Lab Assignment 1

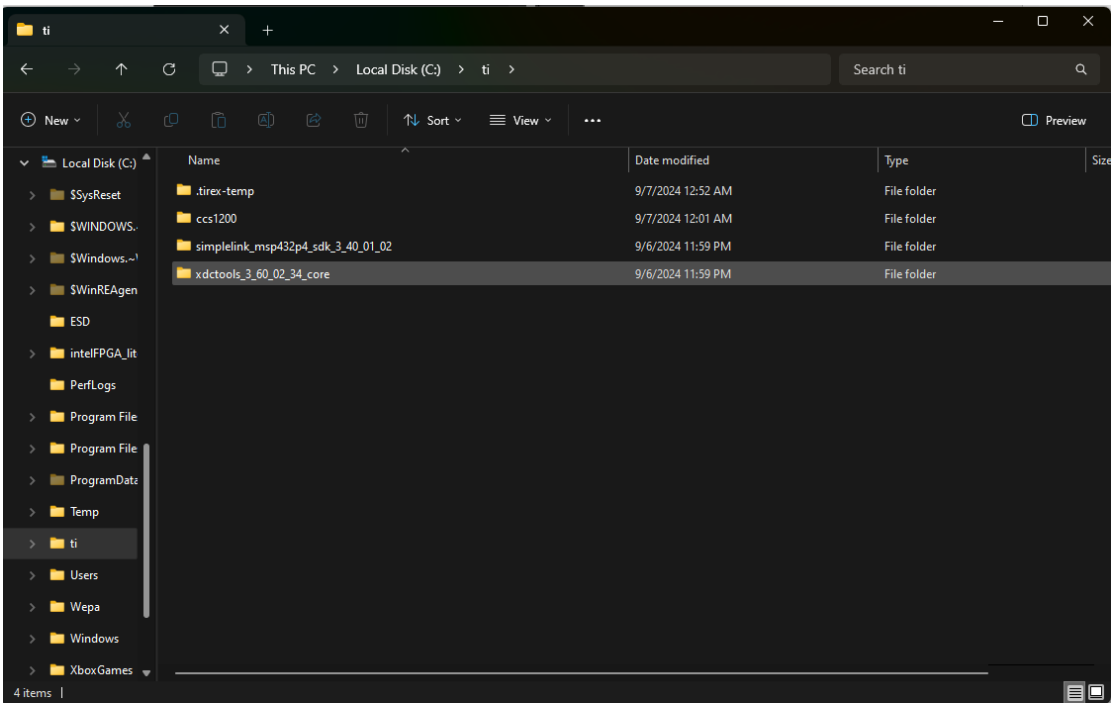
Jose Monroy Villalobos

San José State University
Department of Computer Engineering
CMPE 146-01, Real-Time Embedded System Co-Design, Fall 2024

Exercise 1 CCS and SDK

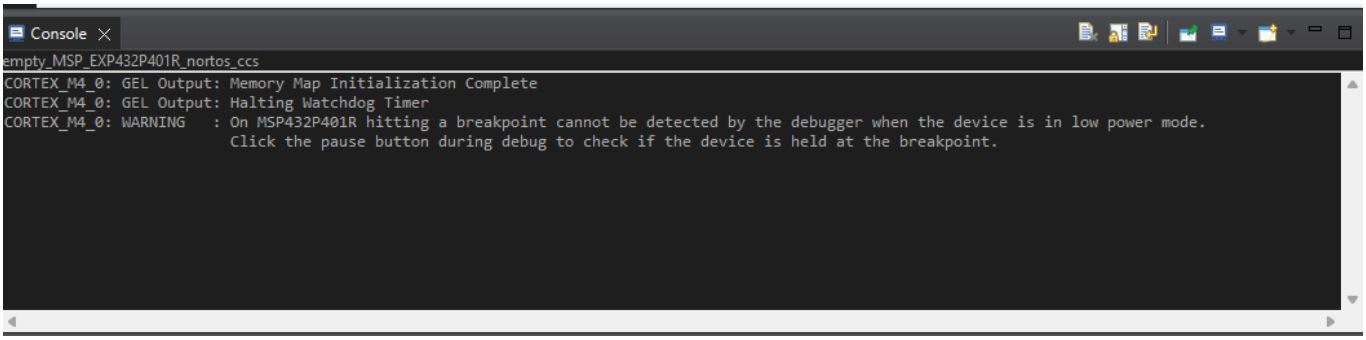
For the Installation of the Code Composer studio, I followed the direction and went to the website for the CCS and downloaded the newest version which was 12.8.0 and then I went to the google drive and downloaded the SDK. I opted to use a windows environment for this lab and encountered some issues. The debugger needed to be updated to version 31 and it was version 28. When attempting to update the board, the green LED would turn off. CCS told me try to update using a utility so I went the terminal and attempted to update that way. The board would turn off the green led and the attempt would fail. I proceeded to uninstall both CCS and SDK and deleted the TI folder and did a complete reinstall with CSS version 12.0.0 instead. This resolved my issues, and I was able to proceed with this lab.

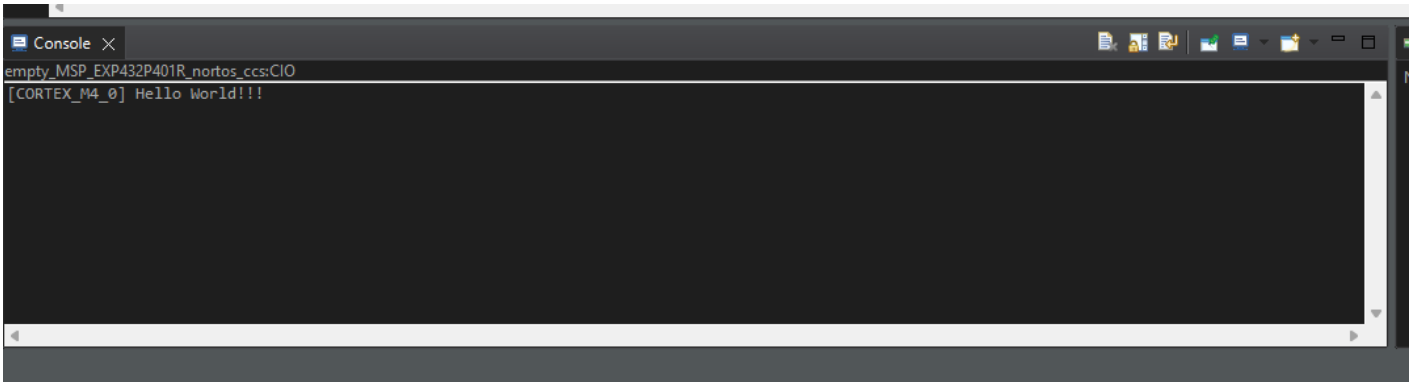
Screenshot of the TI folder showing the CCS and SDK folders



Exercise 2 Hello World

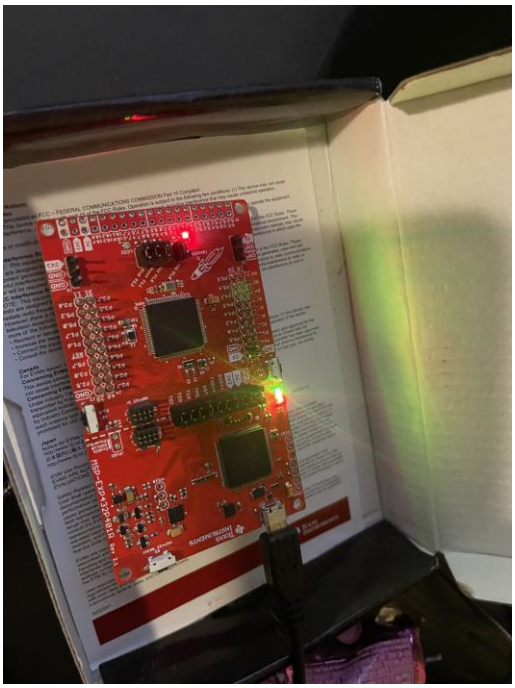
Screenshot of the debug console showing the console tab and the outputs





Exercise 3 Blink LED

Picture showing the LED lit



5000 For Loop Link

Raw Link: https://youtu.be/0Y17rPpufBc?si=zBX5mDIeLUqH_Be6

50000 For Loop Link

Raw Link:

https://youtu.be/8XAO7bepZ0k?si=PNmS8GRSMXI2MsNB

Exercise 4 Access MCU Info

Code Snippet that reads and prints:

***DISCLAIMER* If you want to test this Program please go to the end of the document and use the full code provided in the appendix for exercise 4**

```
//TLV CHECKSUM VALUE is printed
printf("\n\nTLV CHECKSUM: %08x",*addressptr);
addressptr++;


while(*addressptr!=TLV_END_VALUE)
{
   //print and assign tag value;
   printf("\nTag: %08x ", *addressptr);
   tag = *addressptr;
   addressptr++;
   printf("Length %08x ", *addressptr);
   //print and assign length value;
   length = *addressptr;
   addressptr++;
   //utilize for loop to go through block and print data
   for(i=0;i<length;i++){
      printf(" Data %08x ", *addressptr);
      addressptr++;
   }
 }
//TLV END value is printed
printf("\nTLV END VALUE: %08x\n", *addressptr);
```

I print the checksum and the endvalue separate from the data blocks inbetween as they differ from them due to them lacking tag and length values.

Output for exercise 4:

TLV CHECKSUM: a77e4ddc

Tag: 0000000b Length 00000004 Data 0000a000 Data 00000044 Data 00420044 Data 03000010

Tag: 0000000c Length 00000008 Data 00000037 Data 00000054 Data 0000000a Data 00011412 Data fffffc42 Data 00000000 Data 00000000 Data 00474345

Tag: 00000003 Length 00000010 Data 0000019c Data 0000019b Data 000000c0 Data 000002c0 Data 0000002a Data 000002c0 Data 3ba73088 Data 3ba57299 Data 000001ca Data 000001c9 Data 000000b4 Data 000002c0 Data 00000028 Data 000002c0 Data 3ba1288a Data 3b9e180a

Tag: 00000005 Length 00000018 Data 0000fffd Data 00000000 Data 00000000 Data 00000000 Data 00000000 Data 00000000 Data 00000000 Data 00000000 Data 00000000 Data 00000000 Data 00000000 Data 00000000 Data 0000fff1 Data 00000000 Data 00000000 Data 00000000 Data 0000fffd Data 0000fff1 Data 0000288c Data 00002e84 Data 00002190 Data 00002680 Data 00001377 Data 00001652

Tag: 00000008 Length 00000003 Data 3f9986ae Data 3fb96146 Data 401fa863

Tag: 00000004 Length 00000002 Data 00000005 Data 0000014e

Tag: 0000000d Length 00000004 Data d259c170 Data 81e4d7af Data ae77c631 Data 61ae418a

Tag: 0000000f Length 00000004 Data ffc2d0c0 Data fcfffda0 Data f0ff9770 Data fcffff72

TLV END VALUE: 0bd0e11d

Appendix:

Exercise 2 Code:

```
/* DriverLib Includes */
#include <ti/devices/msp432p4xx/driverlib/driverlib.h>

/* Standard Includes */
#include <stdint.h>
#include <stdbool.h>
#include <stdio.h>

#define Start_address 0x00201000
#define TLV_END_VALUE 0x0bd0e11d
int main(void)
{
    uint32_t *addressptr = (uint32_t*)Start_address;

    uint32_t i, tag, length;

    //TLV CHECKSUM VALUE is printed
    printf("\n\nTLV CHECKSUM: %08x",*addressptr);
    addressptr++;


    while(*addressptr!=TLV_END_VALUE)
    {
        //print and assign tag value;
        printf("\nTag: %08x ", *addressptr);
```

```
        tag = *addressptr;
        addressptr++;
        printf("Length %08x ", *addressptr);
        //print and assign length value;
        length = *addressptr;
        addressptr++;

        //utilize for loop to go through block and print data
        for(i=0;i<length;i++){
            printf(" Data %08x ", *addressptr);
            addressptr++;
        }

          }

    //TLV END value is printed
    printf("\nTLV END VALUE: %08x\n", *addressptr);
}
```

Exercise 3 Code:

```
/* DriverLib Includes */
#include <ti/devices/msp432p4xx/driverlib/driverlib.h>

/* Standard Includes */
#include <stdint.h>
#include <stdbool.h>

//![Simple GPIO Config]
```

```
int main(void)
{
    volatile uint32_t ii;

    /* Halting the Watchdog */
    MAP_WDT_A_holdTimer();

    /* Configuring P1.0 as output */
    MAP_GPIO_setAsOutputPin(GPIO_PORT_P1, GPIO_PIN0);

    while (1)
    {
        /* Delay Loop */
//This ii was changed from 5000 to 50000
        for(ii=0;ii<50000;ii++)
        {
        }

        MAP_GPIO_toggleOutputOnPin(GPIO_PORT_P1, GPIO_PIN0);
    }
}
```

Exercise 4 Code:

```
/* DriverLib Includes */
#include <ti/devices/msp432p4xx/driverlib/driverlib.h>

/* Standard Includes */
#include <stdint.h>
```

```
#include <stdbool.h>
#include <stdio.h>

#define Start_address 0x00201000
#define TLV_END_VALUE 0x0bd0e11d
int main(void)
{
   uint32_t *addressptr = (uint32_t*)Start_address;

   uint32_t i, tag, length;

   //TLV CHECKSUM VALUE is printed
   printf("\n\nTLV CHECKSUM: %08x",*addressptr);
   addressptr++;


   while(*addressptr!=TLV_END_VALUE)
   {
      //print and assign tag value;
      printf("\nTag: %08x ", *addressptr);
      tag = *addressptr;
      addressptr++;
      printf("Length %08x ", *addressptr);
      //print and assign length value;
      length = *addressptr;
      addressptr++;

      //utilize for loop to go through block and print data
      for(i=0;i<length;i++){
         printf(" Data %08x ", *addressptr);
         addressptr++;
      }
```

```
        }

    //TLV END value is printed
    printf("\nTLV END VALUE: %08x\n", *addressptr);
}
```