Lab Submission-Week 1

ESE 3005: EMBEDDED SYSTEMS ARCHITECTURE II

Lambton College in Toronto

Instructor: Takis Zourntos

STUDENT NAME & ID:

Vishal Hasrajani(C0761544)

Parth Patel(C0764929)

Goutham Reddy Alugubelly(C0747981)

Ratnajahnavi rebbapragada(C0762196)

INTRODUCTION:

To get the Yocto working by following the Yocto mega manual and to build “core-image-sato” operating systems. Also, to display it that it runs on an emulated machine using qemuarm.
We must check the requirements for installing Yocto, the system should be having an updated software along with 50GB of space.

DESCRIPTION:

YOCTO PROJECT QUICK BUILD

In this process we are going to build a typical image using the Yocto-project. We will configure a build for a specific hardware and also to build a reference embedded operating systems called Poky.
To build the host packages we need to install essential host packages based on ubuntu distribution by running the following command.

```
$ sudo apt-get install gawk wget git-core diffstat unzip texinfo gcc-multilib \
    build-essential chrpath socat cpio python3 python3-pip python3-pexpect \
    xz-utils debianutils iputils-ping python3-git python3-jinja2 libegl1-mesa libsdl1.2-dev \
    pylint3 xterm
```

Then, we need to use Git to Clone Poky, after the setup on machine we need to get a copy of poky repository on your build host.
To clone the poky repository run the below command on the machine.

```
$ git clone git://git.yoctoproject.org/poky
```

To initialize the build, we need to follow the below steps.
Initializing the build environment

```
$cd ~/poky
```

In this poky directory, we have to run oe-init-build-env environment script for building yocto project environment on host machine.

```
$source oe-init-build-env
```

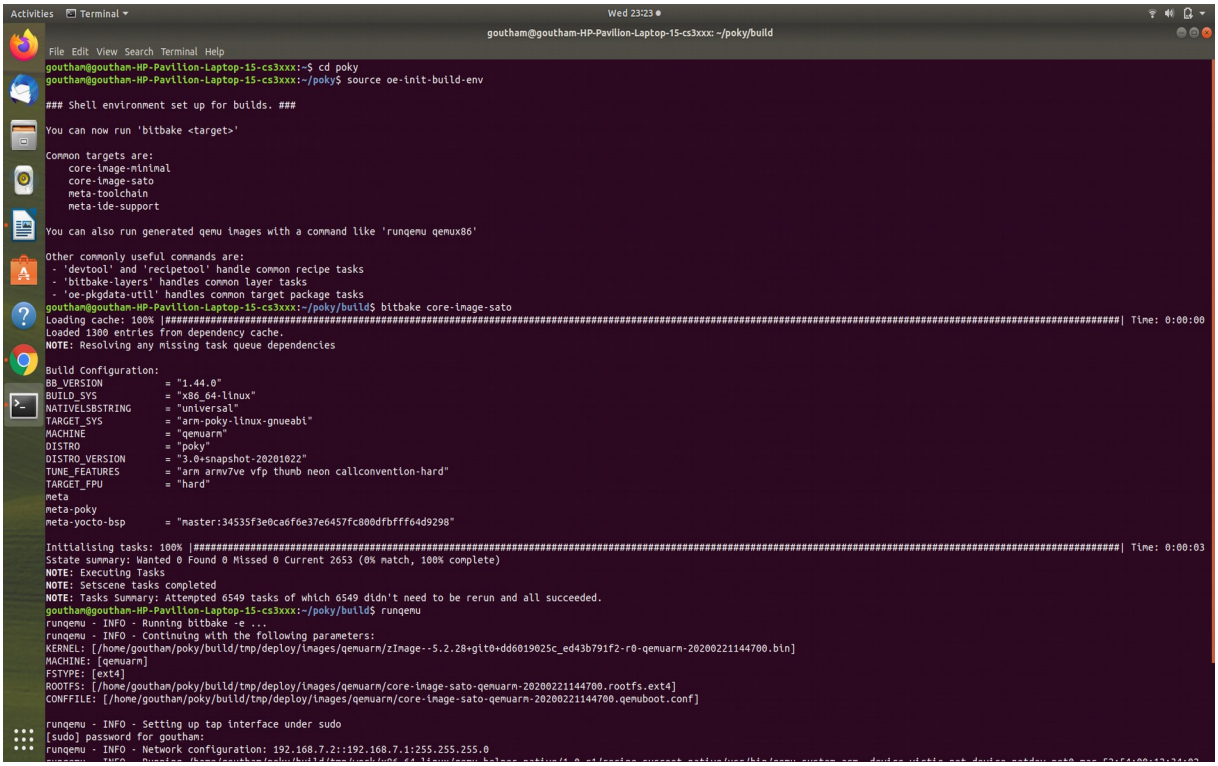
Now a local conf file will be created in conf subdirectory folder with some default values. Then we can edit it by selecting any machine.

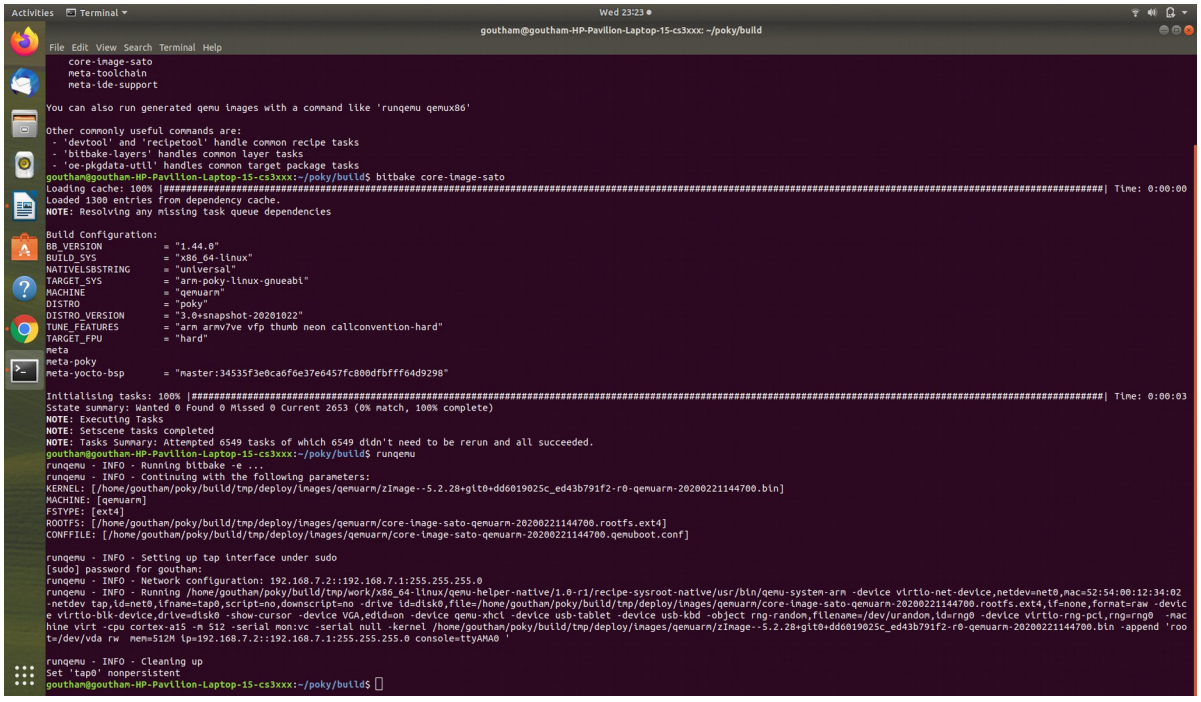
Now to build an OS image for this process, we need to run,

$bitbake core-image-sato

Now we need to simulate our image using qemu and then exit from qemu.

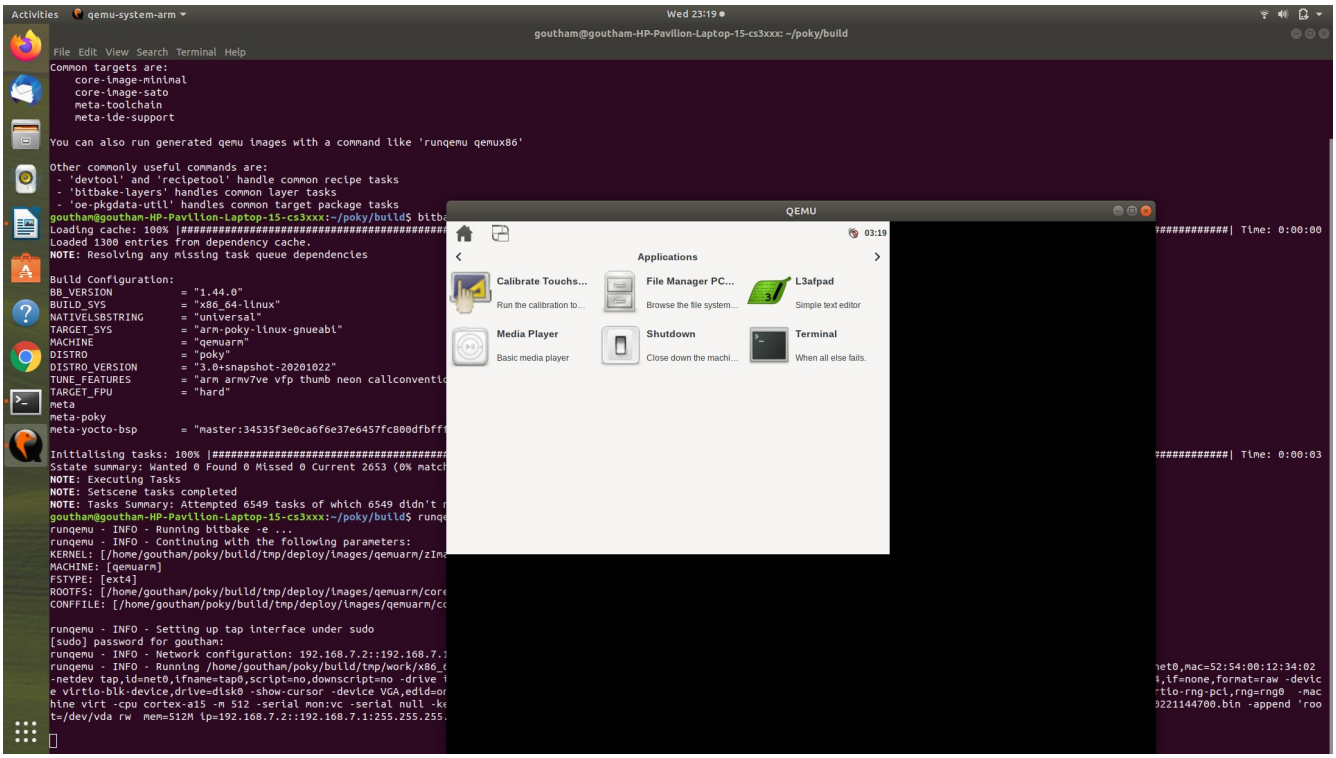
OUTPUTS:



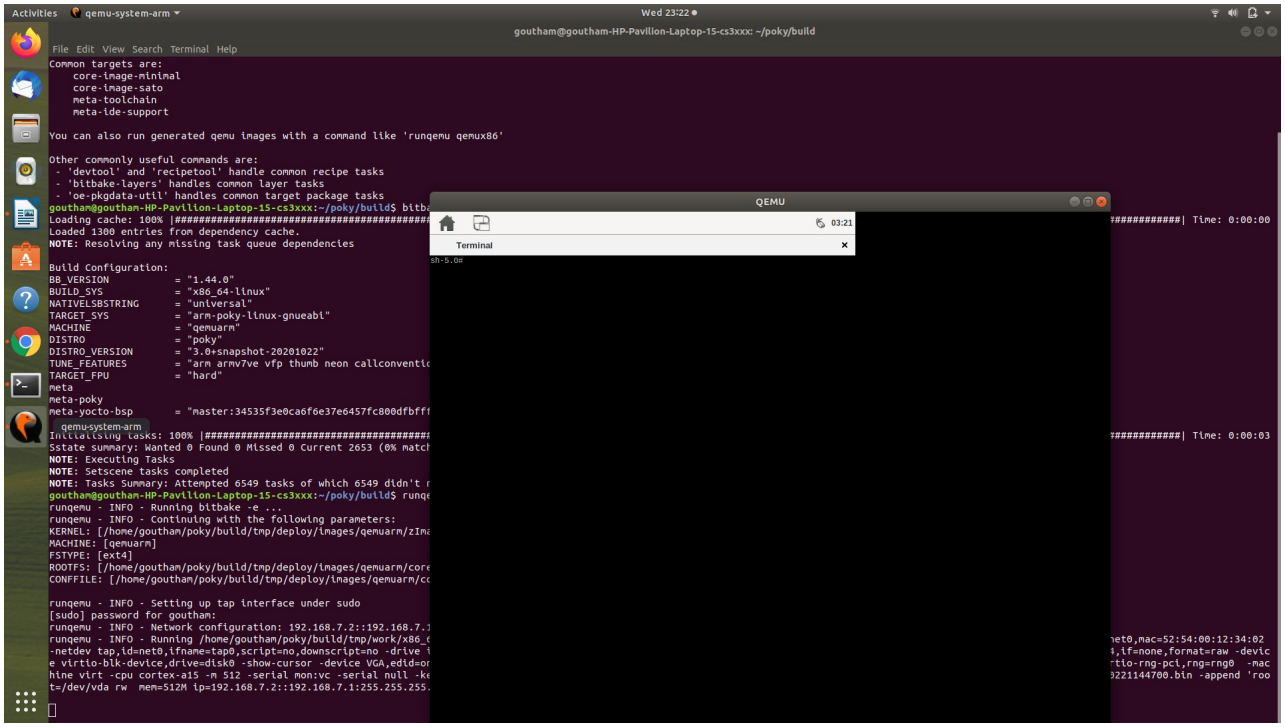


QEMU OUTPUTS:

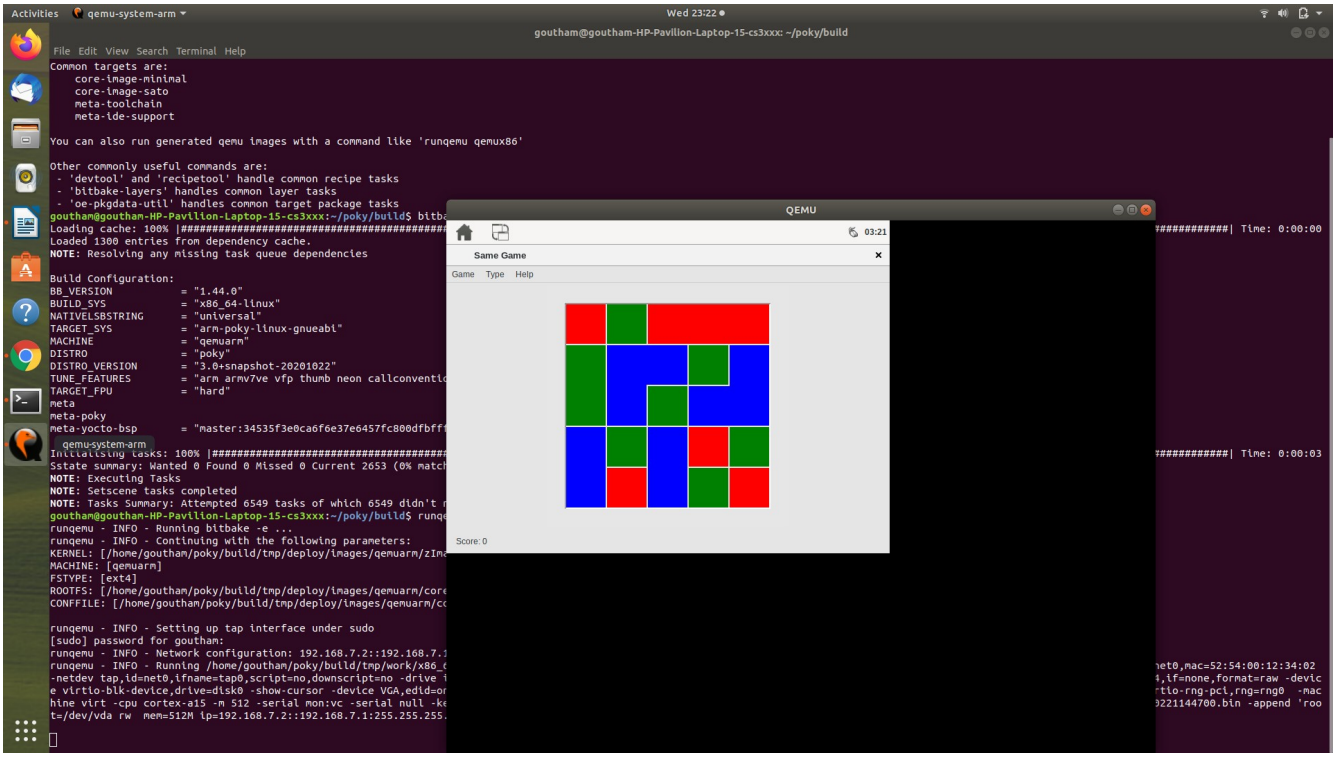
Applications in QEMU



Terminal in QEMU:



Games In QEMU:



CONCLUSION:

Building the Yocto project has been done. We have build our “core-image-sato” operating systems and after that it ran on an emulated machine using qemuarm. By doing the Yocto project we have build an reference embedded operating system called Poky. In QEMU we have different applications build, it is a qucik EMUlator along with Yocto-project.