



**UNIVERSITATEA
TEHNICĂ
DIN CLUJ-NAPOCA**

Afisator de GIF-uri

Proiectare cu Microprocesoare

Autor: Szekely Gergo

Grupa: 30237

FACULTATEA DE AUTOMATICA
SI CALCULATOARE

18 Ianuarie 2024

Cuprins

1	Descriere	2
2	Componentele folosite	2
3	Schema electrica a componentelor	4
4	Functionalitati si observatii	4
5	Bibliografie	6

1 Descriere

Am ales un proiect care preia un Gif dintr-un folder de pe un calculator dupa ce trece prin Python si se converteste intr-un bitmap. Dupa aceea, din Python, prin conexiune seriala, se trimite catre placa Arduino, iar aceasta, primind codul, afiseaza Gif-ul pe dispozitivul OLED.

2 Componentele folosite

Pentru simplitate m-am folosit de placuta Arduino Mega 2560 din cauza ca avea pini de alimentare la 3.3V si din aceasta cauza nu trebuia sa comand rezistente sa reduc de la 5V.

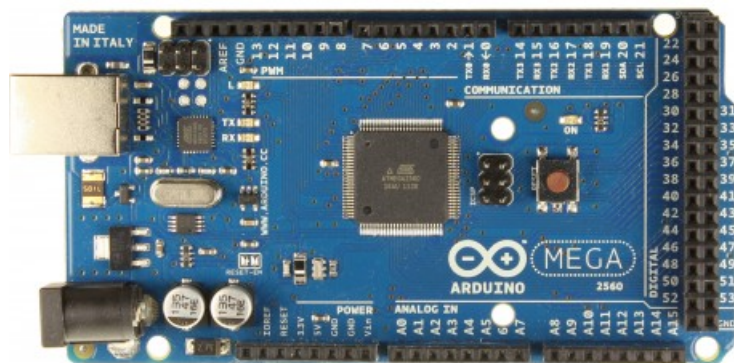


Figura 1: Arduino Mega 2560

SH110X OLED este un ecran cu 128x64 pixeli, care foloseste tehnologia OLED pentru imagini de calitate si contrast bun. Este mic si eficient, utilizand controlerul SS110X pentru a afisa rapid informatii grafice, fiind potrivit in diverse proiecte electronice pentru a prezenta date si grafice clar si eficient



Figura 2: SH110X 126x64

Breadboard-ul a fost folosit pentru a conecta dispozitivul de oled cu placuta Arduino Mega 2560.

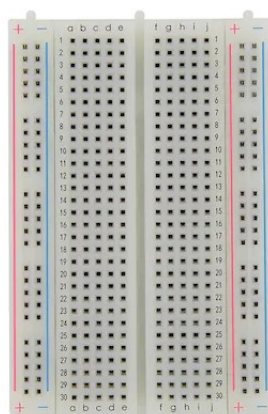


Figura 3: Breadboard

Cablurile folosite sunt doar de "tata-tata". Acestea au fost folosite pentru a conecta dispozitivul de ssd1306 oled la breadboard si de acolo la Arduino.



Figura 4: Cabluri "jumper"

3 Schema electrica a componentelor

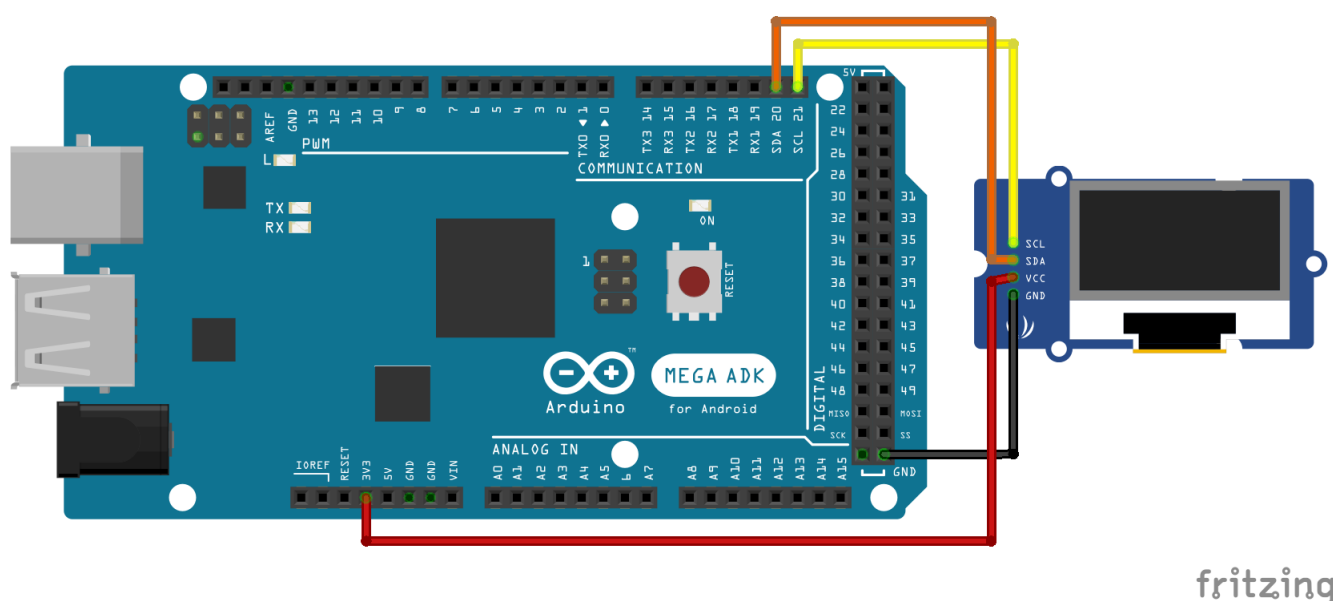


Figura 5: Schema electrica

4 Functionalitati si observatii

Intr-un folder punem gif-ul dorit. Dupa ce punem placuta la curent de la laptop si incarcam codul de Arduino care ne afiseaza imaginea sau gif-ul. Dar mai intai trece printr-un cod Python unde este convertit intr-un bitmap, dupa care este trimis prin conexiune seriala spre port-ul "COM12" prin care placa Arduino primeste informatia.

Dupa mai multe incercari si erori, am reusit sa afisez imaginea dorita, asa cum se poate vedea si in videoclipul realizat. Am experimentat cu mai multe coduri pentru dispozitivul meu OLED si am ales varianta care a functionat cel mai bine. Dupa ce am reusit sa scriu codul in Arduino,

m-am dedicat dezvoltarii codului in Python, unde am testat si incercat mai multe variante. Odata ce am observat ca imaginea bitmap este generata corect, am stabilit conexiunea intre cele doua coduri folosind comunicarea seriala pe portul "COM12". Dupa testarea proiectului, am avut bucuria de a constata ca am reusit sa afisez imaginea transformata in biti pe dispozitivul meu OLED.

```
1  #include <Wire.h>
2  #include <Adafruit_GFX.h>
3  #include <Adafruit_SH110X.h>
4
5  #define SCREEN_WIDTH 128
6  #define SCREEN_HEIGHT 64
7  #define OLED_RESET -1
8  Adafruit_SH1106G display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);
9
10 void setup()
11 {
12     Serial.begin(9600);
13     delay(250);
14     display.begin(0x3c, true);
15     display.clearDisplay();
16 }
17
18 void loop()
19 {
20     uint8_t picture[1024];
21     Serial.readBytes(picture, sizeof(picture));
22     display.drawBitmap(0, 0, picture, SCREEN_WIDTH, SCREEN_HEIGHT, 1);
23     display.display();
24
25 }
26
```

python cod:

```
1  import serial
2  import time
3  from PIL import Image
4
5  image_path = "D:\\Egyetem\\PMP\\PROJECT\\whale.jpg"
6  output_bitmap_path = "D:\\Egyetem\\PMP\\PROJECT\\picture_bitmap.bmp"
7
8  image = Image.open(image_path)
9  resized_image = image.resize((128, 64))
10 bitmap = resized_image.convert("1")
11 bitmap.save(output_bitmap_path, "BMP")
12
13 bitmap_bytes = bytearray(bitmap.tobytes())
14
15 ser = serial.Serial('COM12', 9600)
```

```
16 time.sleep(2)
17 ser.write(bitmap_bytes)
18 ser.close()
19
```

5 Bibliografie

Click pentru redirectionare catre pagini cu referinte.

[AdafruitSSD1306 Library](#)

[Conectiunea Serial in Python](#)