

# Assignment 2

## Task 1

For **etimer**

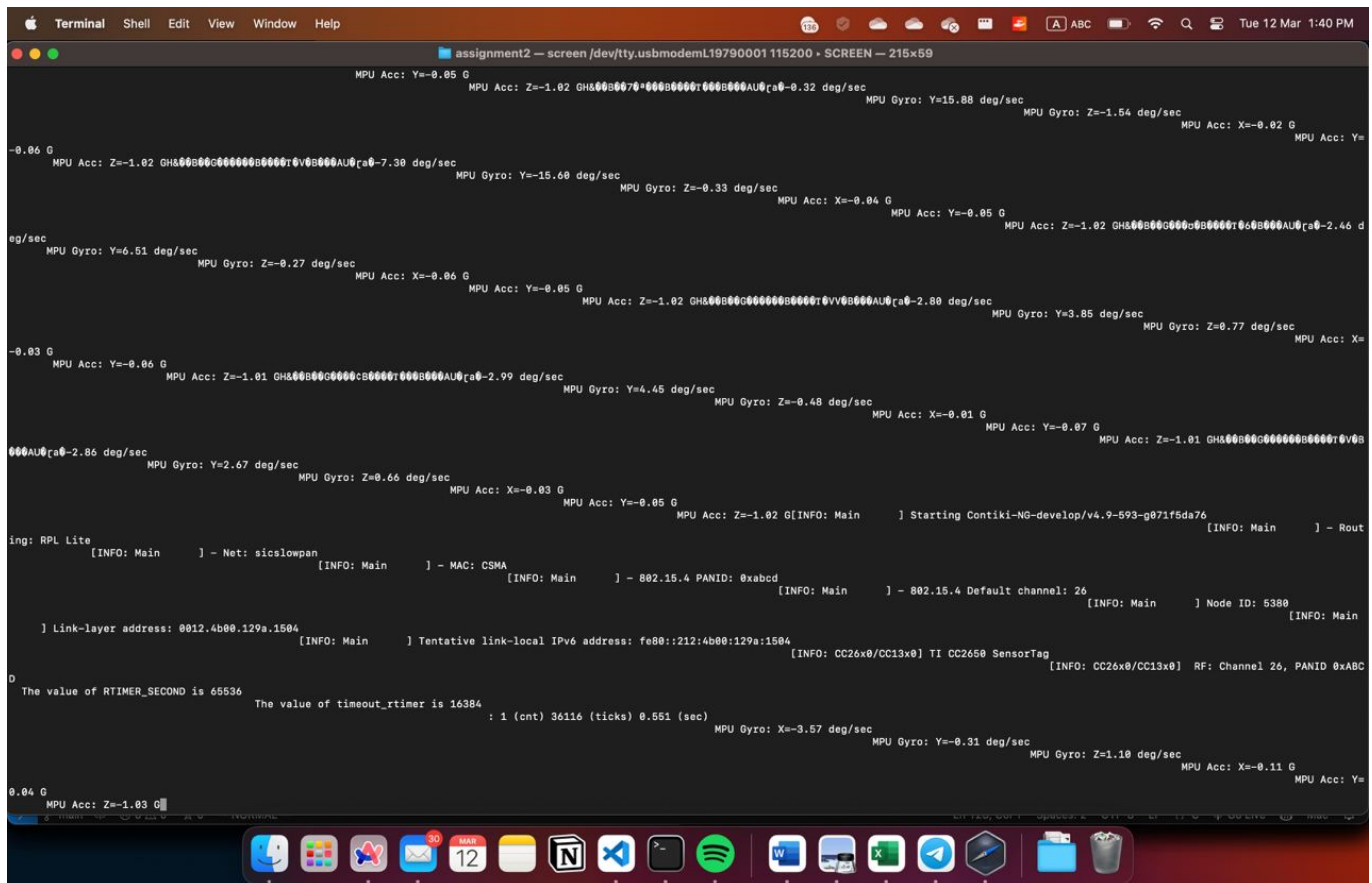
```
[INFO: Main ] Node ID: 5380
INFO: Main ] Link-layer address: 0012.4b00.129a.1504
[INFO: Main ] Tentative link-local IPv6 address: fe80::212:4b00:129a:1504
[INFO: CC26x0/CC13x0] TI CC2650 SensorTag
[INFO: CC26x0/CC13x0] RF: Channel 26, P
ANID 0xABCD
The value of CLOCK_SECOND is 128
Time(E): 1 (cnt) 166 (ticks) 1.296 (sec)
Time(E): 2 (cnt) 294 (ticks) 2.296 (sec)
Time(E): 3 (cnt) 422 (ticks) 3.296 (sec)
Time(E): 4 (cnt) 550 (ticks) 4.296 (sec)
Time(E): 5 (cnt) 678 (ticks) 5.296 (sec)
Time(E): 6 (cnt) 806 (ticks) 6.296 (sec)
Time(E): 7 (cnt) 934 (ticks) 7.296 (sec)
Time(E): 8 (cnt) 1062 (ticks) 8.296 (sec)
Time(E): 9 (cnt) 1190 (ticks) 9.296 (sec)
Time(E): 10 (cnt) 1318 (ticks) 10.296 (sec)
```

**CLOCK\_SECOND = 128**

```
assignment2 — screen /dev/tty.usbmodemL19790001115200 • SCREEN — 215x69
Time(E): 141 (cnt) 18087 (ticks) 141.304 (sec)
Time(E): 142 (cnt) 18215 (ticks) 142.304 (sec)
Time(E): 143 (cnt) 18343 (ticks) 143.304 (sec)
Time(E): 144 (cnt) 18471 (ticks) 144.304 (sec)
Time(E): 145 (cnt) 18599 (ticks) 145.304 (sec)
Time(E): 146 (cnt) 18727 (ticks) 146.304 (sec)
Time(E): 147 (cnt) 18855 (ticks) 147.304 (sec)
Time(E): 148 (cnt) 18983 (ticks) 148.304 (sec)
Time(E): 149 (cnt) 19111 (ticks) 149.304 (sec)
Time(E): 150 (cnt) 19239 (ticks) 150.304 (sec)
Time(E): 151 (cnt) 19367 (ticks) 151.304 (sec)
Time(E): 152 (cnt) 19495 (ticks) 152.304 (sec)
Time(E): 153 (cnt) 19623 (ticks) 153.304 (sec)
Time(E): 154 (cnt) 19751 (ticks) 154.304 (sec)
Time(E): 155 (cnt) 19879 (ticks) 155.304 (sec)
Time(E): 156 (cnt) 20007 (ticks) 156.304 (sec)
Time(E): 157 (cnt) 20135 (ticks) 157.304 (sec)
Time(E): 158 (cnt) 20263 (ticks) 158.304 (sec)
Time(E): 159 (cnt) 20391 (ticks) 159.304 (sec)
Time(E): 160 (cnt) 20519 (ticks) 160.304 (sec)
Time(E): 161 (cnt) 20647 (ticks) 161.304 (sec)
Time(E): 162 (cnt) 20775 (ticks) 162.304 (sec)
Time(E): 163 (cnt) 20903 (ticks) 163.304 (sec)
Time(E): 164 (cnt) 21031 (ticks) 164.304 (sec)
Time(E): 165 (cnt) 21159 (ticks) 165.304 (sec)
Time(E): 166 (cnt) 21287 (ticks) 166.304 (sec)
Time(E): 167 (cnt) 21415 (ticks) 167.304 (sec)
Time(E): 168 (cnt) 21543 (ticks) 168.304 (sec)
Time(E): 169 (cnt) 21671 (ticks) 169.304 (sec)
Time(E): 170 (cnt) 21799 (ticks) 170.304 (sec)
Time(E): 171 (cnt) 21927 (ticks) 171.304 (sec)
Time(E): 172 (cnt) 22055 (ticks) 172.304 (sec)
Time(E): 173 (cnt) 22183 (ticks) 173.304 (sec)
Time(E): 174 (cnt) 22311 (ticks) 174.304 (sec)
Time(E): 175 (cnt) 22439 (ticks) 175.304 (sec)
Time(E): 176 (cnt) 22567 (ticks) 176.304 (sec)
Time(E): 177 (cnt) 22695 (ticks) 177.304 (sec)
Time(E): 178 (cnt) 22823 (ticks) 178.304 (sec)
Time(E): 179 (cnt) 22951 (ticks) 179.304 (sec)
Time(E): 180 (cnt) 23079 (ticks) 180.304 (sec)
Time(E): 181 (cnt) 23207 (ticks) 181.304 (sec)
Time(E): 182 (cnt) 23335 (ticks) 182.304 (sec)
Time(E): 183 (cnt) 23463 (ticks) 183.304 (sec)
Time(E): 184 (cnt) 23591 (ticks) 184.304 (sec)
Time(E): 185 (cnt) 23719 (ticks) 185.304 (sec)
Time(E): 186 (cnt) 23847 (ticks) 186.304 (sec)
Time(E): 187 (cnt) 23975 (ticks) 187.304 (sec)
Time(E): 188 (cnt) 24103 (ticks) 188.304 (sec)
```

Taking the clock ticks count between 2 consecutive prints, for example, 24103-23975=128. **number of clock ticks = 128 per second**

For **rtimer**



RTIMER\_SECOND = 65536

RTIMER clock ticks = 16384

## Instruction of running the program

Ensure that you are in Linux environment and with Contiki OS and UniFlash software installed. You may refer to the [setup guide](#) to install and set up all necessary dependencies if you haven't done so.

Furthermore, to run our programs for task 2 and task 3, ensure you copy the **Makefile** to the same directory of **task2.c** and **task3.c**.

### Task 2

1. Run the following command to compile and build the binary file:

```
make TARGET=cc26x0-cc13x0 BOARD=sensortag/cc2650 task2
```

2. You will see a binary file called **task2.cc26x0-cc13x0** generated in the same folder.
3. Load **task2.cc26x0-cc13x0** to UniFlash and run the program on the sensor. Then you may start monitoring the behaviours of the CC2650 microcontroller.

### Task 3

1. Run the following command to compile and build the binary file:

```
make TARGET=cc26x0-cc13x0 BOARD=sensortag/cc2650 task3
```

- 2. You will see a binary file called `task3.cc26x0-cc13x0` generated in the same folder.
- 3. Load `task3.cc26x0-cc13x0` to UniFlash and run the program on the sensor. Then you may start monitoring the behaviours of the CC2650 microcontroller.

Group member

Name	Student ID
Galvin Chan Shi Yuan	A0217812J
Mun Le Zong	A0218136J
Swann Tet Aung	A0217516H
Tan Rui Yang	A0219814B