

main_master.cpp

```
1/*
2 * CMPE 146: I2C Lab main_master.cpp
3 */
4
5/**
6 * @file
7 * @brief This is the application entry point.
8 */
9#include <stdio.h>
10#include "utilities.h"
11#include "io.hpp"
12#include <tasks.hpp>
13#include "i2c2.hpp"
14#include "time.h"
15void vCalculate(void *pvParameters){
16    uint8_t op_1, op_2, opr, result;
17    I2C2& i2c = I2C2::getInstance(); // Get I2C driver instance
18    const uint8_t slaveAddr = 0xC0; // Pick any address other than an existing
    one at i2c2.hpp
19    while (1){
20        uint8_t arr[3] = { 0 };
21        op_1 = rand() % 16;
22        op_2 = rand() % 16;
23        opr = rand() % 3;
24        arr[0] = op_1;
25        arr[1] = op_2;
26        arr[2] = opr;
27        i2c.writeRegisters(slaveAddr, 0x01, arr, 3);
28        vTaskDelay(500);
29        result = i2c.readReg(0xc0, 0x04);
30        switch (opr){
31            case 0:
32                if ((op_1 + op_2) == result){
33                    printf("%u + %u = %u\n", op_1, op_2, result);
34                }
35                else {
36                    printf("error: got %u + %u = %u\nexpected %u + %u = %u\n",
    op_1, op_2, result, op_1, op_2, op_1 + op_2);
37                }
38                break;
39            case 1:
40                if (op_1 > op_2){
41                    if (result == (op_1 - op_2)){
42                        printf("%u - %u = %u\n", op_1, op_2, result);
43                    }
44                    else {
45                        printf("error: got %u - %u = %u\nexpected %u - %u =
```

main_master.cpp

```
%u\n", op_1, op_2, result, op_1, op_2, op_1 - op_2);
46         }
47     }
48     break;
49     case 2:
50         if ((op_1 * op_2) == result){
51             printf("%u * %u = %u\n", op_1, op_2, result);
52         }
53         else {
54             printf("error: got %u * %u = %u\nexpected %u * %u = %u\n",
55 op_1, op_2, result, op_1, op_2, op_1 * op_2);
56         }
57     }
58 }
59 }
60 int main(void)
61 {
62     srand(time(NULL));
63     xTaskCreate(vCalculate, "Calc", 1024, NULL, PRIORITY_LOW, NULL);
64     scheduler_add_task(new terminalTask(PRIORITY_HIGH));
65     scheduler_start();
66     return -1;
67 }
```