Revista Brasileira de Ensino de Física https://doi.org/10.1590/1806-9126-RBEF-2020-0502

"Construção de um luxímetro digital utilizando plataforma Arduino para uso em laboratórios didáticos"

Códigos Arduino

Biblioteca "functions.h"

```
void add_pulse() {

  // increase pulse count
  pulse_cnt++;
  //Serial.println(pulse_cnt);
  return;
}
```

Biblioteca "pins.h"

```
//These define the pin connections of the Arduino.
//They can be changed but only use digital in 2 or 3 for the Freq pin
#define TSL_FREQ_PIN 2 // output use digital pin2 for interrupt

// the digital pins that connect to the LEDs
#define redLEDpin 5
#define blueLEDpin 4

//the digital pin that connect to the button

#define BOTAO 3

#define BOTAO06 6
```

Biblioteca "variables.h"

```
float fD = 0.1; // Dark frequency
float Re = 2.3; // Responsivity

String start_by_serial;

bool start_by_button = false;
bool buttonState;

int erro=0;

int timing = 1000; // in milliseconds
volatile unsigned long pulse_cnt = 0;
//unsigned long pulse_cnt = 0;

 float Ee = 0;
```

```
 float Ee2 = 0;
 float EeT = 0;
 float a = 0;
 int leitura;
 int leitura6;
```

Programa “PhotonMeter.ino”

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include "pins.h"
#include "variables.h"
#include "functions.h"
LiquidCrystal_I2C lcd(0x27,20,4);  // set LCD address 0x27, 16 chars, 2 line display
void setup() {
Serial.begin(9600);
Wire.begin();
pinMode(redLEDpin, OUTPUT);
pinMode(blueLEDpin, OUTPUT);
pinMode(BOTAO, INPUT_PULLUP);
pinMode(BOTAO06, INPUT_PULLUP);
pinMode(TSL_FREQ_PIN, INPUT);
{
lcd.init();                      // initialize the lcd
lcd.backlight();
lcd.setCursor(0,0);
lcd.print("Luxime. iniciado");
Serial.println("Luximetro iniciado");
lcd.setCursor(0,1);
lcd.print("Aperte o botao");
Serial.println("Aguardando...");
Serial.println();
}
}
void loop() {
if (digitalRead(BOTAO)==LOW){
leitura=1;
}
if (digitalRead(BOTAO06)==LOW){
leitura6=1;
}
 if (leitura==1){
   leitura=0;
    digitalWrite(redLEDpin, HIGH);
    pulse_cnt=0;
    uint32_t millis1 = millis();
    attachInterrupt(0, add_pulse, RISING);
    delay(timing);
    detachInterrupt(0);
    uint32_t millis2 = millis();
    unsigned long finalcnt=pulse_cnt;
    // Flash blue LED at end of sampling interval.
    digitalWrite(blueLEDpin, HIGH);
    delay(500);
    digitalWrite(blueLEDpin, LOW);
    Ee = (finalcnt/(timing/1000.0) - fD)/1000.0/Re; // fO = fD + (Re)(Ee)
```

```
    EeT = Ee - Ee2;
 {
  lcd.init();                      // initialize the lcd
  lcd.backlight();
  lcd.setCursor(2,0);
  lcd.print("Luminosidade");
  lcd.setCursor(1,1);
  lcd.print(EeT, 4);
  lcd.setCursor(9,1);
  lcd.print("Lux");
}
 }    else {
    digitalWrite(redLEDpin, LOW);
  }
if (leitura6==1){
leitura6=0;
digitalWrite(redLEDpin, HIGH);
pulse_cnt=0;
uint32_t millis1 = millis();
attachInterrupt(0, add_pulse, RISING);
delay(timing);
detachInterrupt(0);
uint32_t millis2 = millis();
unsigned long finalcnt=pulse_cnt;
// Flash blue LED at end of sampling interval.
digitalWrite(blueLEDpin, HIGH);
delay(500);
digitalWrite(blueLEDpin, LOW);
Ee2 = (finalcnt/(timing/1000.0) - fD)/1000.0/Re; // fO = fD + (Re)(Ee)
    {
  lcd.init();                      // initialize the lcd
  lcd.backlight();
  lcd.setCursor(2,0);
  lcd.print("Lumi. Residual");
  lcd.setCursor(1,1);
  lcd.print(Ee2, 4);
  lcd.setCursor(9,1);
  lcd.print("Lux");
}
} else {
digitalWrite(redLEDpin, LOW);
}
}
```