

[code, online](#)

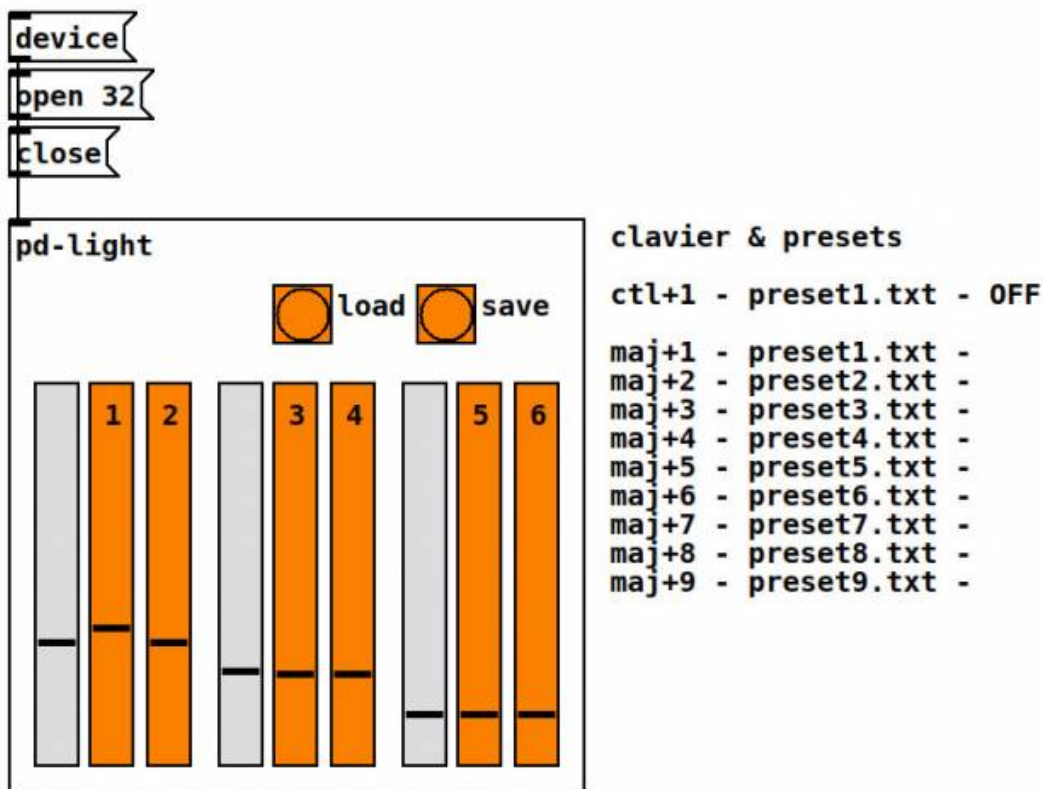
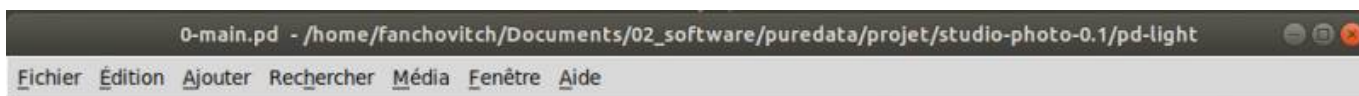
[Code] Pure Data - RGB strips driver

Codes .pd et .ino permettant de contrôler 2 RGB strips ou 6 monochrome strips via pure data et 2 "grove led strips RGB driver"

Codes : [studio-photo-0.1.zip](#)

Patch PureData

Libraries: comport - zexy - moocow



Code Arduino

```
#include "RGBdriver.h"
#define CLK 3//pins definitions for the driver
#define DIO 2
```

```
RGBdriver Driver(CLK, DIO);
String inputString = ""; // chaine de caractères pour contenir les données
boolean stringComplete = false; // pour savoir si la chaine est complète
String fct = "";
String arg = "";
String arg2 = "";
int index;
int led1;
int led2;
int led3;
int led4;
int led5;
int led6;
//-----
void setup()
{
  Serial.begin(9600); // port série
  pinMode(13, OUTPUT);
  pinMode(9, OUTPUT);
}
void loop()
{
  if (stringComplete) {
    //Serial.println(inputString);
    index = inputString.indexOf(' '); // on récupère la position du
séparateur (l'espace " ")
    fct = inputString.substring(0, index); // on coupe la chaine en deux :
la fonction et l'argument
    arg = inputString.substring(index, inputString.length());
    // deuxième découpage
    index = arg.lastIndexOf(' ');
    arg2 = arg.substring(index, arg.length());
    arg = arg.substring(0, index);
    // appel de ma fonction en transformant la chaine en nombre
    light(arg.toInt(), arg2.toInt());
    // vide la chaine
    inputString = "";
    stringComplete = false;
  }
}
// fonction personnalisable
void light(int strip, int brightness) {
  switch (strip) {
    case 2:
      led1 = (brightness);
      break;
    case 1:
      led2 = (brightness);
      break;
    case 3:
      led3 = (brightness);
```

```
        break;
    case 5:
        led4 = (brightness);
        break;
    case 4:
        led5 = (brightness);
        break;
    case 6:
        led6 = (brightness);
        break;
    }
Driver.begin(); // begin
Driver.SetColor(led1, led2, led3); //Red. first node data
Driver.SetColor(led4, led5, led6); //Blue. second node data
Driver.end();
}
void serialEvent() {
    while (Serial.available()) {
        char inChar = (char)Serial.read(); // récupérer le prochain octet (byte
ou char)
        inputString += inChar; // concaténation des octets
        if (inChar == '\n') { // caractère de fin pour notre chaîne
            stringComplete = true;
        }
    }
}
}
```

From:

<https://exsitu.xyz/> - **ExSitu**

Permanent link:

<https://exsitu.xyz/prod/recherches/code/pd-rgb-strip>

Last update: **17 12 2019**