Preparation for Labs

C programming basics

- Functions
- Structures
- Pointers

- Bit Masking Tutorial
- Basic parts and toolkit
  - Look for any basic starter kit on amazon - mostly Breadboard, Jumper cables, LEDs, Potentiometer, Buttons, Resistors
  - Something like this

Part 0: Compile the sample project

In this part, the objective is simply to compile your project, and to make sure you can load the compiled image onto your board.

- Erase the contents of your existing main.c and use the code given below
- Use the Google Chrome based serial terminal to confirm the output

```
// file: main.c
#include <stdio.h>
// Separate include files for Clang to sort separately
#include "delay.h"
void main(void) {
  unsigned counter = 0;

  while (1) {
    printf("Running: %u\n", counter);
    ++counter;
    delay__ms(1000U);
  }
```

```
}
```

Part 1: Get familiar with GPIO Blinky API

The objectives of this part is:

- Learn how to use existing API to blink LEDs
- Get familiar with how to develop clean C APIs
- Explore the code of the existing APIs to assess how it works

```
#include <stdio.h>
#include "board_io.h"
#include "delay.h"
#include "gpio.h"
/* This is just a sample
 * Cross-reference the schematic, and browse through the header files included above
 * to go above and beyond, and in general play with existing APIs
 */
int main(void) {
  gpio_s led0 = board_io__get_led0();
  while (1) {
    gpio__toggle(led0);
    delay__ms(500U);
  }
  return 0;
}
```

Conclusion

This exercise was mostly to get familiar with the board, and be excited about how your software can control hardware. In the following weeks, you will be developing your own code by directly manipulating the microcontroller's memory to achieve your objectives.

Revision #12
Created 3 years ago by sree harsha
Updated 3 years ago by Preet Kang