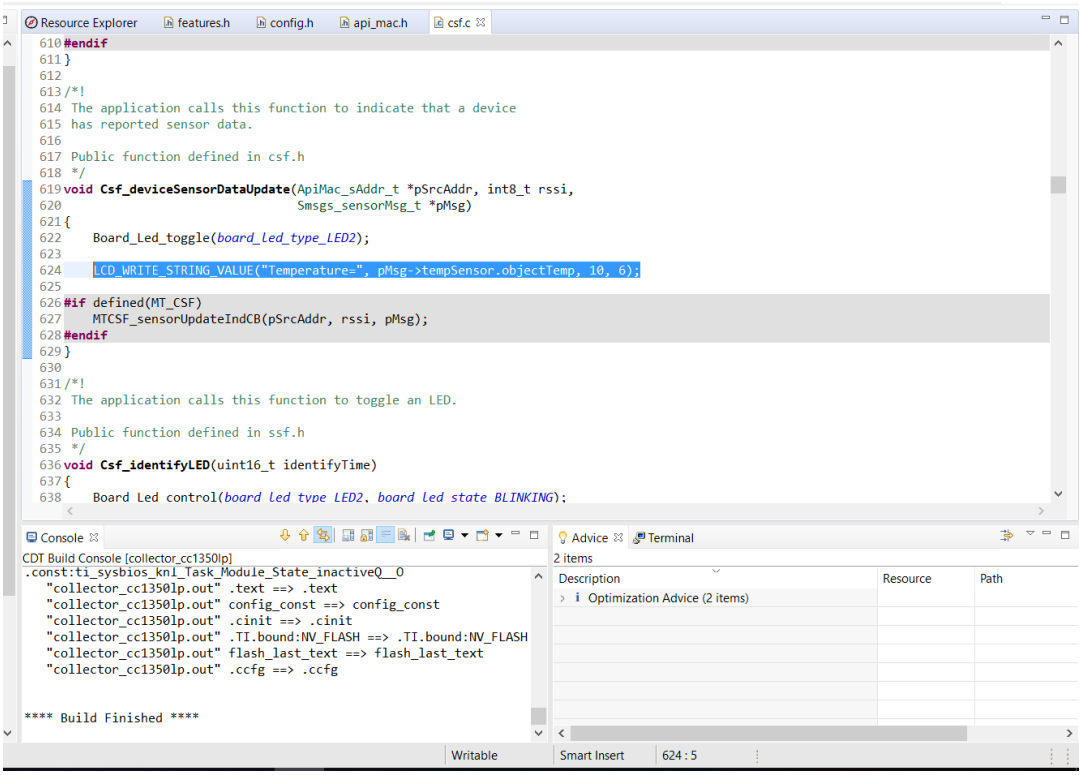
Name: Enrique Saldana
Partner: Damian Cisneros
Github root directory: https://github.com/enri10

Date Submitted: 11-25-2018

Task 1+2: Building and loading the collector example and the sensor example.

```
163 #define CONFIG_CHANNEL_MASK          { 0x00, 0x00, 0x04, 0x00, 0x00, 0x00, \
164                                          0x00, 0x00, 0x00, 0x00, 0x00, 0x00, \
165                                          0x00, 0x00, 0x00, 0x00, 0x00 }
166 /*!
```

Channel mask config for both collector and sensor.



Modified line from csf.c on collector to print out temperature. (highlighted blue)

Youtube Link: No video required for this part.

Modified Code: highlighted in image above.

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

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

Task 03: **Using the Collector and Sensor**

Sensor UART. Successfully starting.

Collector UART. Successfully starting.

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

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

Collector UART. Sensor has joined and sent over temperature.

Youtube Link: https://youtu.be/EeSAvy8vkXU

Modified Code: No code was modified.

Task 04: Updating the sensor's reporting rate

```
#if (((CONFIG_PHY_ID >= APIMAC_MRFSK_STD_PHY_ID_BEGIN) && (CONFIG_PHY_ID <= APIMAC_MRFSK_GENERIC_PHY_ID_BEGIN)) || \
    ((CONFIG_PHY_ID >= APIMAC_GENERIC_US_915_PHY_132) && (CONFIG_PHY_ID <= APIMAC_GENERIC_ETSI_863_PHY_133)))
/*! Default Polling interval in milliseconds. It will get updated upon reception
 of a config request message */
#define CONFIG_POLLING_INTERVAL      100
/*! PAN Advertisement Solicit trickle timer duration in milliseconds */
#define CONFIG_PAN_ADVERT_SOLICIT_CLK_DURATION    6000
/*! PAN Config Solicit trickle timer duration in milliseconds */
#define CONFIG_PAN_CONFIG_SOLICIT_CLK_DURATION    6000
/*! Default Reporting Interval - in milliseconds. It will get updated upon
 reception of a config request message */
#define CONFIG_REPORTING_INTERVAL  500
#else
/*! Default Polling interval in milliseconds. It will get updated upon reception
 of a config request message */
#define CONFIG_POLLING_INTERVAL      60000
/*! PAN Advertisement Solicit trickle timer duration in milliseconds */
#define CONFIG_PAN_ADVERT_SOLICIT_CLK_DURATION    60000
/*! PAN Config Solicit trickle timer duration in milliseconds */
#define CONFIG_PAN_CONFIG_SOLICIT_CLK_DURATION    60000
/*! Default Reporting Interval - in milliseconds. It will get updated upon
 reception of a config request message */
#define CONFIG_REPORTING_INTERVAL  600000
#endif
```

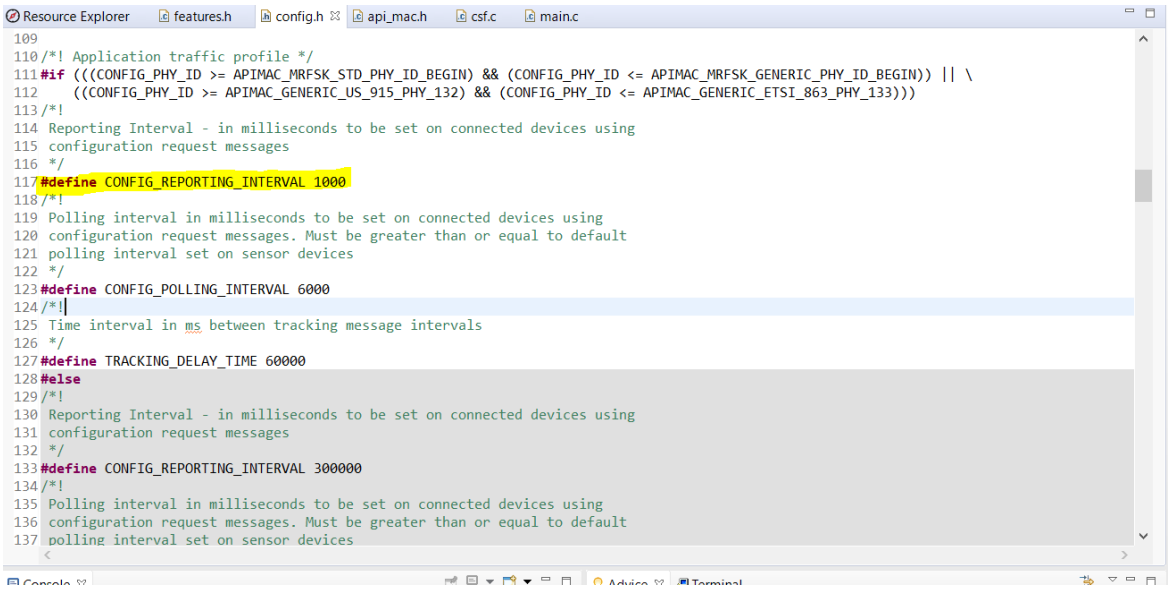
Reporting and Polling intervals changed on sensor config. (highlighted in yellow)

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

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

```
/* Polling Interval Min and Max (in milliseconds) */
#define MIN_POLLING_INTERVAL 100
#define MAX_POLLING_INTERVAL 10000
```

Minimum polling interval changed in sensor sensor.c. (highlighted in yellow)



Reporting interval change for collector config. (highlighted in yellow)

Youtube Link: https://youtu.be/XFslj5QEuDs

Modified Code: Highlighted in yellow on images above.

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