Department of Information Technology

Academic Term: Jan-May 2021

Class : T.E IT Sem -VI
Subject : Sensor Network Lab (ITL604)

Practical No:	4
Title:	**Basic programs of cooja (Hello_world)**
Date of Performance:	
Date of Submission:	
Roll No:	8669
Name of the Student:	Mareena Mark Fernandes

Evaluation:

Sr. No	Rubric	Grade
1	**On time Completion & Submission (2)**	
2	**Output (3)**	
3	**Code Optimization (3)**	
4	**Knowledge of the topic (2)**	
5	**Total (10)**	

Signature of the Teacher :

PRACTICAL – 4

Basic programs of cooja (Hello_world)

AIM: To write and execute Hello World program in Contiki Cooja.

OBJECTIVE:

1. To build and test the program successfully.
2. To learn different types of sensors from Motes families.

DESCRIPTION:

Start Cooja

In the terminal window, go to the Cooja directory:

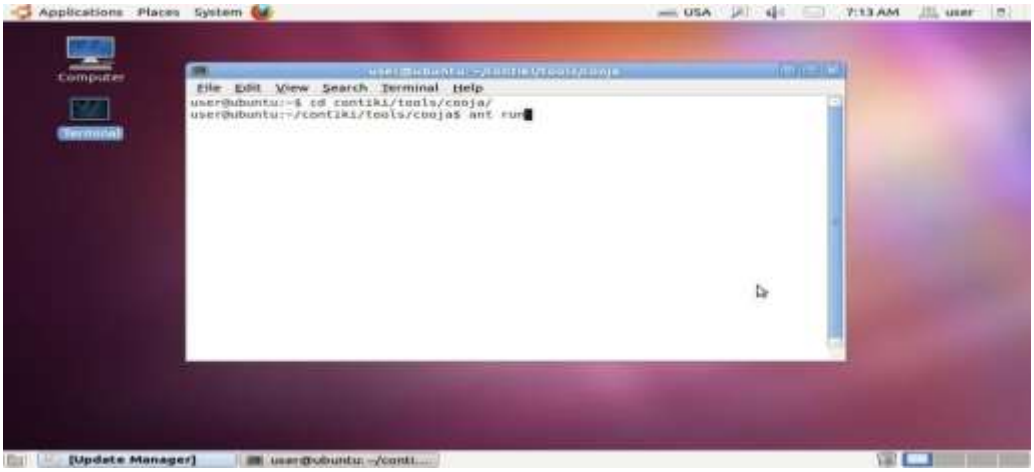
cd contiki/tools/cooja

Start Cooja with the command:

ant run

Wait for Cooja to start

When Cooja first starts, it will first compile itself, which may take some time.



When Cooja is compiled, it will start with a blue empty window.

Now that Cooja is up and running, we can try it out with an example simulation.

Step 3: Run Contiki in simulation

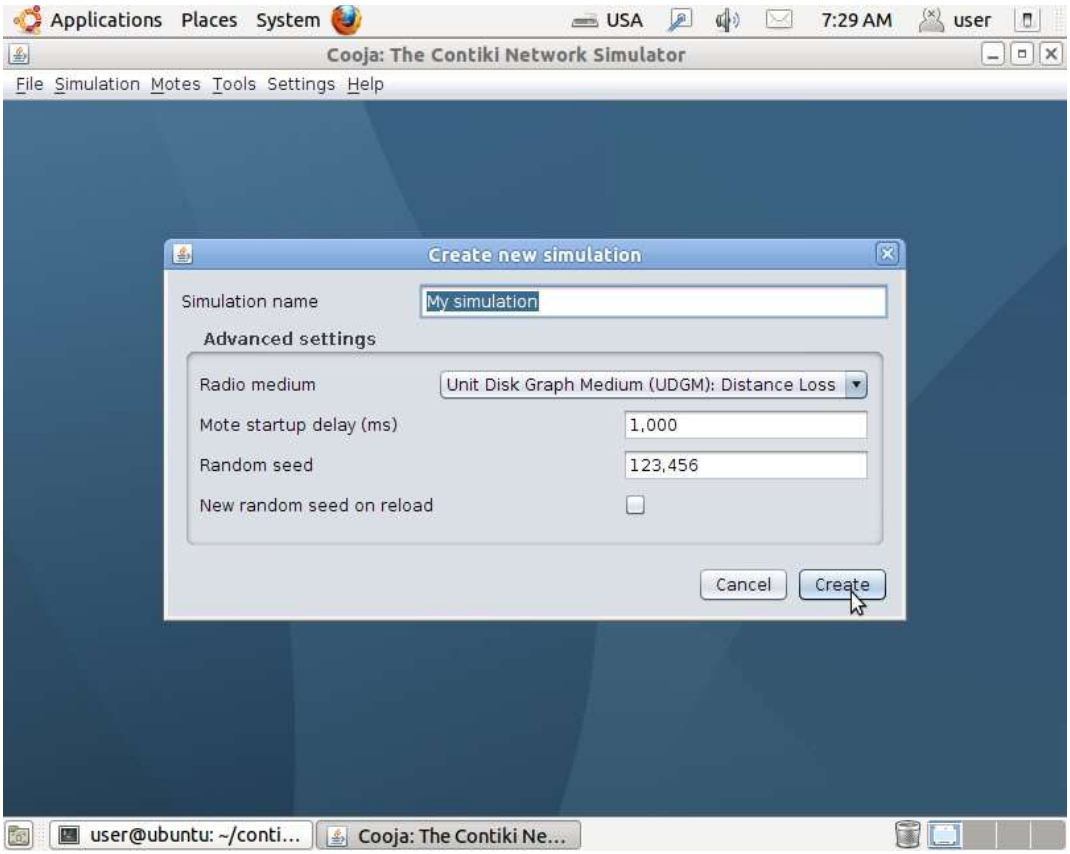
Create a new simulation

Create new simulation

Click the File menu and click New simulation....

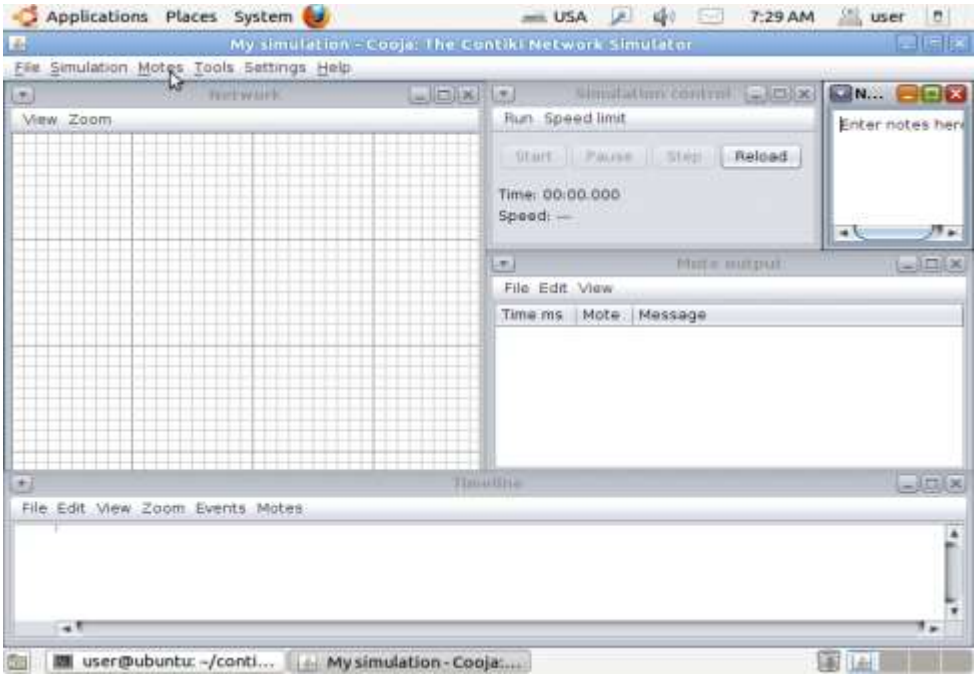
Set simulation options

Cooja now opens up the Create new simulation dialog. In this dialog, we may choose to give our simulation a new name, but for this example, we'll just stick with My simulation. Click the Create button.



Simulation windows

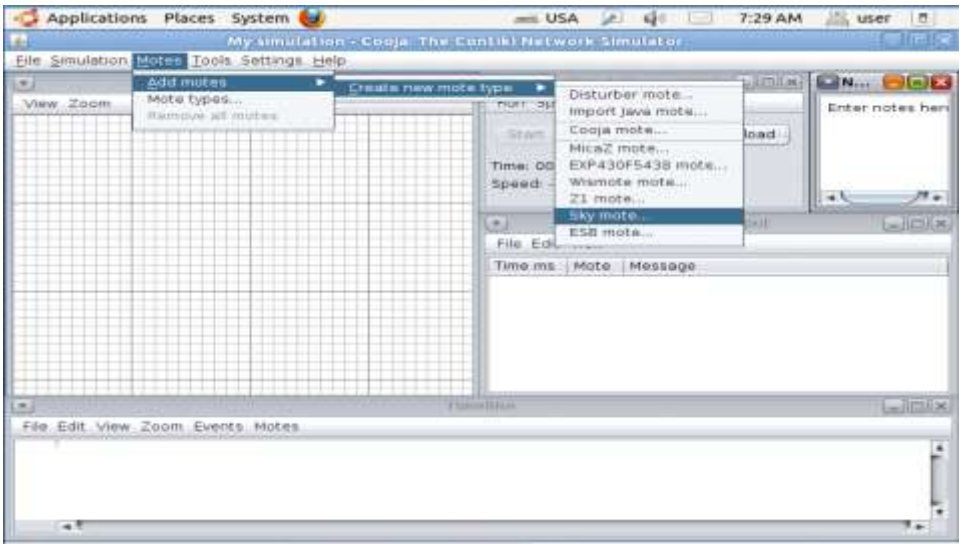
Cooja brings up the new simulation. The Network window, at the top left of the screen, shows all the motes in the simulated network - it is empty now, since we have no motes in our simulation. The Timeline window, at the bottom of the screen, shows all communication events in the simulation over time - very handy for understanding what goes on in the network. The Mote output window, on the right side of the screen, shows all serial port printouts from all the motes. The Notes window on the top right is where we can put notes for our simulation. And the Simulation control window is where we start, pause, and reload our simulation.



Add motes to the simulation

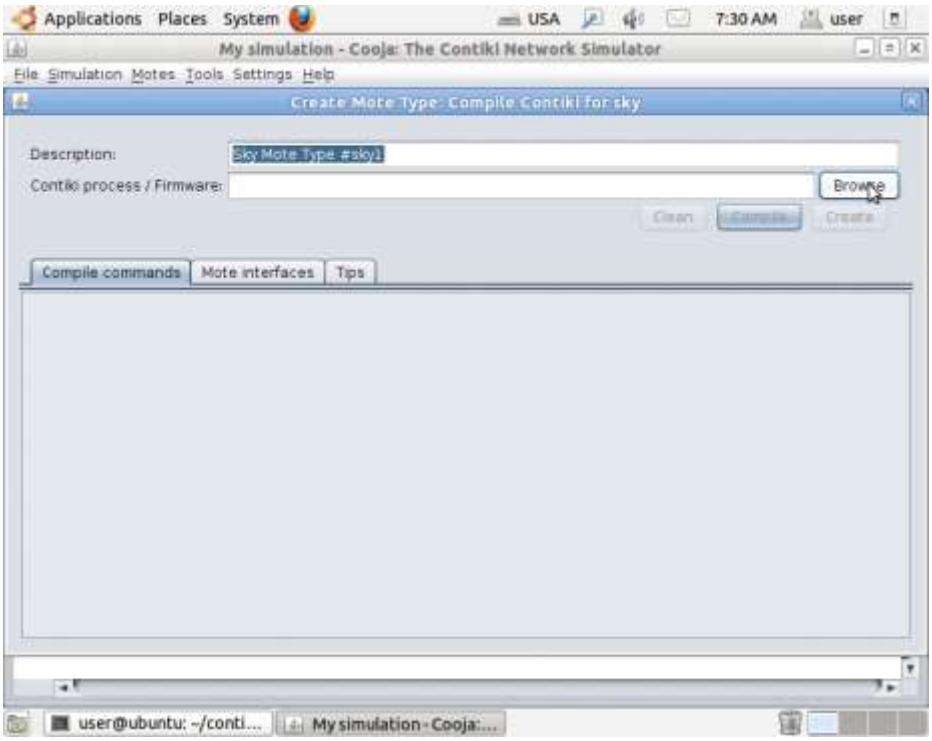
Add motes

Before we can simulate our network, we must add one or more motes. We do this via the Motes menu, where we click on Add motes. ...Since this is the first mote we add, we must first create a mote type to add. Click Create new mote type ...and select one of the available mote types. For this example, we click Sky mote... to create an emulated Tmote Sky mote type.



Create a new mote type

Cooja opens the Create Mote Type dialog, in which we can choose a name for our mote type as well as the Contiki application that our mote type will run. For this example, we stick with the suggested name, and instead click on the Browse button on the right hand side to choose our Contiki application.



Find example in Contiki application

We go to the directory /home/user/examples/hello-world/

The below program is the simplest Contiki program you can write which contains most of the necessary components that should be included in any Contiki program.

```
1:  #include “contiki.h”
2:  #include <stdio.h>
3:  PROCESS (my_first_process, “Hello World Process”);
4:  AUTOSTART_PROCESSES(&my_first_process);
5: PROCESS_THREAD (my_first_process, ev, data)
6: {
7:  PROCESS_BEGIN ();
8:  printf(“Hello World!\n”);
9:  PROCESS_END();
10: }
```

Save this file as hello_world.c

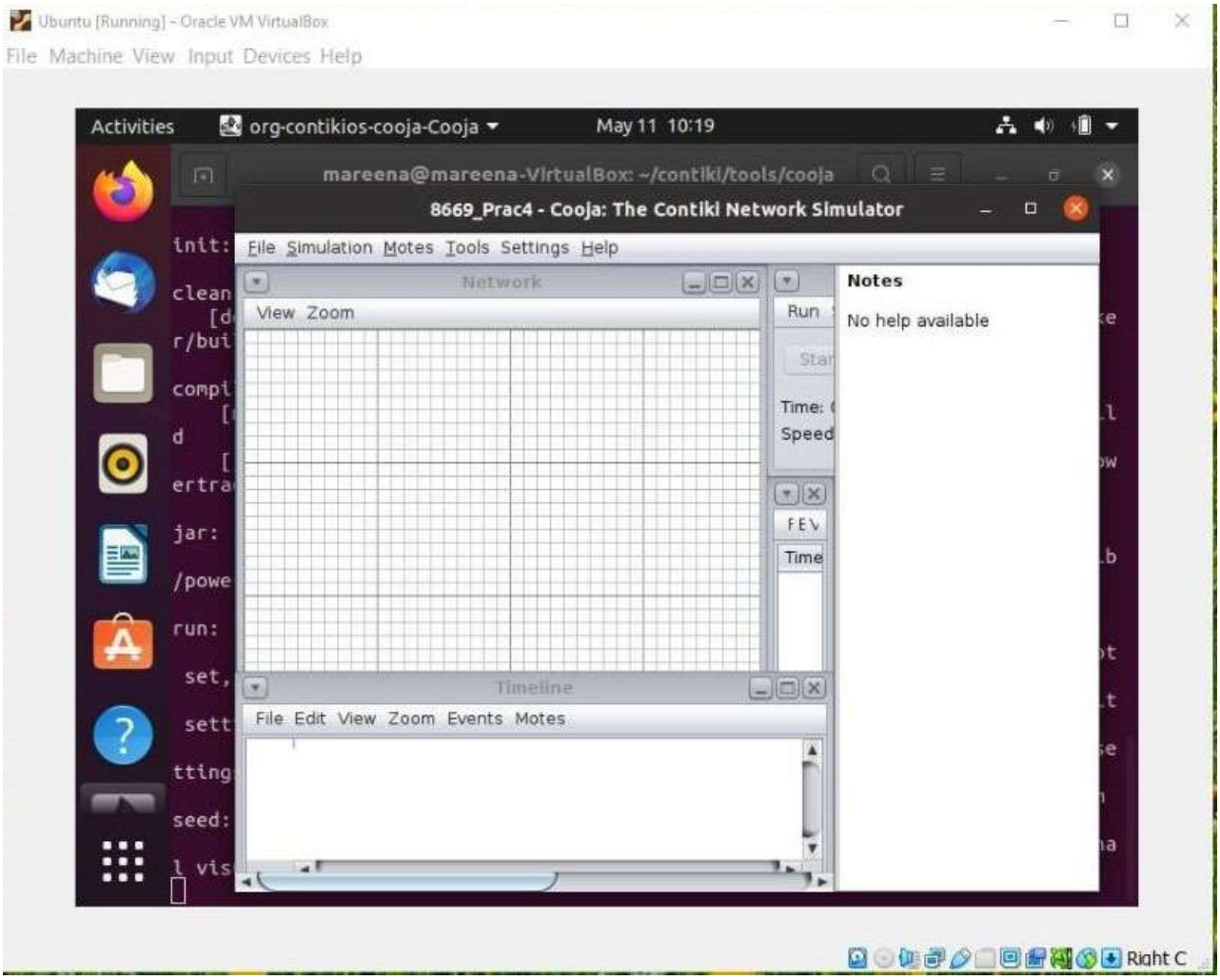
The header file “contiki.h” which is included at the very first contains all the declarations of the Contiki operating systems abstractions. stdio.h is included only because I have used printf() function inside the program. The Macro in third line declares a new Contiki process. First parameter of it is the variable for the process and the second parameter is a string name for the process. Fourth line, specify that this process should be started in the startup of the Contiki operating system. Therefore, when the hardware device which will be the destination of compiled code switches on, our process also begins running.

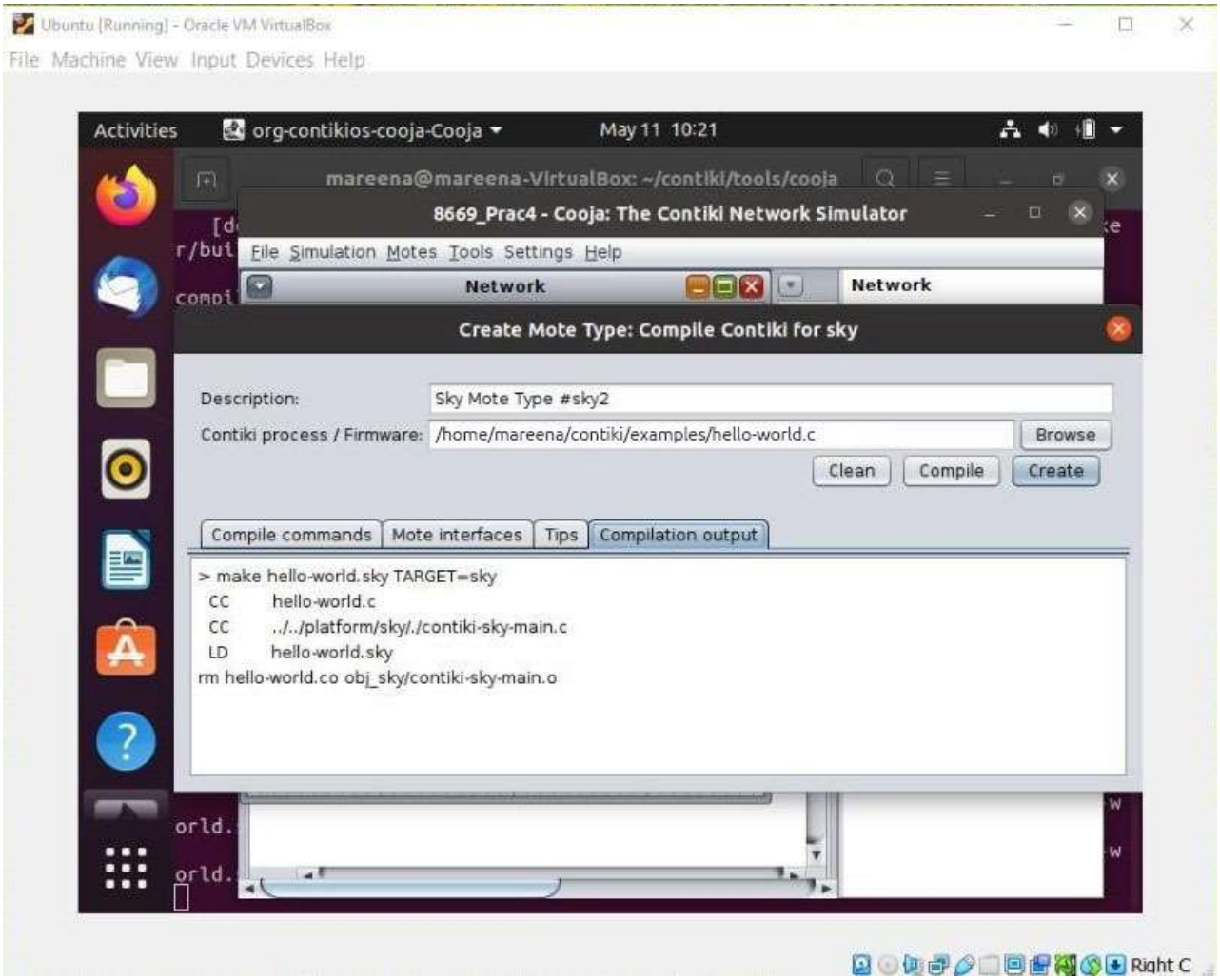
Fifth line of the code opens the definition of our process. As the first parameter, we pass the process’ variable which is my_first_process in this scenario. The second parameter is ‘ev’ which is the event parameter. It can be used to make the program responding to events that occur in the

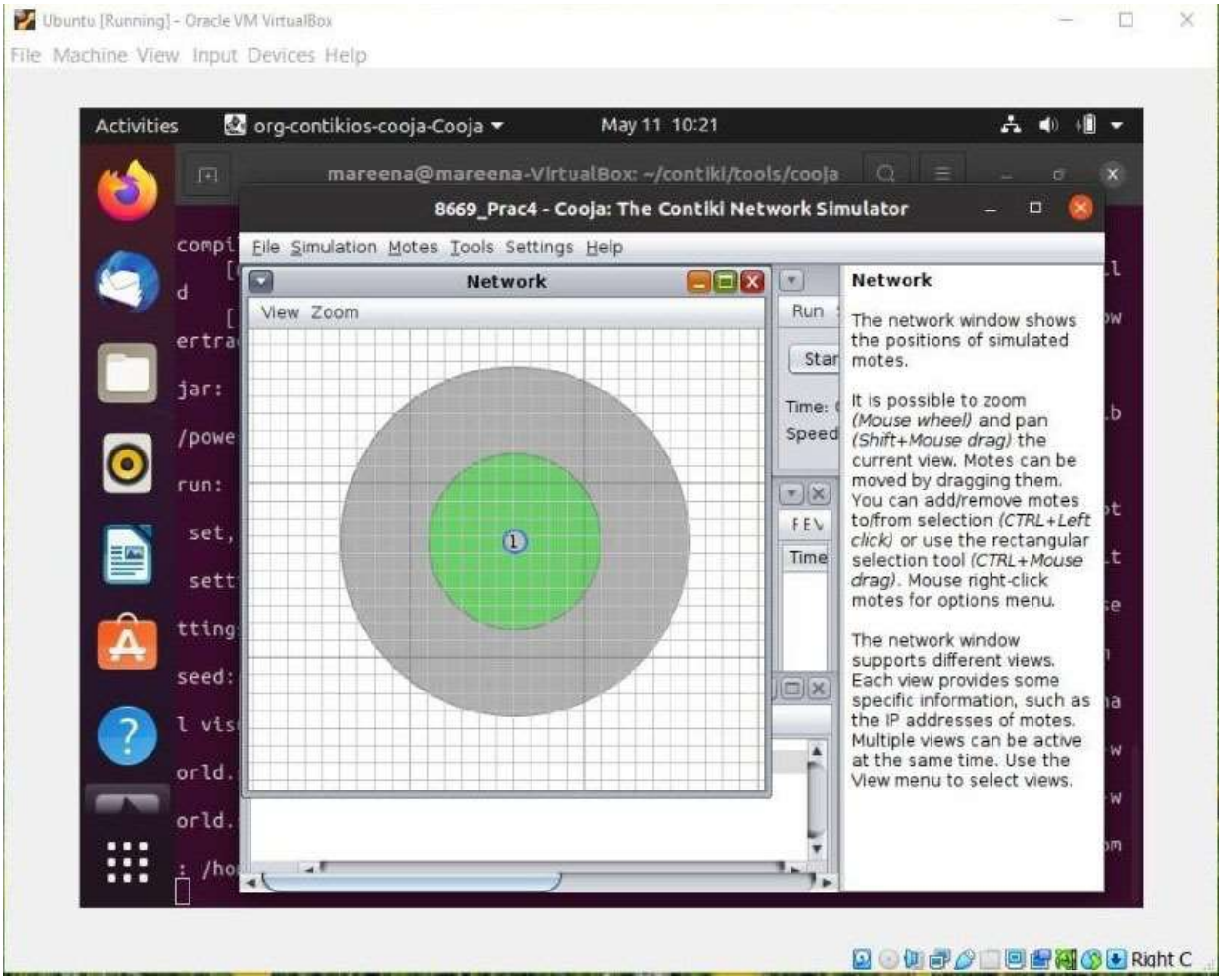
system. The third parameter ‘data’ can be used to receive some data which are coming with those events.
The rest of the instructions of the Contiki process will be included inside two important lines which are PROCESS_BEGIN() and PROCESS_END(). In our Contiki process we just print a string. This output goes to the standard output in the platform where our code will be running. If you compile this code to native platform and run in the terminal, the output will be printed in the terminal. If you run this in a mote, the output most probably will go to the UART port of the mote. However, this behavior depends on the design of a particular WSN mote.

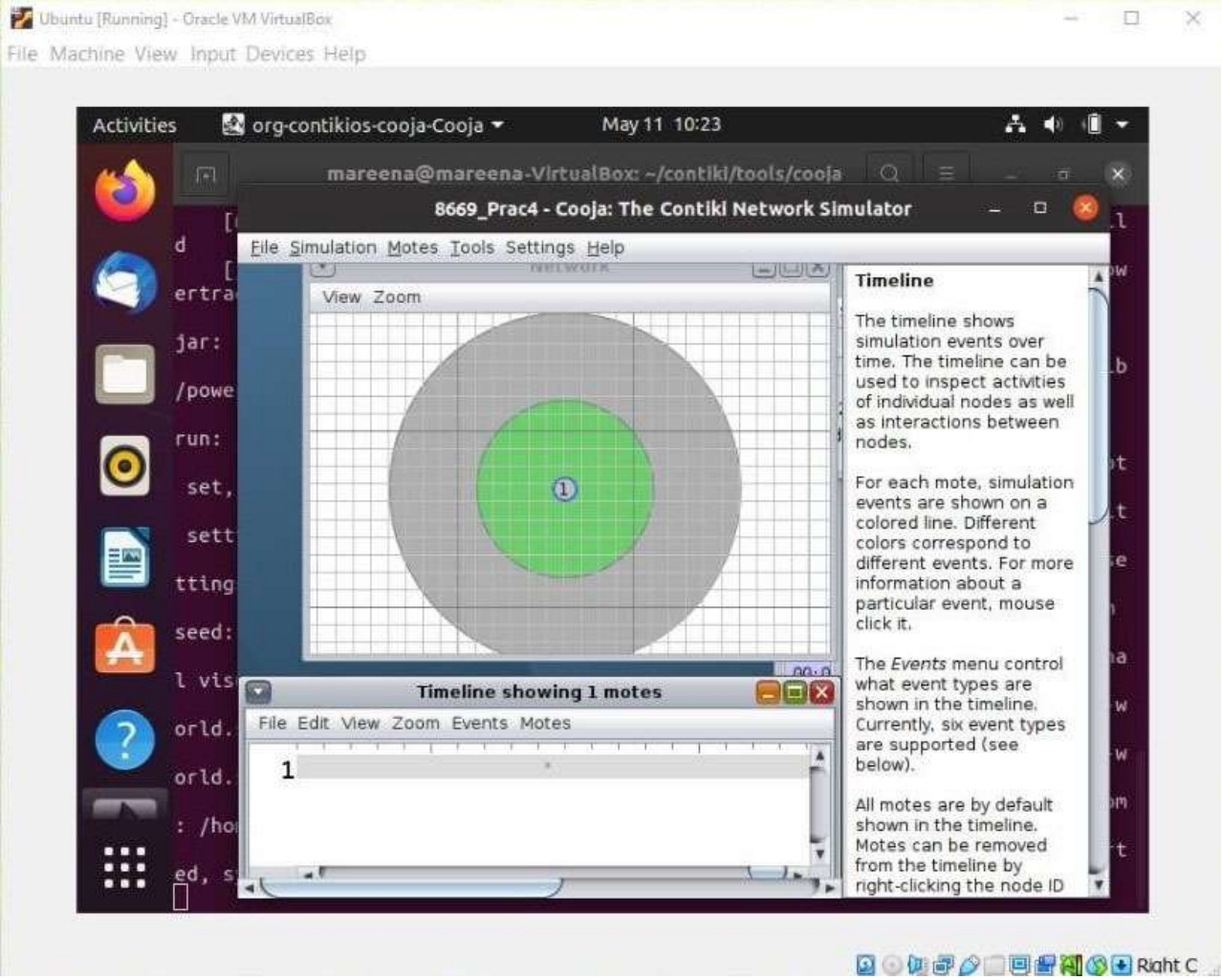
Step 1: Open the terminal window.
Step 2: In the terminal window, go to the hello world example folder.
cd /examples/hello-world
Step 3: Compile the code for the native platform (to be used when no mote is connected to the laptop.)
make TARGET=native
If there is any bug during compilation which says "#include<curses.h>: No such file or directory" you need to do the following on Ubuntu.
If you are currently in the terminal, type "sudo apt-get install libncurses5-dev"(for which you should have installed the "apt" for 32-bit using "sudo dpkg -i apt_0.8.16~exp12ubuntu10.10_i386.deb" and for 64-bit using sudo dpkg -i apt_0.8.16~exp12ubuntu10.10_amd64.deb")
You need to also install a version of javac compiler using the command "sudo apt-get install openjdk-7-jdk".
Step 4: Once the compilation is over, run the Hello World program.
./hello-world.native
Step 5: You should be seeing the following on the terminal:
Contiki 3.0 started with IPV6, RPL
Rime started with address 1.2.3.4.5.6.7.8
MAC nullmac RDC nullrdc NETWORK sicslowpan
Tentative link-local IPv6 address fe80:0000:0000:0000:0302:0304:0506:0708
Hello, world
The code will appear to hang, however, it is still running on Contiki, but not producing any output as the Hello World program is finished. Press Ctrl-C to quit.

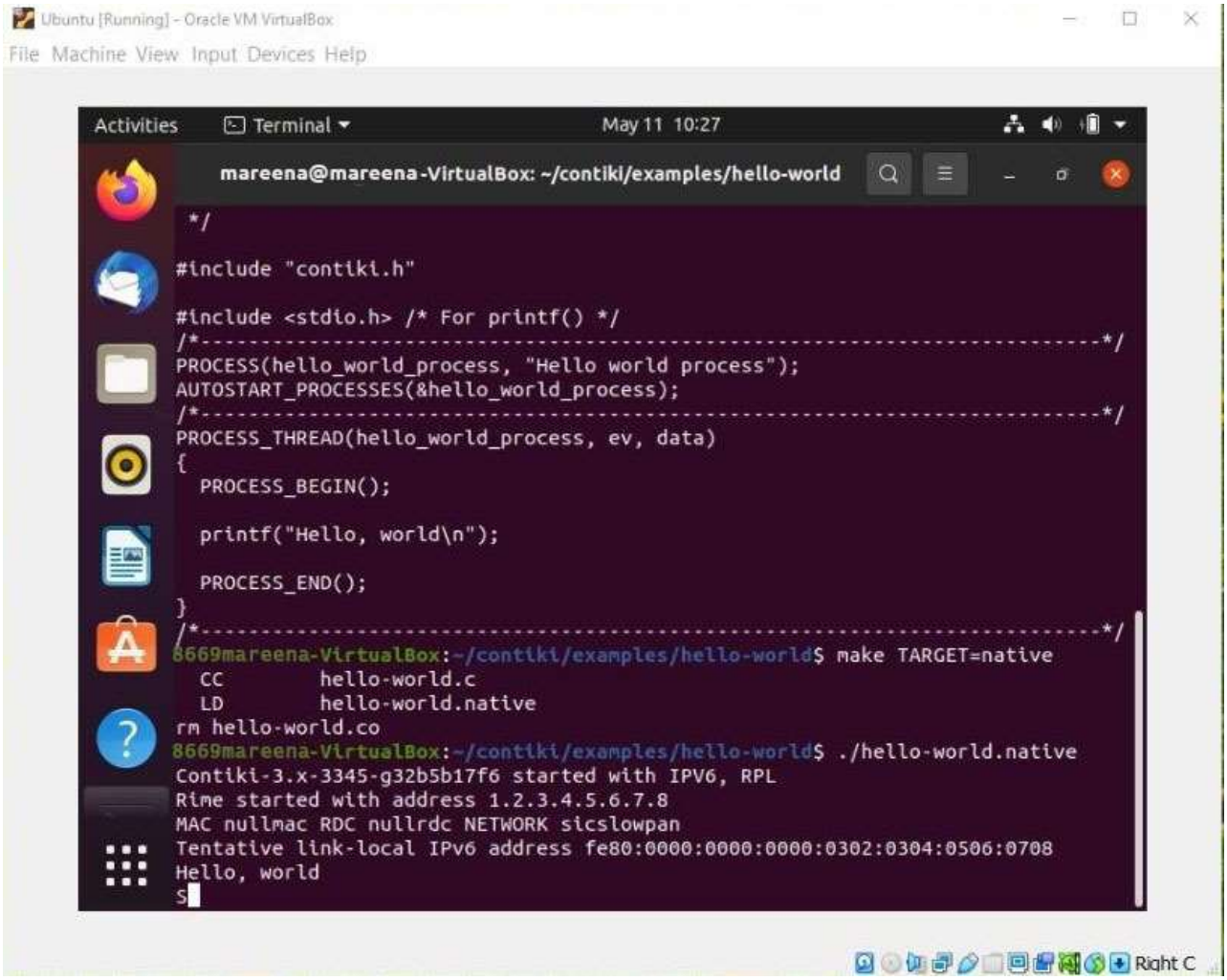
CONCLUSION: We understood the basic programming in Contiki and implemented Hello World program successfully.


Ubuntu [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
org-contikios-cooja-Cooja
May 11 10:19
mareena@mareena-VirtualBox: ~/contiki/tools/cooja
8669_Prac4 - Cooja: The Contiki Network Simulator
File Simulation Motes Tools Settings Help
Network
View Zoom
Notes
No help available
Run
Time:
Speed
Timeline
File Edit View Zoom Events Motes
Right C


Ubuntu [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
org-contikios-cooja-Cooja
May 11 10:21
mareena@mareena-VirtualBox: ~/contiki/tools/cooja
8669_Prac4 - Cooja: The Contiki Network Simulator
File Simulation Motes Tools Settings Help
Network
Network
Create Mote Type: Compile Contiki for sky
Description:
Sky Mote Type #sky2
Contiki process / Firmware:
/home/mareena/contiki/examples/hello-world.c
Browse
Clean
Compile
Create
Compile commands
Mote interfaces
Tips
Compilation output
> make hello-world.sky TARGET=sky
CC hello-world.c
CC ../../platform/sky/./contiki-sky-main.c
LD hello-world.sky
rm hello-world.co obj_sky/contiki-sky-main.o
Right C


Ubuntu [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
org-contikios-cooja-Cooja
May 11 10:21
mareena@mareena-VirtualBox: ~/contiki/tools/cooja
8669_Prac4 - Cooja: The Contiki Network Simulator
File Simulation Motes Tools Settings Help
Network
View Zoom
Run
Time:
Speed
Network
The network window shows the positions of simulated motes.
It is possible to zoom (Mouse wheel) and pan (Shift+Mouse drag) the current view. Motes can be moved by dragging them. You can add/remove motes to/from selection (CTRL+Left click) or use the rectangular selection tool (CTRL+Mouse drag). Mouse right-click motes for options menu.
The network window supports different views. Each view provides some specific information, such as the IP addresses of motes. Multiple views can be active at the same time. Use the View menu to select views.
Right C


Ubuntu [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
org-contikios-cooja-Cooja
May 11 10:23
mareena@mareena-VirtualBox: ~/contiki/tools/cooja
8669_Prac4 - Cooja: The Contiki Network Simulator
File Simulation Motes Tools Settings Help
View Zoom
Timeline
The timeline shows simulation events over time. The timeline can be used to inspect activities of individual nodes as well as interactions between nodes.
For each mote, simulation events are shown on a colored line. Different colors correspond to different events. For more information about a particular event, mouse click it.
The Events menu control what event types are shown in the timeline. Currently, six event types are supported (see below).
All motes are by default shown in the timeline. Motes can be removed from the timeline by right-clicking the node ID
Timeline showing 1 motes
File Edit View Zoom Events Motes
1
Right C


Ubuntu [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Terminal
May 11 10:27
mareena@mareena-VirtualBox: ~/contiki/examples/hello-world
*/
#include "contiki.h"
#include <stdio.h> /* For printf() */
/*---*/
PROCESS(hello_world_process, "Hello world process");
AUTOSTART_PROCESSES(&hello_world_process);
/*---*/
PROCESS_THREAD(hello_world_process, ev, data)
{
PROCESS_BEGIN();
printf("Hello, world\n");
PROCESS_END();
}
/*---*/
8669mareena-VirtualBox:~/contiki/examples/hello-world$ make TARGET=native
CC hello-world.c
LD hello-world.native
rm hello-world.co
8669mareena-VirtualBox:~/contiki/examples/hello-world$ ./hello-world.native
Contiki-3.x-3345-g32b5b17f6 started with IPV6, RPL
Rime started with address 1.2.3.4.5.6.7.8
MAC nullmac RDC nullrdc NETWORK sicslowpan
Tentative link-local IPv6 address fe80:0000:0000:0000:0302:0304:0506:0708
Hello, world
S
Right C