

Annexe

1. Images

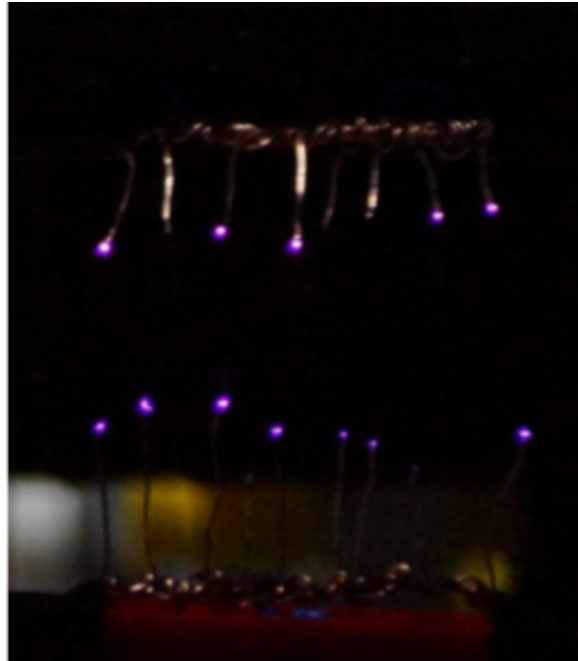


Figure 1: Photo de multiples décharges corona dans l'air ambiant

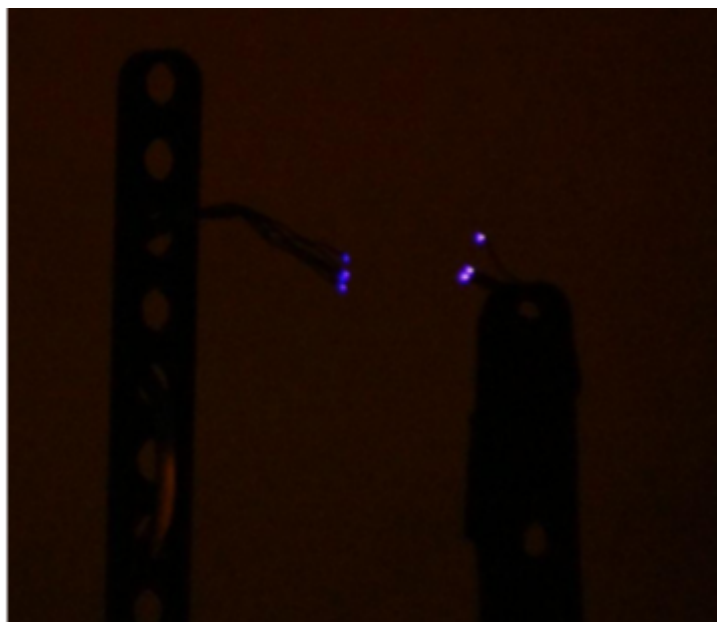


Figure 2: Photo de décharges corona dans l'air ambiant

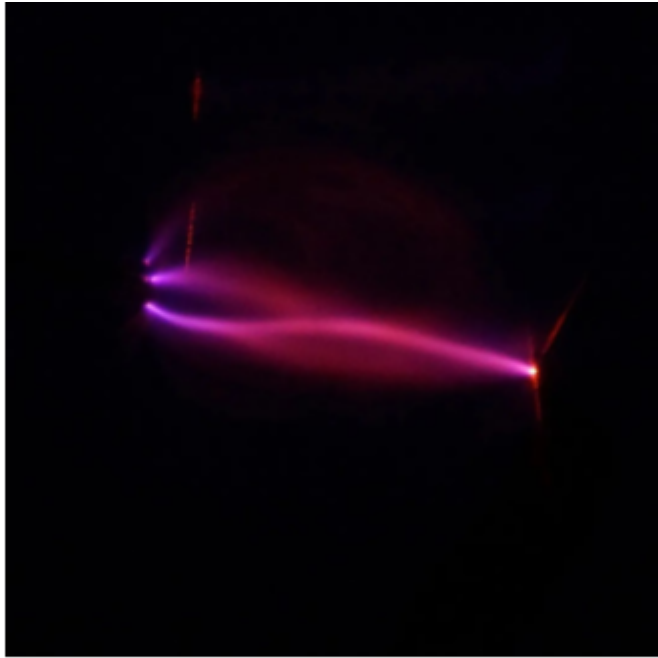


Figure 3: Photo d'une décharge corona avec une majorité de Néon en composition atmosphérique

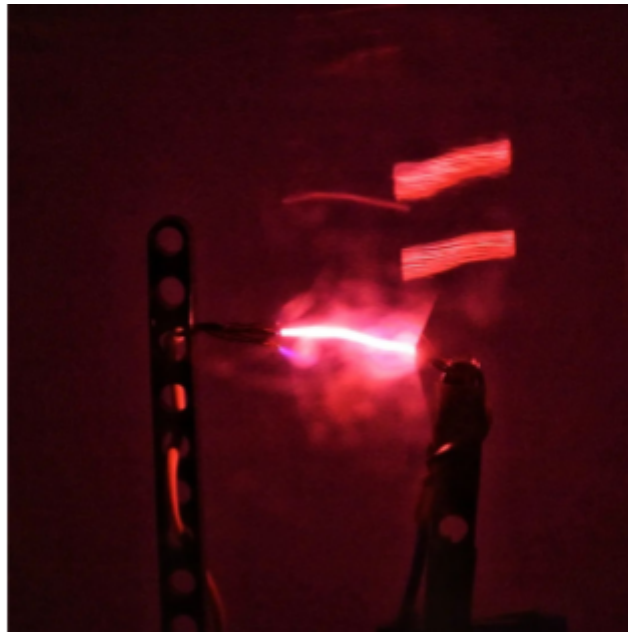


Figure 4: Photo d'un arc électrique avec une majorité de Néon en composition atmosphérique

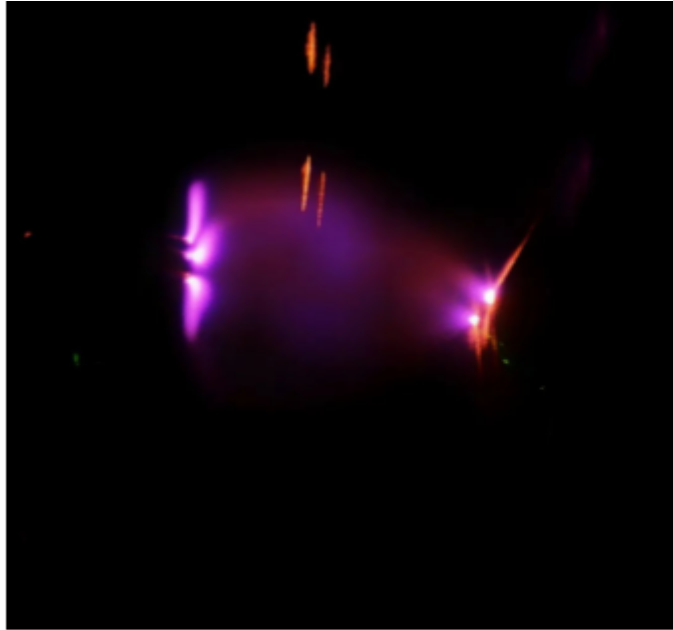


Figure 5: Photo d'une décharge corona avec une majorité de Néon en composition atmosphérique



Figure 6: Photo d'un arc électrique avec une majorité d'Argon en composition atmosphérique

2. Codes Arduino

2.1. Code servomoteur et télécommande

```
#include "IRremote.h"

#include <Servo.h>          // inclusion des librairies

int receiver = 11; // Signal Pin of IR receiver to Arduino Digital Pin 11

Servo myservo;

int pos = 88;           // angle du servo moteur qui commence à 88 degrés

int modifier = 1;      // var modifier : prends les valeurs de 1 à 9 et correspond à l'angle auquel va tourner le servo

int dir = 0;           // var dir : prends les valeurs -1 et 1 et correspond au sens de rotation du servo

IRrecv irrecv(receiver);

decode_results results;

void translateIR() // fonction qui actionne le moteur

{

  switch(results.value) {

    case 0xFFA25D: Serial.println("POWER"); break;

    case 0xFFE21D: Serial.println("FUNC/STOP"); break;

    case 0xFF629D: Serial.println("VOL+"); break;

    case 0xFF22DD: pos -= 15; break;

    case 0xFF02FD: pos = 88; break;

    case 0xFFC23D: pos += 15; break;

    case 0xFFE01F: dir = -1; break;

    case 0xFFA857: Serial.println("VOL-"); break;

    case 0xFF906F: dir = 1; break;

    case 0xFF9867: Serial.println("EQ"); break;

    case 0xFFB04F: Serial.println("ST/REPT"); break;

    case 0xFF6897: Serial.println("0"); break;

    case 0xFF30CF: modifier = 1; break;

    case 0xFF18E7: modifier = 2; break;

    case 0xFF7A85: modifier = 3; break;

    case 0xFF10EF: modifier = 4; break;

    case 0xFF38C7: modifier = 5; break;
```

```

case 0xFF5AA5: modif = 6; break;
case 0xFF42BD: modif = 7; break;
case 0xFF4AB5: modif = 8; break;
case 0xFF52AD: modif = 9; break;
case 0xFFFFFFFF: Serial.println(" REPEAT");break;
default:
    Serial.println(" other button "); // End Case
delay(500); // Délai pour ne pas que le moteur ait de problèmes
if (dir == 1) { pos += modif; }
if (dir == -1) { pos -= modif; }
// incrémentation ou décrementation de l'angle du servo moteur
myservo.write(pos);
delay(150); // delay pour laisser le temps au moteur de tourner
modif = 0; // réinitialisation de la variable modif
} //END translateIR

void setup() {
    Serial.begin(9600);
    Serial.println("IR Receiver Button Decode");
    irrecv.enableIRIn(); // Start the receiver
    Serial.begin(9600);
    myservo.attach(9); }

void loop() {
    if (irrecv.decode(&results)) // have we received an IR signal?
    {
        translateIR();
        irrecv.resume(); // receive the next value } // FIN

```

2.2. Code afficheur LCD et capteur d'humidité

```
#include <DHT.h>

#include <LiquidCrystal.h>

//LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

const int rs = 3, en = 4, d4 = 5, d5 = 6, d6 = 7, d7 = 8;

LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

#define DHTPIN 13

#define DHTTYPE DHT22

DHT dht(DHTPIN, DHTTYPE);

int chk;

float hum;

float temp;

void setup()

{

  dht.begin(); // entame la communication avec le capteur

  analogWrite(8,15); // permet la communication avec le lcd

  lcd.begin(16, 2); // précise le nombre de caractères utilisables du LCD (tous)

}
```

```
void loop()
{
  hum = dht.readHumidity(); // prend la valeur d'humidité du capteur
  temp= dht.readTemperature(); // prend la valeur de la température du capteur

  lcd.setCursor(0, 0); // place le curseur à la première ligne première colonne
  lcd.print("Humidity: "); // print "humidity" sur le lcd
  lcd.print(hum); // print la valeur de l'humidité

  lcd.setCursor(0, 1); // place le curseur à la seconde ligne première colonne
  lcd.print(" Temp: "); // print "Temp" sur le lcd
  lcd.print(temp); // print la valeur de la température

  delay(150); // delai de 150
```