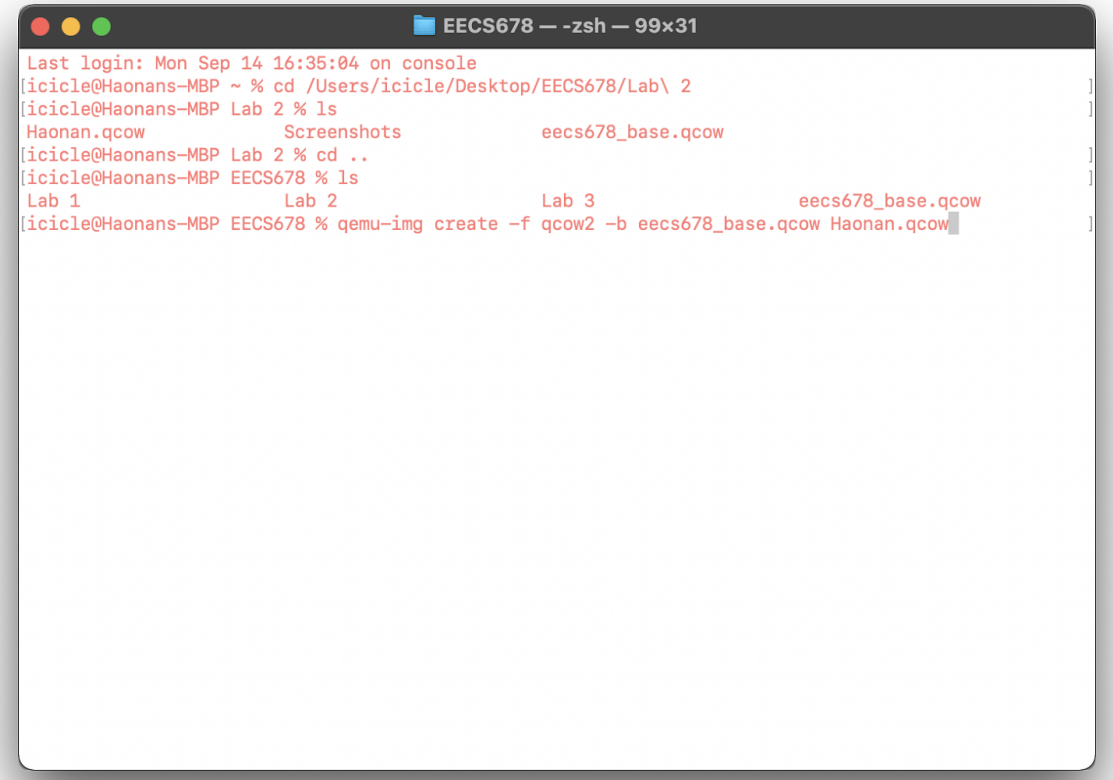
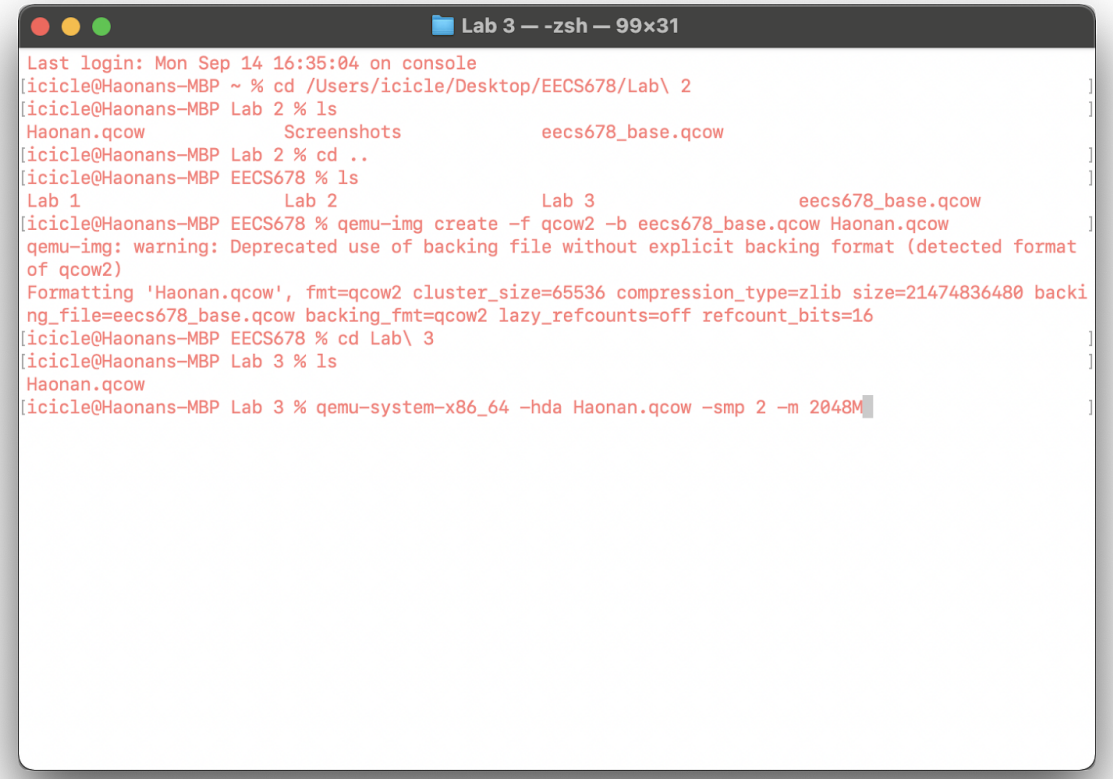
Haonan Hu

2863545

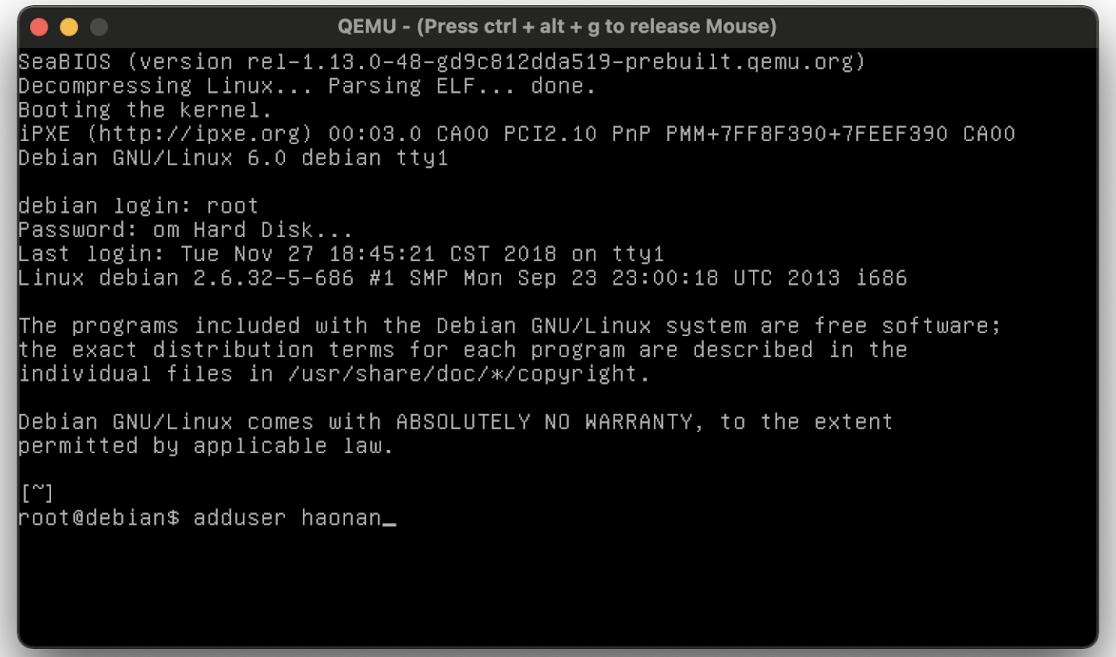
18, Sep, 2020



This will create a differential image of the base image in my first name



The command will open up a GUI for booting the image and give it 2Gigabyes of ram to use

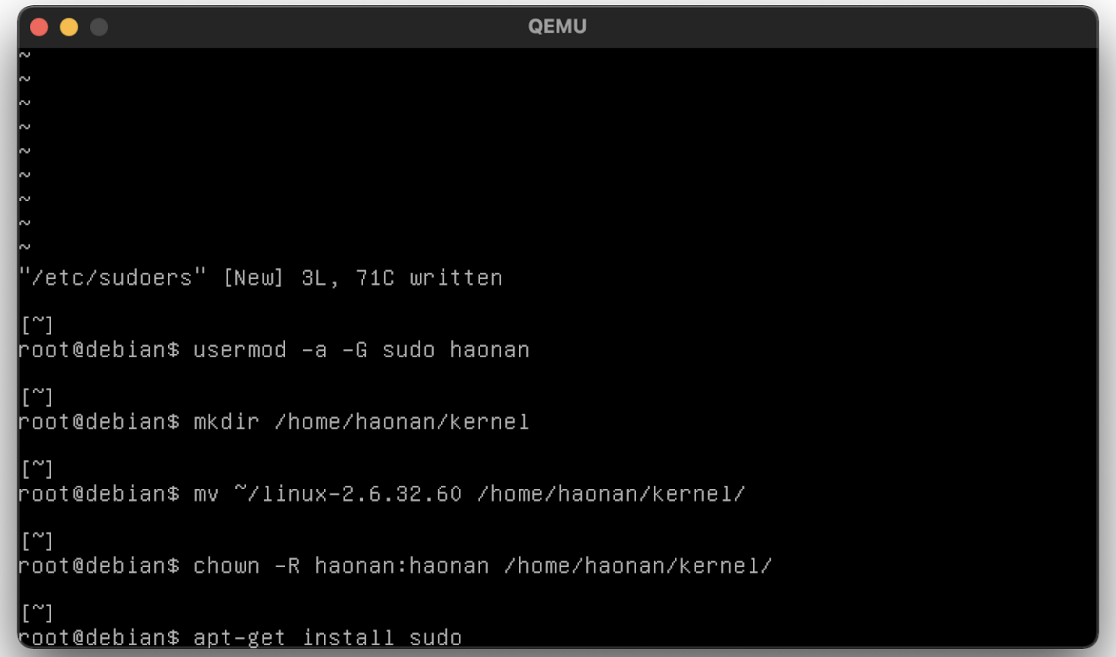


Add a user with my name from root

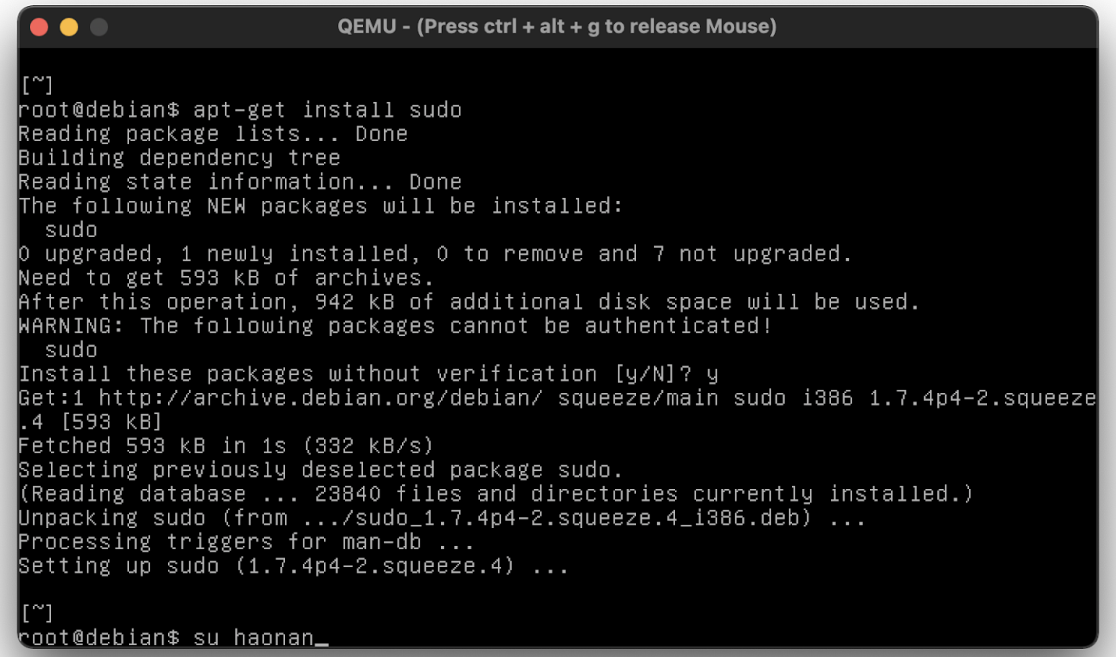
Add user to the sudoer file through vim then quit by command :wq

Add user haonan to the sudo group

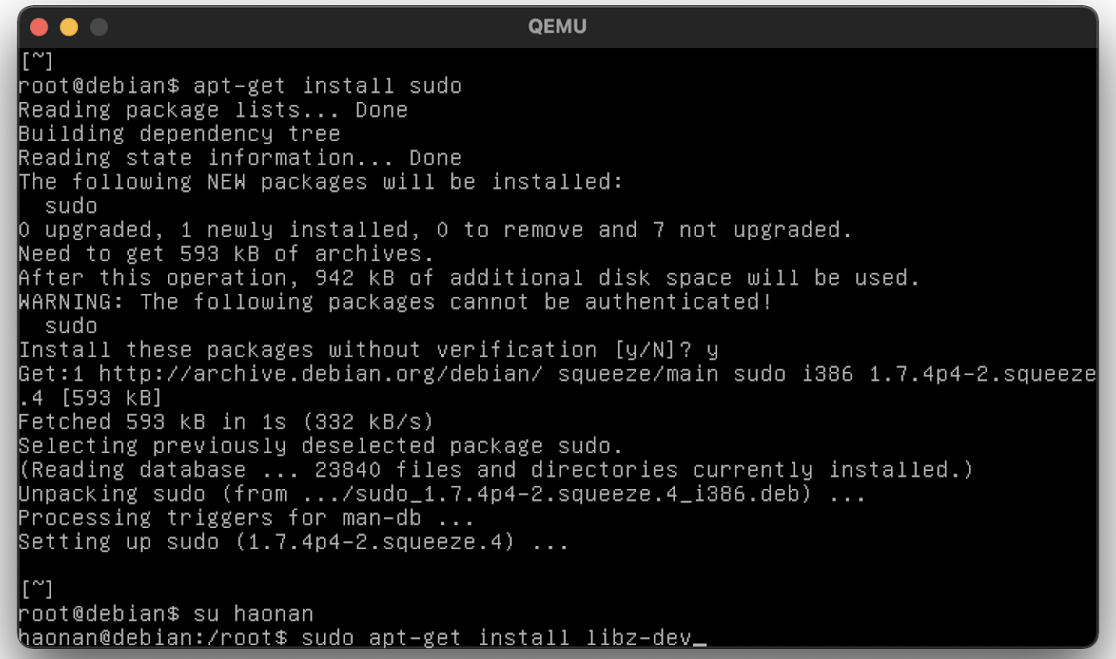
Move the kernel source and the config file to the user’s directory and change the ownership to myself



Install Sudo on kernel in orer to grant user right to execute some command

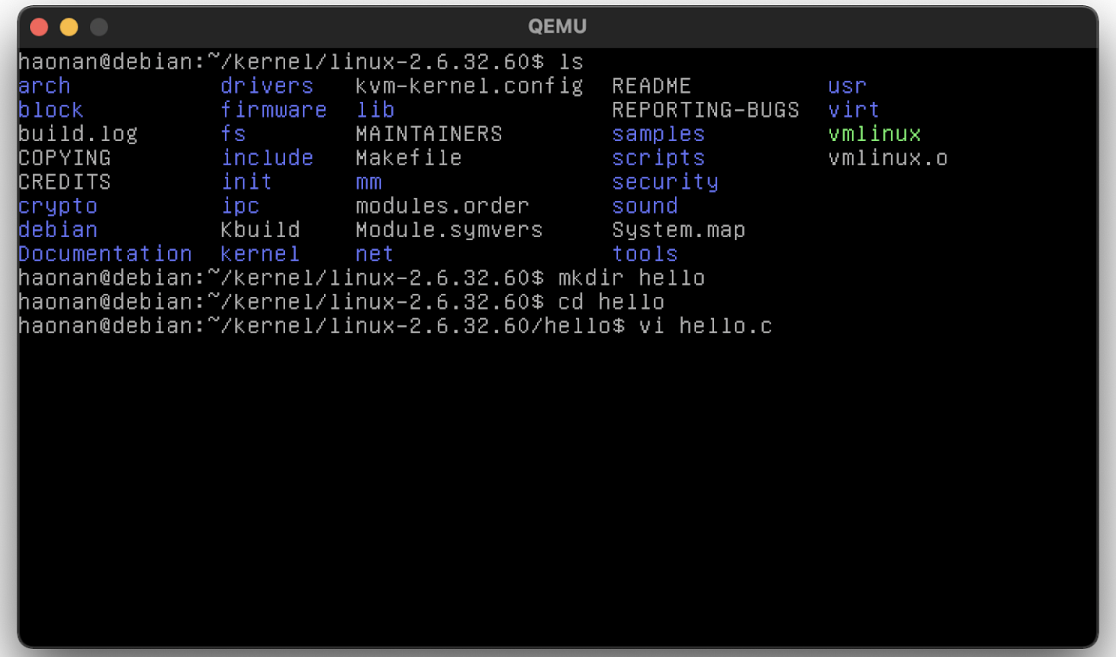


Switch user from root the new user(me)

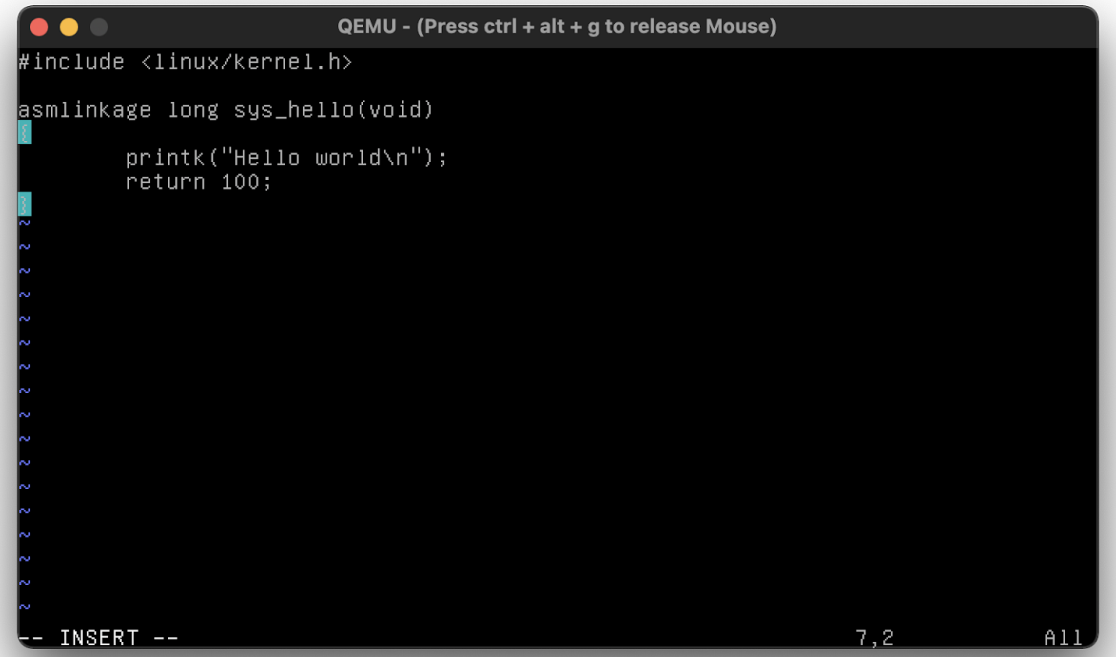


Install libz-dev by sudo command

Go to the directory where linux-2.6.32.60 located



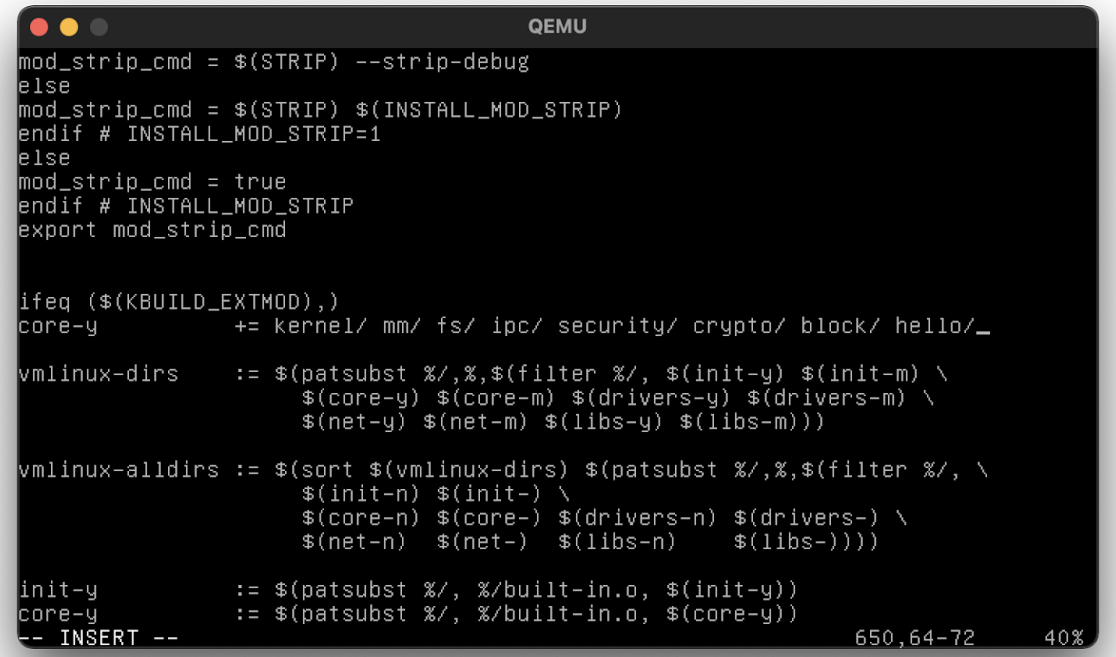
Create a directory called hello and then create a c program called hello.c under this directory.



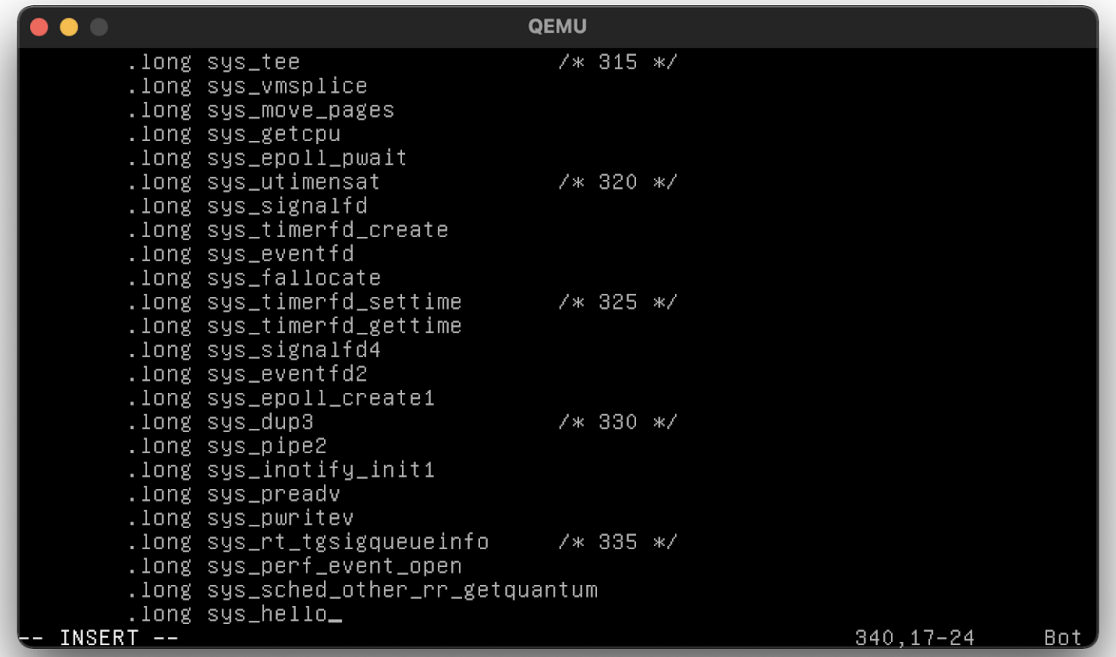
Add lines to print hello world in hello.c file

Create a makefile

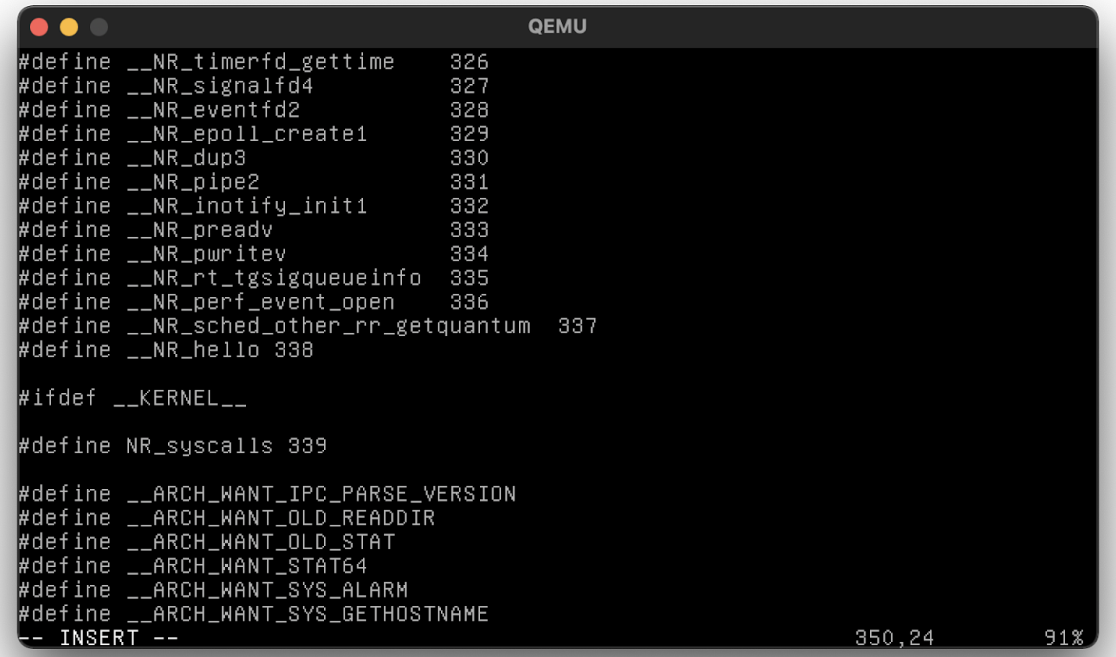
Add content to Makefile



Locate Makefile under linux-2.6.32.60 directory and open it then searching for core-y then edit it with adding hello at the end

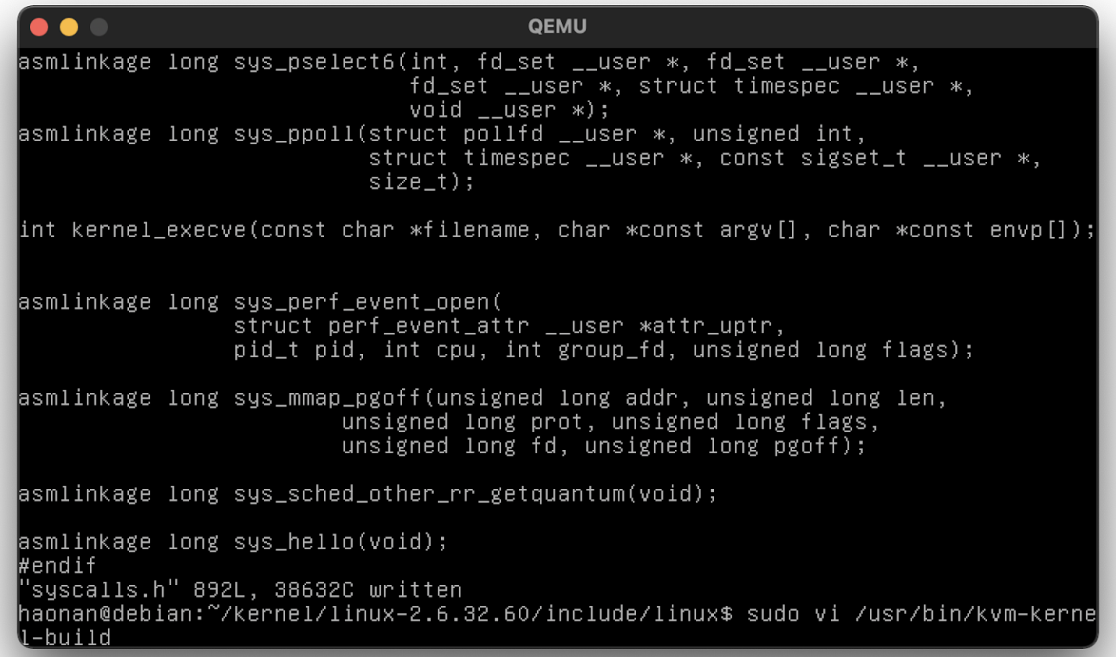


Go to directory “ arch/x86/kernel” find the file syscall_table_32.S, add .long sys_hello at the end of the file.

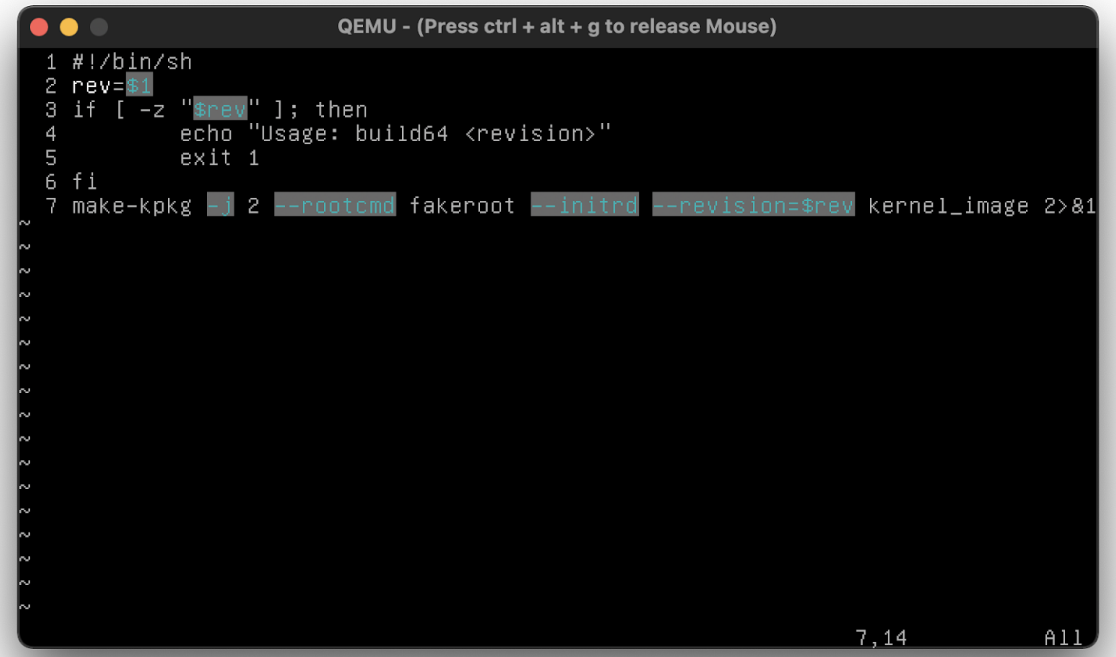


Now go to “include/asm” directory and find the file “unistd_32.h” then go to the bottom by typing in G and add #define __NR_hello 338 at the bottom of #define __NR_sched_other_rr_getquantum 337, then change #define NR_sysvalls 338 to 339.

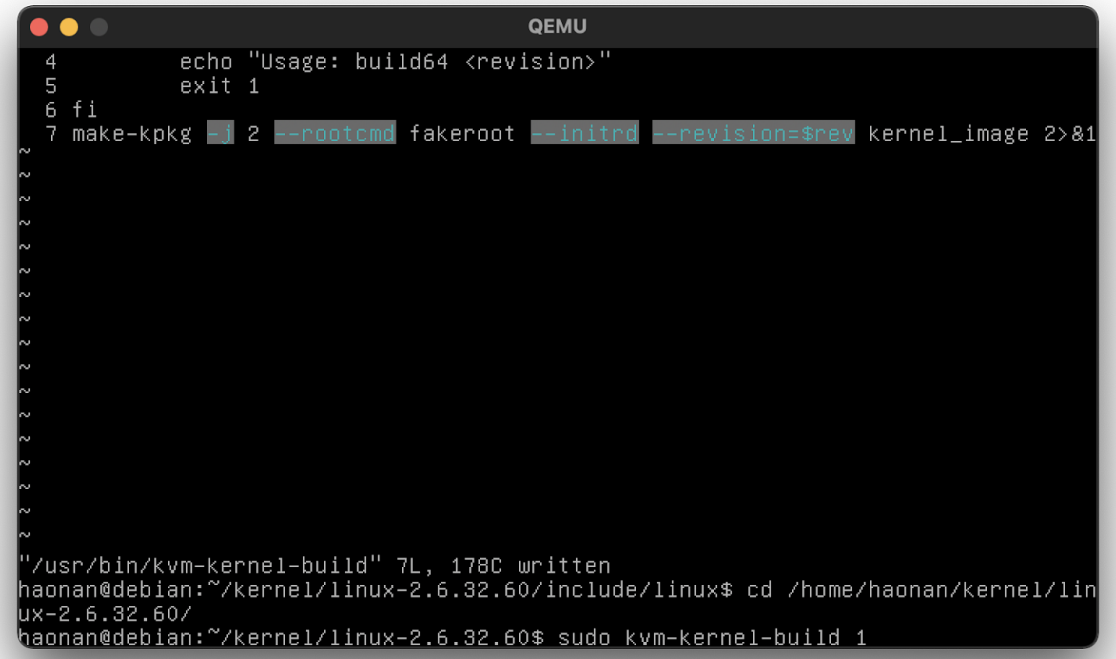
Locate “include/linux”, open file syscalls.h and add asmlinkage long sys_hello(void) the the end of the file



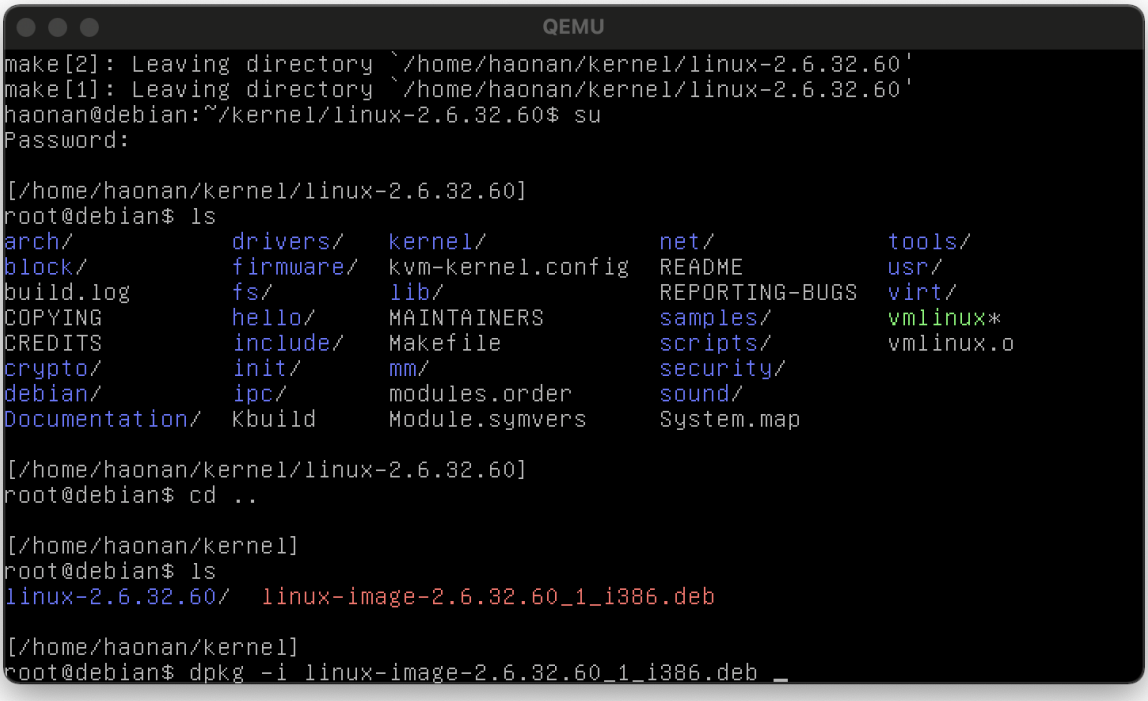
Add level of concurrency to the building process



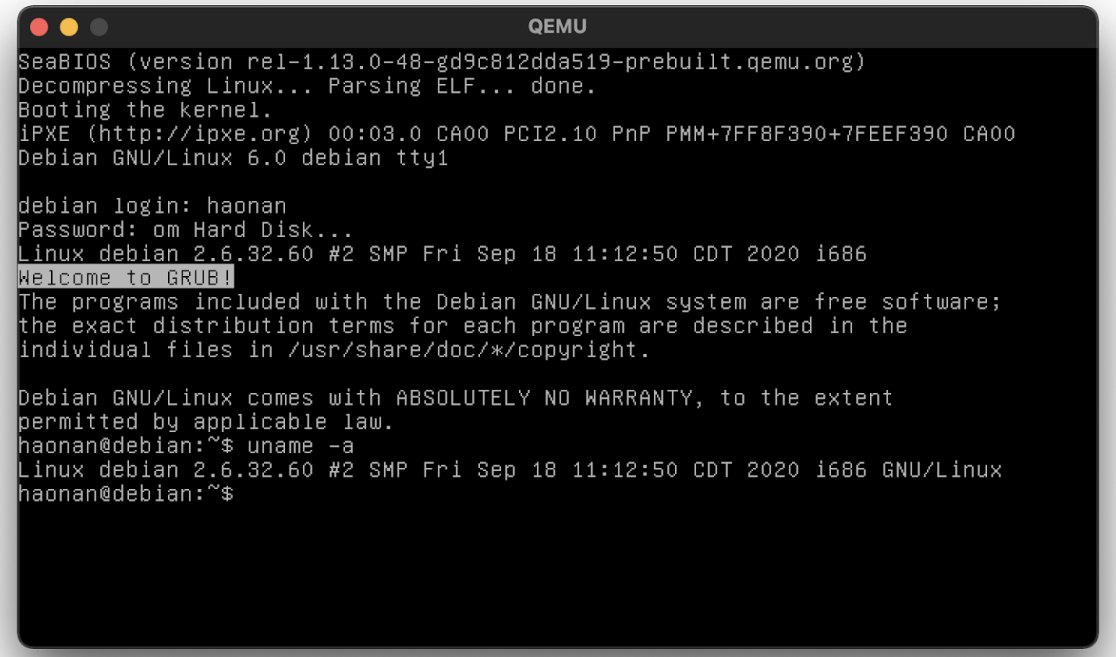
Add -j flag with value to the make-kpkg command at the end



Start build the kernel

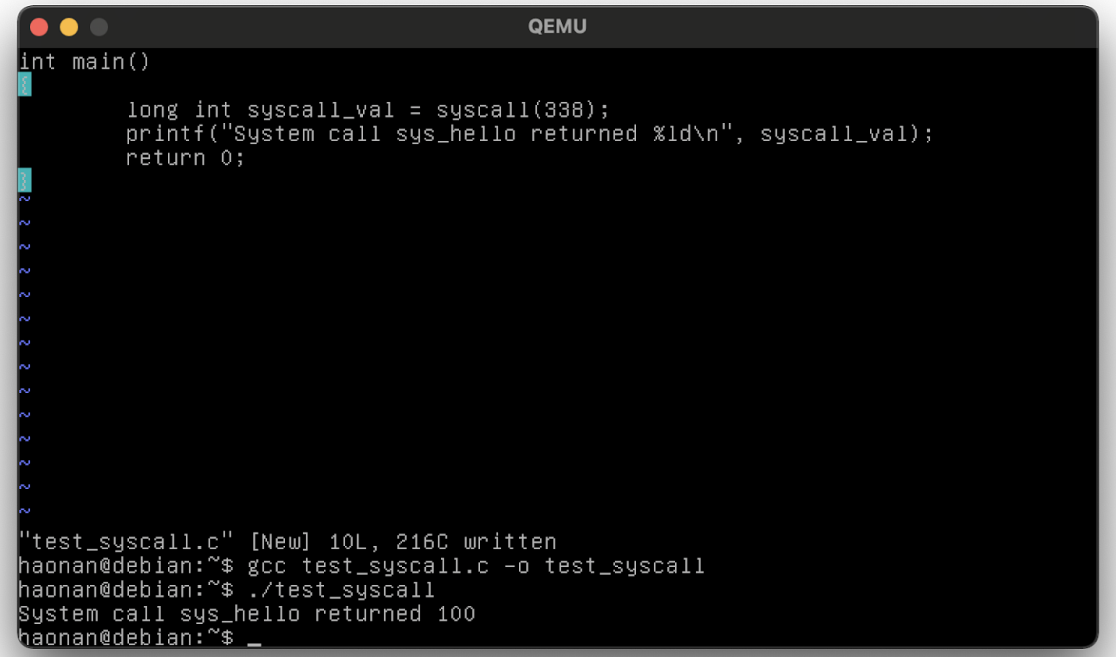


Go back to previous directory, and then start installing

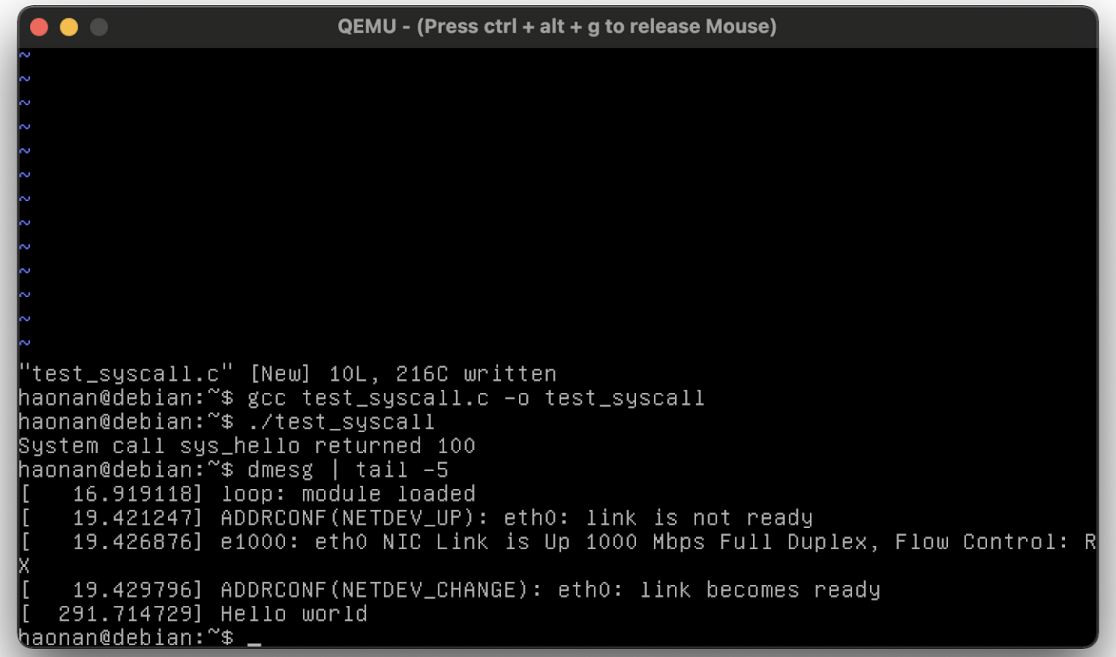


After rebooting, showed my identity and my system version

Make a test file to test syscall



Compiling test file and run it, it appears 100 which is good, means our syscall is working



In the kernel log, It prints Hello world.