LIGHT REACTIVE DIMMER

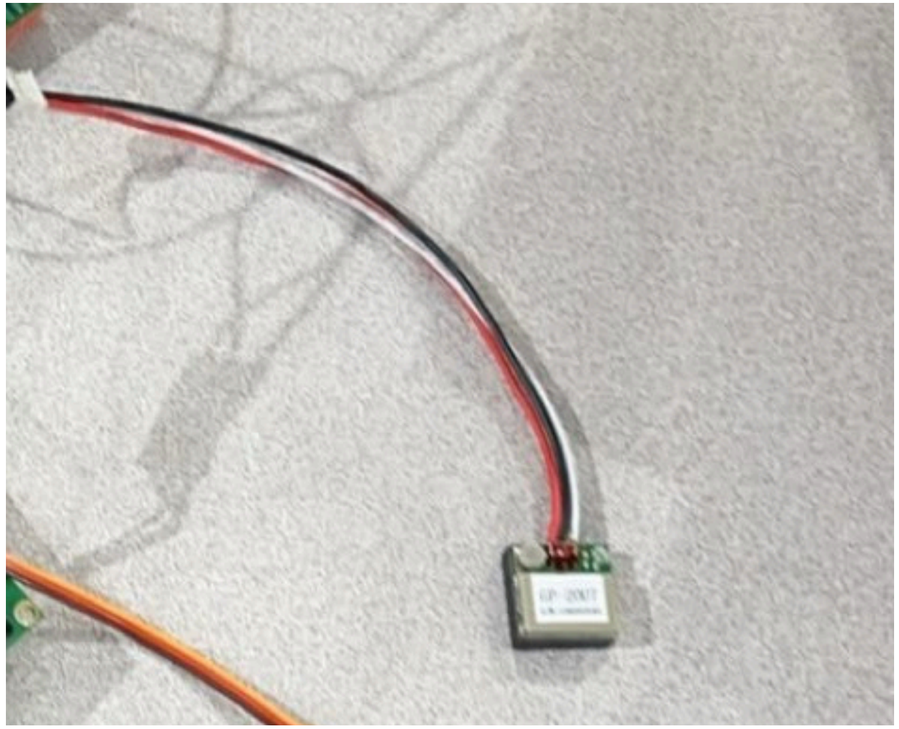
BY:HARNOOR SINGH, YURI BIRCH,

About the Idea

Applied technology building


SUNY BROOME

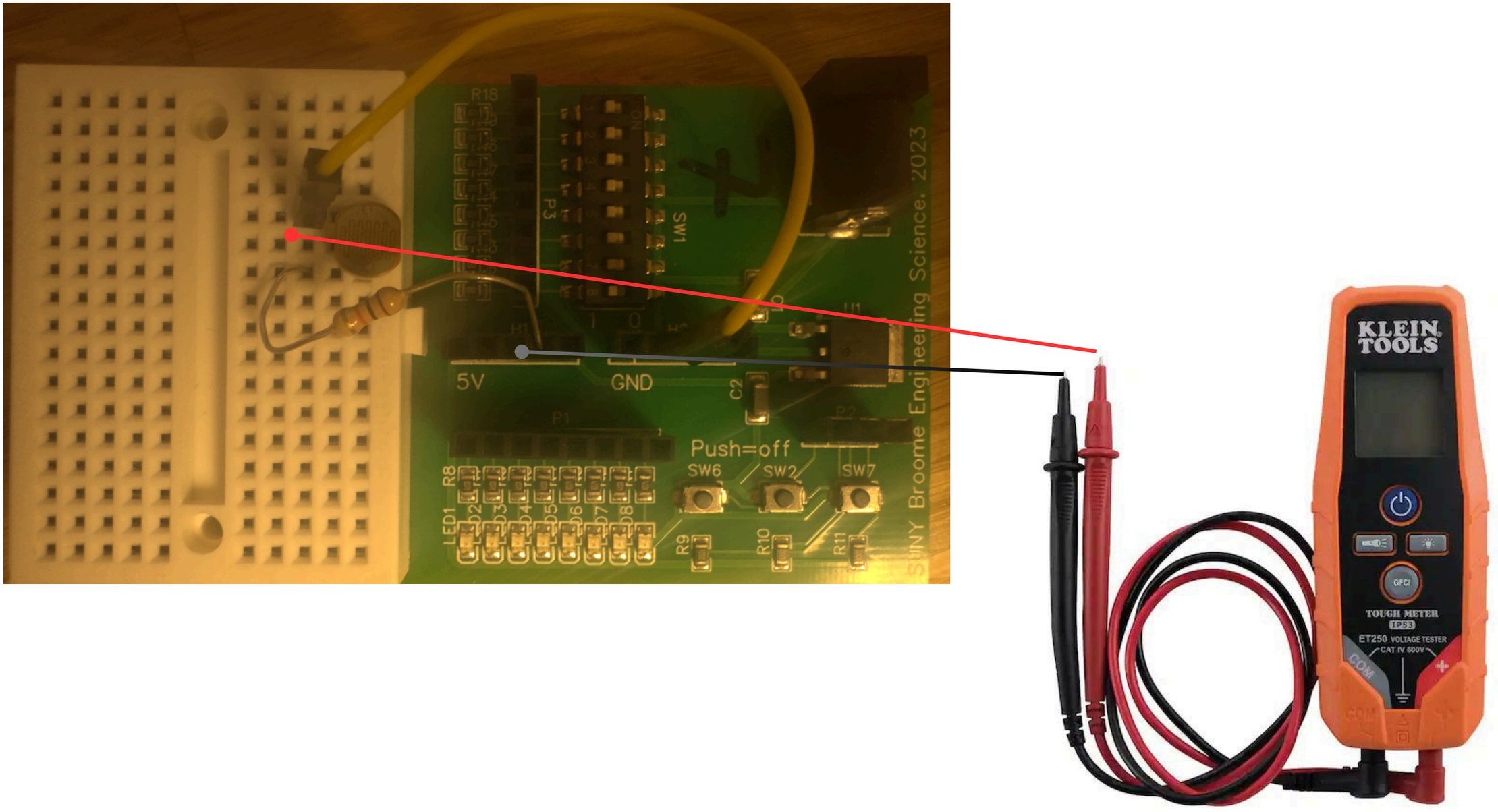
GPS




GPGGA,152244.00,4208
,W,1,04,2.70,281.8,M
.08828,N,0754.59967
,-34.4,M,,*65==$GPGS

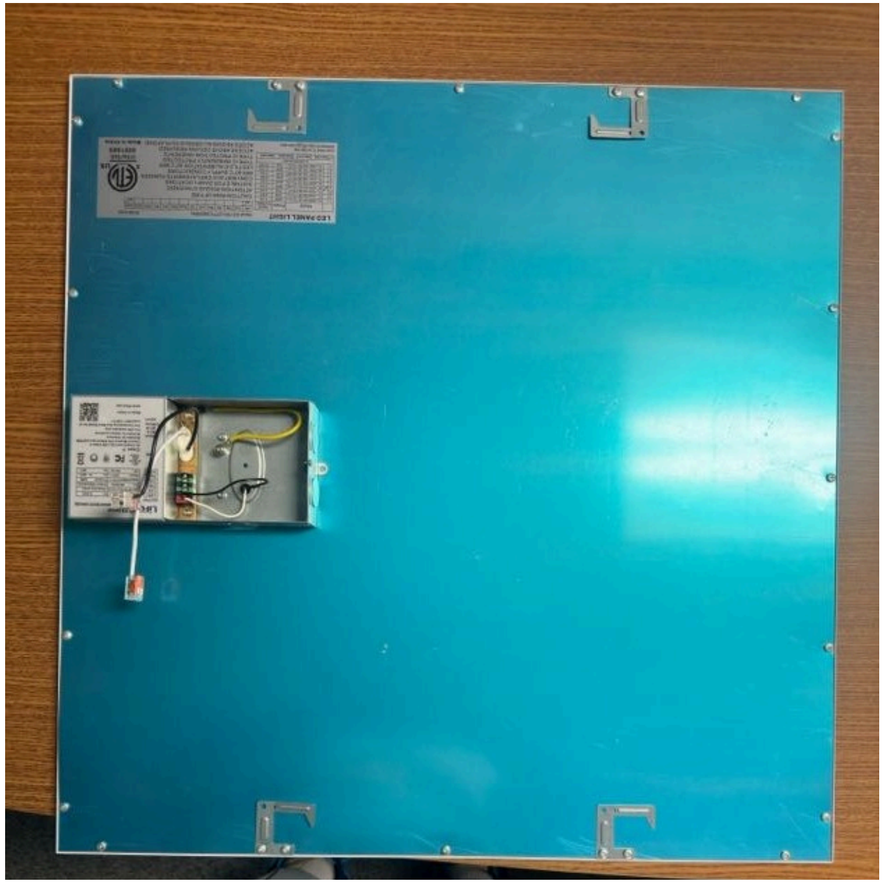
```
int main(void) {
    DDRB = 0b1110;



    class lcd_16x2_i2c lcd;


    freq_8mhz();
    UBRR0H = 0;
    UBRR0L = 103;
    UCSR0A |= (1 << U2X0);
    UCSR0B |= (1 << RXEN0);

    char a;
    do {
        a = serial_receive();
    } while (a != '$');

    const int n = 80;
    char c[100];
    for (int i = 0; i < n; i++) {
        c[i] = serial_receive();
    }

    int ho, mi;
    print_info(c, ho, mi);

```

The Light sensor


R18
SW1
P3
H1
5V
GND
C2
P1
P2
U1
Push=off
SW6
SW2
SW7
R8
LED1
R9
R10
R11
SUNY Broome Engineering Science, 2023
KLEIN TOOLS
GFCI
TOUGH METER
IP53
ET250 VOLTAGE TESTER
CAT IV 600V
COM


ADAPTER POWER
可 调 电 源
REGULATED
MODEL:LGY
CAUTION
RISK OF ELECTRIC
DO NOT OPEN
VOLTAGE
电压显示
CHANGE
电压调节
VOLTAGE

The Led light Panel

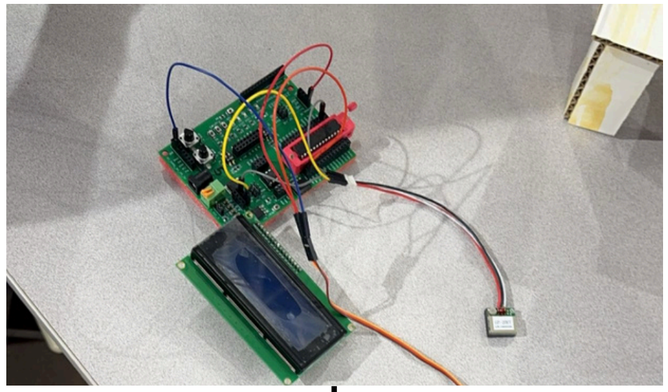


The code Behind it all

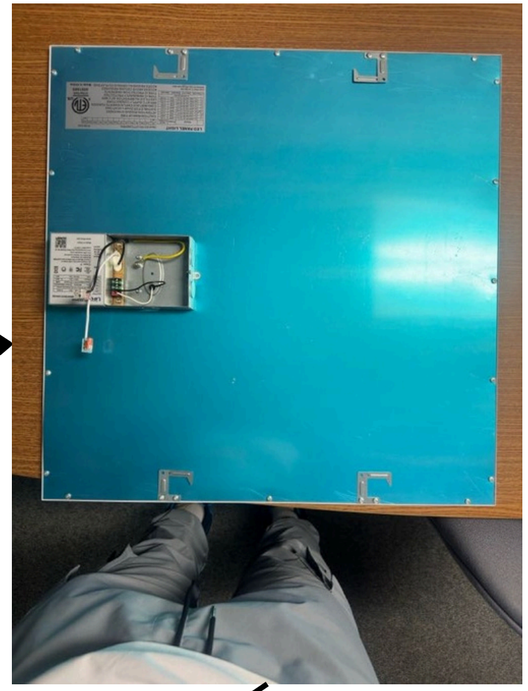
```cpp
#include <avr/io.h>
#include <util/delay.h>
#include <avr/interrupt.h>
#include "c:\avr\freq_328.h"
#include "c:\avr\i2c.h"
#include "c:\avr\lcd_16x2_i2c.h"

char serial_receive();
void print_info(char* data, int& h, int& m) ;

void cycle(int);



int main(void) {
    DDRB = 0b1110;



    class lcd_16x2_i2c lcd;


    freq_8mhz();
    UBRR0H = 0;
    UBRR0L = 103;
    UCSR0A |= (1 << U2X0);
    UCSR0B |= (1 << RXEN0);

    char a;
    do {
        a = serial_receive();
    } while (a != '$');

    const int n = 80;
    char c[100];
    for (int i = 0; i < n; i++) {
        c[i] = serial_receive();
    }

    int ho, mi;
    print_info(c, ho, mi);

```

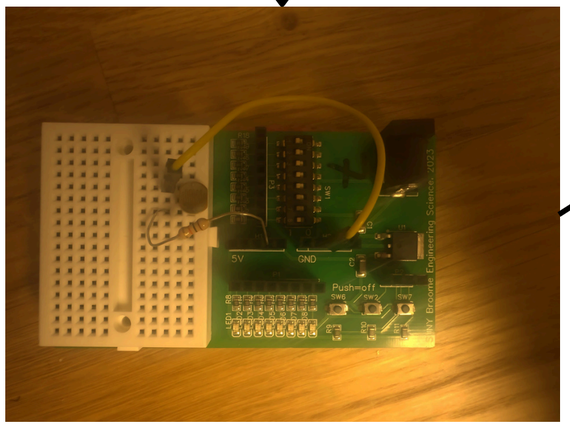
```cpp
char serial_receive() {
    while ((UCSR0A & (1 << RXC0)) == 0);
    int value = UDR0;
    return static_cast<char>(value);
}

void print_info(char* data, int& h, int& m)
{
  class lcd_16x2_i2c lcd;
    int comma_count = 0;
    int i = 0;
    while (comma_count < 1)
      { if (data[i] == ',')    comma_count++;
        i++;
      }

  h = (data[i  ]-0x30)*10 + (data[i+1]-0x30);
  m = (data[i+2]-0x30)*10 + (data[i+3]-0x30);

  lcd.dd(h);
  lcd.dd(m);
}
```

Assembly of the Device




Output
Input




5V
GND
Pushoff
SW6
SW2
SW7

The End

Any Questions