

Name: Enrique Saldana
Partner: Damian Cisneros
Github root directory: <https://github.com/enri10>

Date Submitted: 11-25-2018

Unable to complete first Task01-Task04 with the flashing and using BLE due to Android app not having functions from iOS app as shown in the PDF instructions. Android support does not seem active and we do not have an iPhone.

Task 01: Update CC1350 LaunchPad firmware

The screenshot shows the UniFlash application interface. The top bar is red with 'UniFlash', 'Session', and 'About' menus, along with 'Help' and 'Settings' icons. Below the bar, the configured device is 'Texas Instruments XDS110 USB Debug Probe' with a target of 'CC1350F128'. The main panel is titled 'Select and Load Images' and shows a list of flash images. One image, 'CC1350LaunchPad_ExtFlashErase.hex', is selected, with a size of 290.62 KB. Below the list, there are buttons for 'Load Image' and 'Verify Image'. The 'Run Actions' section has a checkbox for 'Run Target After Program Load/Flash Operation' which is checked. The 'Console' panel at the bottom shows the following log messages:

```
[12/3/2018, 8:34:07 PM] [INFO] Cortex_M3_0: GEL Output: Memory Map Initialization Complete.  
[12/3/2018, 8:34:09 PM] [INFO] Cortex_M3_0: GEL Output: Board Reset Complete.  
[12/3/2018, 8:34:15 PM] [SUCCESS] Program Load completed successfully.
```

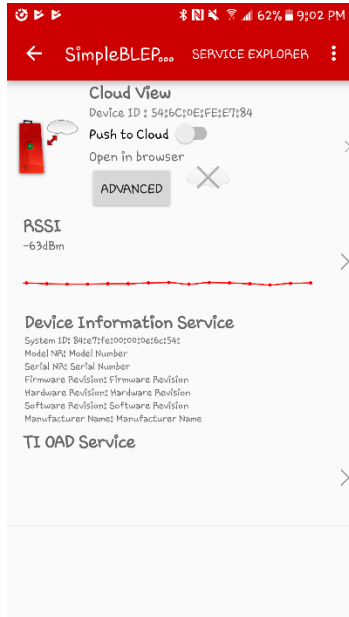
Successfully loading cc1350launchpad_extflasherase.hex

The screenshot shows the UniFlash application interface. The top bar is red with 'UniFlash', 'Session', and 'About' menus, along with 'Help' and 'Settings' icons. Below the bar, the configured device is 'Texas Instruments XDS110 USB Debug Probe' with a target of 'CC1350F128'. The main panel is titled 'Select and Load Images' and shows a list of flash images. One image, 'simple_peripheral_cc1350lp_all.hex', is selected, with a size of 323.57 KB. Below the list, there are buttons for 'Load Image' and 'Verify Image'. The 'Run Actions' section has a checkbox for 'Run Target After Program Load/Flash Operation' which is checked. The 'Console' panel at the bottom shows the following log messages:

```
[12/3/2018, 8:34:07 PM] [INFO] Cortex_M3_0: GEL Output: Memory Map Initialization Complete.  
[12/3/2018, 8:34:09 PM] [INFO] Cortex_M3_0: GEL Output: Board Reset Complete.  
[12/3/2018, 8:34:15 PM] [SUCCESS] Program Load completed successfully.  
[12/3/2018, 8:36:45 PM] [INFO] Cortex_M3_0: GEL Output: Memory Map Initialization Complete.  
[12/3/2018, 8:36:47 PM] [INFO] Cortex_M3_0: GEL Output: Board Reset Complete.  
[12/3/2018, 8:36:54 PM] [SUCCESS] Program Load completed successfully.
```

Successfully loading simple_peripheral_cc1350lp_all.hex

Grading scheme: 30% Coding, 30% Documentation, 40% Execution/Video.



Able to connect to it but no capability of sending firmware like shown on iOS app.

Youtube Link: No video required.

Modified Code: No coding required

WSN Concentrator and WSN Node Part

Task 01: Importing the WSN Examples



WSN Concentrator waiting for nodes. Node is not connected at this point.

Youtube Link: No video required for this part.

Modified Code: Same imported WSN Concentrator/Node project

Task 02: Putting it all to work



```
COM5 - PuTTY
Nodes  Value  SW  RSSI
0x0a   0x33a  0   -020
0x01   0x306  0   -014
```

Concentrator receiving values.



```
COM8 - PuTTY
Node ID: 0x01
Node ADC Reading: 0771
Advertiser Mode: Eddystone URL
Advertisement success: 436 out of 474
```

Node sending values.

Youtube Link: <https://youtu.be/IeDKvveIXQQ>

Modified Code: No code was modified from initial project import.

Task 03: Change RF Channel

```
/* If you wish to use a frequency other than the default use
 * the below API
 * EasyLink_setFrequency(868000000);
 */
EasyLink_setFrequency(868000000);
```

Modified frequency in ConcentratorRadioTask.c (highlighted). Operation is the same as Task02.

Modified Code: Highlighted portion above was modified in ConcentratorRadioTask.c

Task 04: Switch from 2-GFSK 50 kbps to Long Range Mode (LRM)

```
39 #define RADIO_CONCENTRATOR_ADDRESS    0x00
40 #define RADIO_EASYLINK_MODULATION      EasyLink_Phy_5kbpsS1Lr
```

Switched to long range mode on Radioprotocol.h (highlighted in blue)

Modified Code: Highlighted portion above was modified in Radioprotocol.h

Task 06: Modify Code Running on the Sensor Controller

ADC Sample - Execution Code

```
1 // Enable the ADC
2 adcEnableSync(ADC_REF_FIXED, ADC_SAMPLE_TIME_2P7_US, ADC_TRIGGER_MANUAL);
3
4 // Sample the ADC
5 S16 adcValue;
6 adcGenManualTrigger();
7 adcReadFifo(adcValue);
8 output.adcValue = adcValue;
9
10 // Disable the ADC
11 adcDisable();
12
13 // Alert the driver if outside of change mask
14 U16 adcMaskedBits = adcValue & cfg.changeMask;
15 if (adcMaskedBits != state.oldAdcMaskedBits) {
16     fwGenAlertInterrupt();
17     state.samplesSinceLastReport = 0;
18 } else {
19     state.samplesSinceLastReport = state.samplesSinceLastReport + 1;
20 }
21
22 //Alert driver if minimum report interval has expired
23 if(cfg.minReportInterval != 0) {
24     if(state.samplesSinceLastReport >= cfg.minReportInterval) {
25         fwGenAlertInterrupt();
26         state.samplesSinceLastReport = 0;
27     }
28 }
29
30 // Save old masked ADC value
31 state.oldAdcMaskedBits = adcValue & cfg.changeMask;
32
33 // Schedule the next execution
34 fwScheduleTask(2);
```

ADC Sample modified to read every 2s instead of 1s.