CPE403 – Advanced Embedded Systems

Design Assignment #5

DO NOT REMOVE THIS PAGE DURING SUBMISSION:

Name: Joe Uesato

Email: uesatj1@unlv.nevada.edu

Github Repository link (root): https://github.com/joeuesato/lab_submissions

Youtube Playlist: https://www.youtube.com/playlist?list=PLSBOvuRedzOf8JAhpVx0VsSteisJQUKv3

Follow the submission guideline to be awarded points for this Assignment.

Submit the following for all Assignments:

1. In the document, for each task submit the modified or included code (from the base code) with highlights and justifications of the modifications. Also include the comments. If no base code is provided, submit the base code for the first task only.
2. Create a private Github repository with a random name (no CPE/403, Lastname, Firstname). Place all labs under the root folder TIVAC, sub-folder named Assignment1, with one document and one video link file for each lab, place modified c files named as asng_taskxx.c.
3. If multiple c files or other libraries are used, create a folder asng1_t01 and place these files inside the folder.
4. The folder should have a) Word document (see template), b) source code file(s) with startup_ccs.c and other include files, c) text file with youtube video links (see template).
5. Submit the doc file in canvas before the due date. The root folder of the github assignment directory should have the documentation and the text file with youtube video links.
6. Organize your youtube videos as playlist under the name “cpe403”. The playlist should have the video sequence arranged as submission or due dates.
7. Only submit pdf documents. Do not forget to upload this document in the github repository and in the canvas submission portal.

1. Code for Tasks. for each task submit the modified or included code (from the base code) with highlights and justifications of the modifications. Also include the comments. If no base code is provided, submit the base code for the first task only. Use separate page for each task.

Modified code for outputting sensor data

```
else
{
    CUI_statusLinePrintf(csfCuiHndl, deviceStatusLine, "Sensor - Addr=0x%04x, Temp=%d, Humidity=%d, Light=%d, RS
                pSrcAddr->addr.shortAddr,
                pMsg->humiditySensor.temp,
                pMsg->humiditySensor.humidity,
                pMsg->lightSensor.rawData,
                rssi);
```

Enabling POSIX libraries

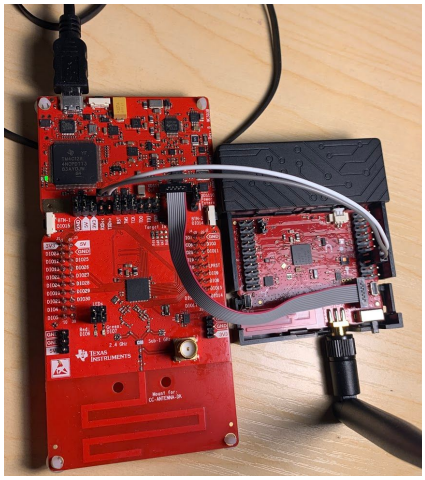
```
628 /* ================ POSIX configuration ================ */
629 var Settings = xdc.useModule('ti.posix.tirtos.Settings');
630
```

2. Block diagram and/or Schematics showing the components, pins used, and interface.

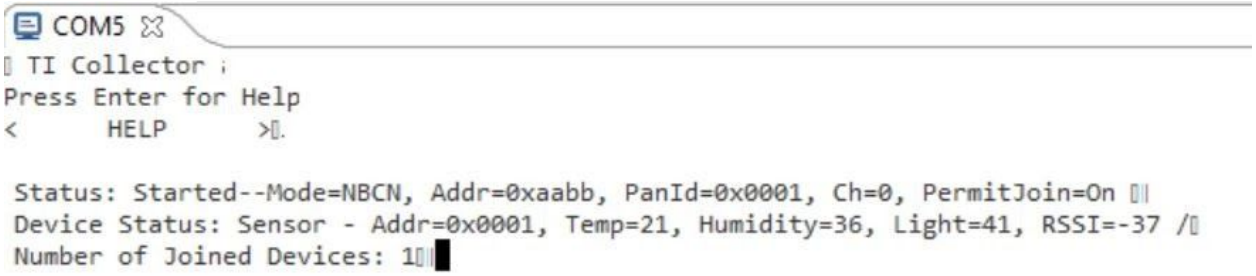
Only used CC1352 Launchpad and LPSTK-CC1352R using the wireless connection to transmit sensor data over the network to the Launchpad

3. Screenshots of the IDE, physical setup, debugging process - Provide screenshot of successful compilation, screenshots of registers, variables, graphs, etc.

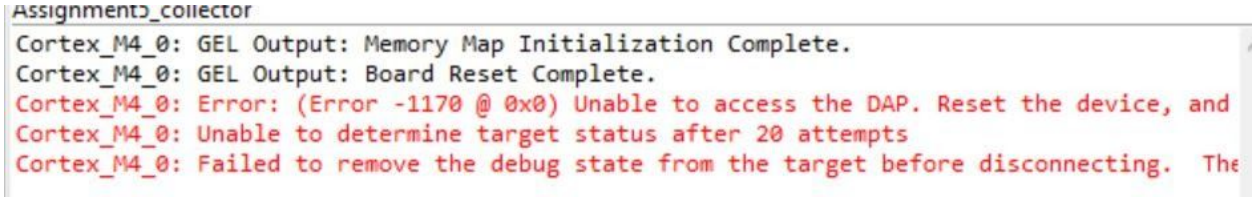
Here is the connection trying to get the LPSTK programmed with the sensor code using the JTAG pins



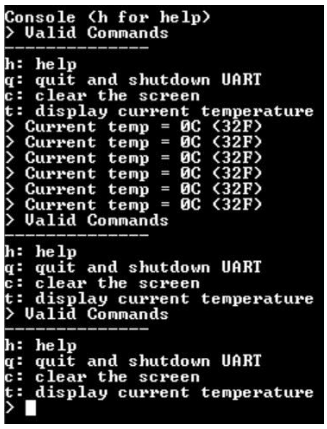
This is showing the terminal output showing the collector outputting over UART the data transmitted from the LPSTK sensor node.



Interesting errors - tried clearing flash memory, hard resetting/power cycling boards, redownloading and rebuilding projects, and couldn't get it to go away. However, it still programmed and ran fine



Terminal UART from portable application



Commented out redefinitions

```
114     /* Initialize the GPIO since multiple threads are using it */
115 //    GPIO_init();
116
117     /* Start the TI-RTOS scheduler */
118 //    BIOS_start();
119
```

4. Declaration

I understand the Student Academic Misconduct Policy - http://studentconduct.unlv.edu/misconduct/policy.html

"This assignment submission is my own, original work".

Joe Uesato