-sdk/bin to PATH
Adding /opt/altera/aocl-sdk/host/linux64/lib to LD_LIBRARY_PATH
Adding /opt/altera/aocl-sdk/board/de1soc/linux64/lib to LD_LIBRARY_PATH
zqiu2@bn20-102:/opt/altera/aocl-sdk$ ]
```

**set up OpenCL environment variables**

**source init\_opencl.sh**

**export LM\_LICENSE\_FILE=50002@escalab.org**

**NC STATE**

# Using VCL

```
[zqiu2@bn20-102:/opt/altera$ cd 14.0/
[zqiu2@bn20-102:/opt/altera/14.0$ ls
embedded ip licenses logs nios2eds quartus uninstall
[zqiu2@bn20-102:/opt/altera/14.0$ cd embedded/
[zqiu2@bn20-102:/opt/altera/14.0/embedded$ ls
ds-5          embedded_command_shell.sh env.sh host_tools version.txt
ds-5_installer embeddedsw           examples ip
[zqiu2@bn20-102:/opt/altera/14.0/embedded$ ./embedded_command_shell.sh
-----
Altera Embedded Command Shell
Version 14.0
-----
groups: cannot find name for group ID 1097555332
zqiu2@bn20-102:/opt/altera/14.0/embedded$ ]]
```

**use the embedded shell in the EDS**

**./embedded\_command\_shell.sh**

# Using VCL

**The compilation takes about 30 mins to finish.**

**You can install screen on the vcl and run top command on one screen to prevent you from disconnecting due to being inactive.**

**sudo apt install screen**

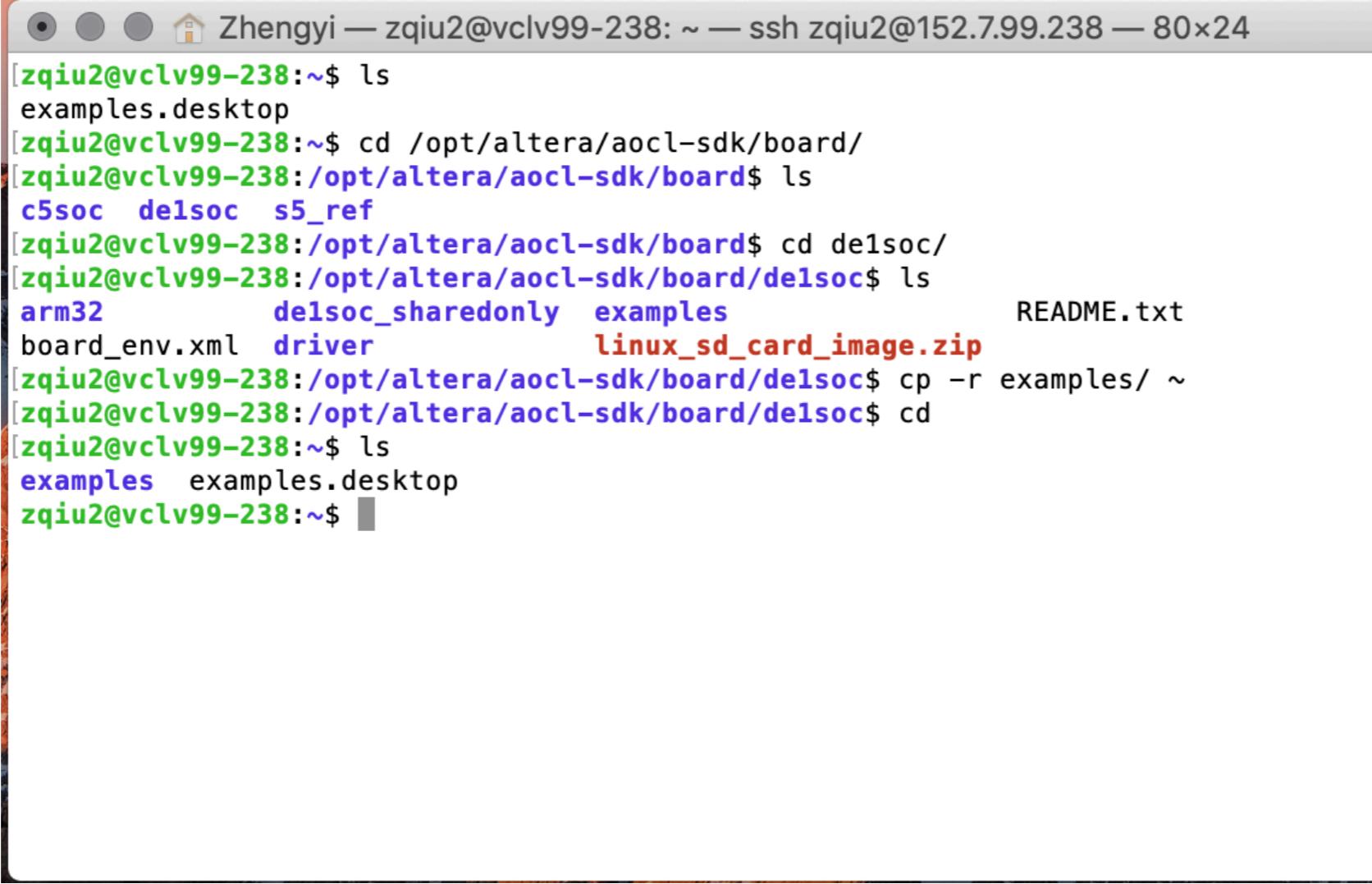
**screen**

**top**

**use Ctrl + a, Ctrl + d to detach from a screen**

**screen -r to re-attach to a screen**

# Demo



```
[zqiu2@vclv99-238:~$ ls
examples.desktop
[zqiu2@vclv99-238:~$ cd /opt/altera/aocl-sdk/board/
[zqiu2@vclv99-238:/opt/altera/aocl-sdk/board$ ls
c5soc  delsoc  s5_ref
[zqiu2@vclv99-238:/opt/altera/aocl-sdk/board$ cd de1soc/
[zqiu2@vclv99-238:/opt/altera/aocl-sdk/board/de1soc$ ls
arm32      delsoc_sharedonly  examples          README.txt
board_env.xml  driver          linux_sd_card_image.zip
[zqiu2@vclv99-238:/opt/altera/aocl-sdk/board/de1soc$ cp -r examples/ ~
[zqiu2@vclv99-238:/opt/altera/aocl-sdk/board/de1soc$ cd
[zqiu2@vclv99-238:~$ ls
examples  examples.desktop
zqiu2@vclv99-238:~$ ]
```

**copy the example directory to home directory to run examples**

**cp -r /opt/altera/aocl-sdk/board/de1soc/examples ~**

# Demo

```
[zqiu2@vclv99-238:~/examples/hello_world$ aoc device/hello_world.cl -o bin/hello_world.aocx --board de1soc_sharedonly
[zqiu2@vclv99-238:~/examples/hello_world$ make
arm-linux-gnueabihf-g++ host/src/main.cpp ../common/src/AOCL_Utils.cpp -o hello_world -I/opt/altera/aocl-sdk/host/include -I../common/inc -Ihost/inc -L/opt/altera/aocl-sdk/board/de1soc/arm32/lib -L/opt/altera/aocl-sdk/host/arm32/lib -L/opt/altera/aocl-sdk/host/arm32/lib -lalteracl -ldl -lacl_emulator_kernel_rt -lalterahalmm -lalterammdpcie -lelf -lrt -lstdc++
zqiu2@vclv99-238:~/examples/hello_world$ ]
```

go to ~/**examples** directory

**cd ~/*examples***

compile kernel code

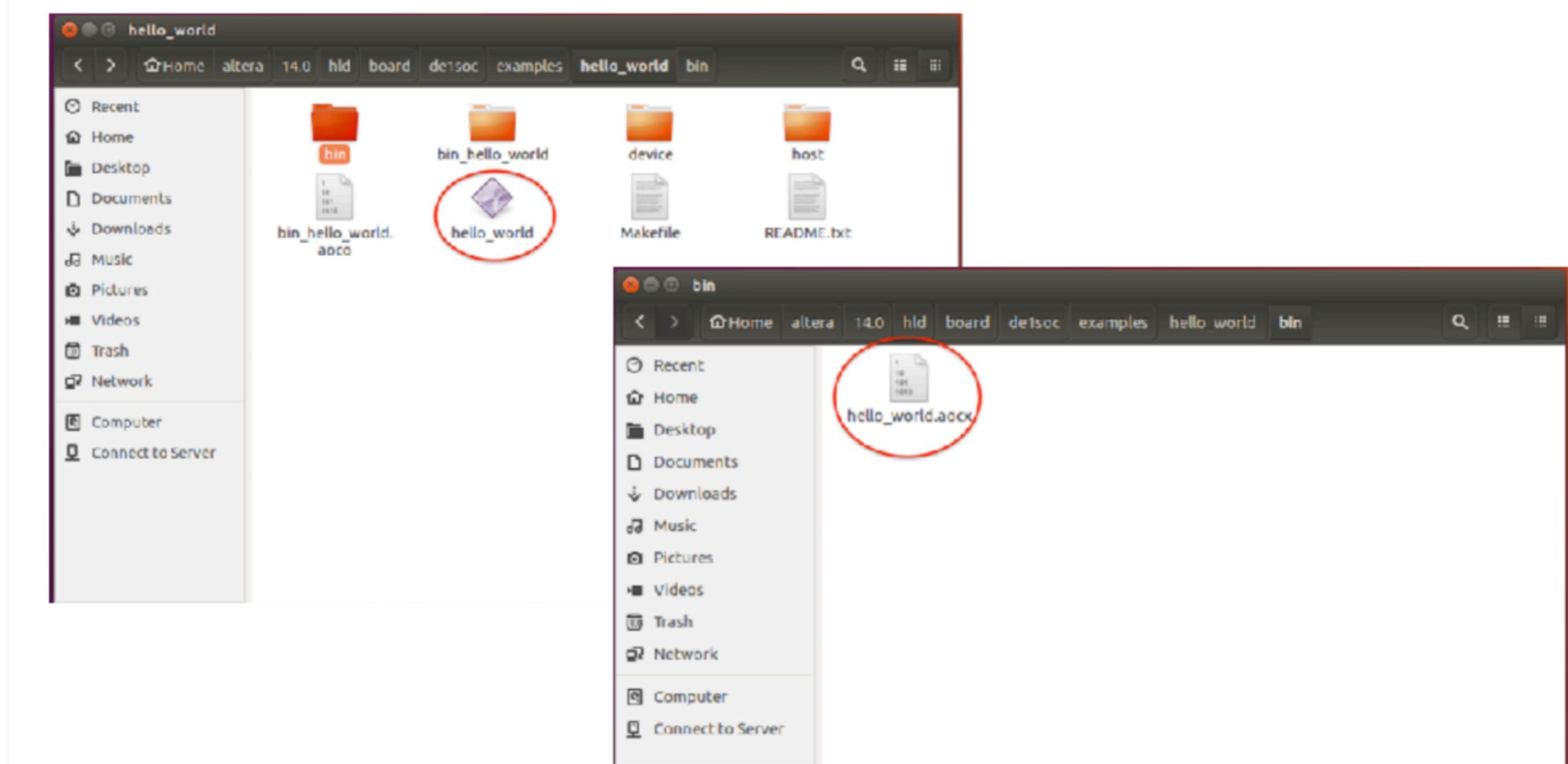
**aoc device/hello\_world.cl -o bin/hello\_world.aocx --board de1soc\_sharedonly**  
The compiling takes about 30 mins to finish, make sure you are using screen before start compiling

In the same directory

**make** to generate **hello\_world** the executable file

use **scp** to download **hello\_world** and **hello\_world.aocx** to local

# Demo



**copy hello\_world.aocx and executable hello\_world to your board:**

**using the microSD card (/media/fat\_partition) or SSH**

**(figure from my local toolchain, on VCL you will not see icons)**

# Demo - On De1-SoC

```
root@de1soclinux:/media/fat_partition# ls
hello_world  hello_world.aocx  soc_system.rbf  socfpga.dtb  uImage
root@de1soclinux:/media/fat_partition# cd
root@de1soclinux:~# mkdir test
root@de1soclinux:~# pwd
/home/root
root@de1soclinux:~# cd /media/fat_partition/
root@de1soclinux:/media/fat_partition# cp hello_world /home/root/test/
root@de1soclinux:/media/fat_partition# cp hello_world.aocx /home/root/test/
root@de1soclinux:/media/fat_partition# cd
root@de1soclinux:~# cd test/
root@de1soclinux:~/test# cd
root@de1soclinux:~# ls
DE1_SoC_Computer.rbf  Music          helloworld      misc
Desktop                OpenCL_Examples  increment_leds  opencl_arm32_rte
Documents              address_map_arm.h init_opencl.sh  pushbutton_irq_handler
Downloads              hello_world     interrupt_ID.h test
root@de1soclinux:~# source init_opencl.sh
root@de1soclinux:~/test# aocl program /dev/acl0 hello_world.aocx
aocl program: Running reprogram from /home/root/opencl_arm32_rte/board/c5soc/arm
32/bin
Reprogramming was successful!
root@de1soclinux:~/test#
```

**in root directory:**

**source init\_opencl.sh**

**in program directory:**

**aocl program /dev/acl0 hello\_world.aocx**

# Demo - On De1-SoC

```
RX bytes:359717211 (359.7 MB) TX bytes:8430218 (8.4 MB)

root@deisoclinux:/test# ls
hello_world hello_world.aocx
root@deisoclinux:/test# aocl program /dev/acl0 hello_world.aocx
aocl program: Running reprogram from /home/root/opencl_arm32_rte/board/c5soc/arm
32/bin
Reprogramming was successful!
root@deisoclinux:/test# ./hello_world
Querying platform for info:
=====
CL_PLATFORM_NAME          = Altera SIK for OpenCL
CL_PLATFORM_VENDOR         = Altera Corporation
CL_PLATFORM_VERSION        = OpenCL 1.0 Altera SDK for OpenCL, Ver
sion 14.0
Querying device for info:
=====
CL_DEVICE_NAME              = deisoc_sharedonly : Cyclone V SoC Dev
elopment Kit
CL_DEVICE_VENDOR            = Altera Corporation
CL_DEVICE_VENDOR_ID          = 4466
CL_DEVICE_VERSION           = OpenCL 1.0 Altera SDK for OpenCL, Ver
sion 14.0
CL_DRIVER_VERSION           = 14.0
CL_DEVICE_ADDRESS_BITS       = 64
CL_DEVICE_AVAILABLE          = true
CL_DEVICE_ENDIAN_LITTLE       = true
CL_DEVICE_GLOBAL_MEM_CACHE_SIZE = 32768
CL_DEVICE_GLOBAL_MEM_CACHELINE_SIZE = 0
CL_DEVICE_GLOBAL_MEM_SIZE      = 536870912
CL_DEVICE_IMAGE_SUPPORT        = false
CL_DEVICE_LOCAL_MEM_SIZE       = 16384
CL_DEVICE_MAX_CLOCK_FREQUENCY = 1000
CL_DEVICE_MAX_COMPUTE_UNITS     = 1
CL_DEVICE_MAX_CONSTANT_ARCS      = 8
CL_DEVICE_MAX_CONSTANT_BUFFER_SIZE = 134217728
CL_DEVICE_MAX_WORK_ITEM_DIMENSIONS = 3
CL_DEVICE_MAX_WORK_ITEM_DIMENSIONS = 8192
CL_DEVICE_MIN_DATA_TYPE_ALIGN_SIZE = 1024
CL_DEVICE_PREFERRED_VECTOR_WIDTH_CHAR = 4
CL_DEVICE_PREFERRED_VECTOR_WIDTH_SHORT = 2
CL_DEVICE_PREFERRED_VECTOR_WIDTH_INT = 1
CL_DEVICE_PREFERRED_VECTOR_WIDTH_LONG = 1
CL_DEVICE_PREFERRED_VECTOR_WIDTH_FLOAT = 1
CL_DEVICE_PREFERRED_VECTOR_WIDTH_DOUBLE = 0
Command queue out of order?    = false
Command queue profiling enabled? = true
Using AOCX: hello_world.aocx

Kernel initialization is complete.
Launching the kernel...

Thread #2: Hello from Altera's OpenCL Compiler!

Kernel execution is complete.
root@deisoclinux:/test#
```

Invoke the host program:  
**./hello\_world**