Abenezer Namaga
Github root directory: https://github.com/Ayertena/CC1350-LABs

Date Submitted: 11/16/2018

```c
#include <unistd.h>
#include <stdint.h>
#include <stddef.h>

/* Driver Header files */
#include <ti/drivers/GPIO.h>
#include <ti/drivers/ADC.h>
#include <ti/display/Display.h>
// #include <ti/drivers/I2C.h>
// #include <ti/drivers/SDSPI.h>
// #include <ti/drivers/SPI.h>
// #include <ti/drivers/UART.h>
// #include <ti/drivers/Watchdog.h>

/* Board Header file */
#include "Board.h"

/* global variableS FOR GUI COMPOSER */
uint16_t adcValue = 0;
uint16_t threshold = 100;
uint16_t trigger = 0;

/*
 *  ======== mainThread ========
 */

void *mainThread(void *arg0)
{
    /* ~10 loops/second */
    uint32_t time = 100000;  // update ~10/second

    /* Call driver init functions */
    GPIO_init();
    ADC_init();
    // I2C_init();
    // SDSPI_init();
    // SPI_init();
    // UART_init();
    // Watchdog_init();

    /* Open ADC Driver */
```

```
      ADC_Handle adc;
      ADC_Params params;
      ADC_Params_init(&params);
      adc = ADC_open(Board_ADC0, &params);
      if (adc == NULL) {
          // Error initializing ADC channel 0
          while (1);
      }

      /* Open Display Driver */
      Display_Handle    displayHandle;
      Display_Params    displayParams;
      Display_Params_init(&displayParams);
      displayHandle = Display_open(Display_Type_UART, NULL);

      while (1) {
          int_fast16_t res;
          res = ADC_convert(adc, &adcValue);
          if (res == ADC_STATUS_SUCCESS) {
              Display_printf(displayHandle, 1, 0, "ADC Reading %d",
adcValue);

              if(adcValue >= threshold){
                  GPIO_write(Board_GPIO_LED0, Board_GPIO_LED_ON);
                  trigger = 1;
              } else{
                  GPIO_write(Board_GPIO_LED0, Board_GPIO_LED_OFF);
                  trigger = 0;
              }
          }

          usleep(time);
      }
  }
```