

Textul si imaginile din acest document sunt licentiate

Attribution-NonCommercial-NoDerivs
CC BY-NC-ND



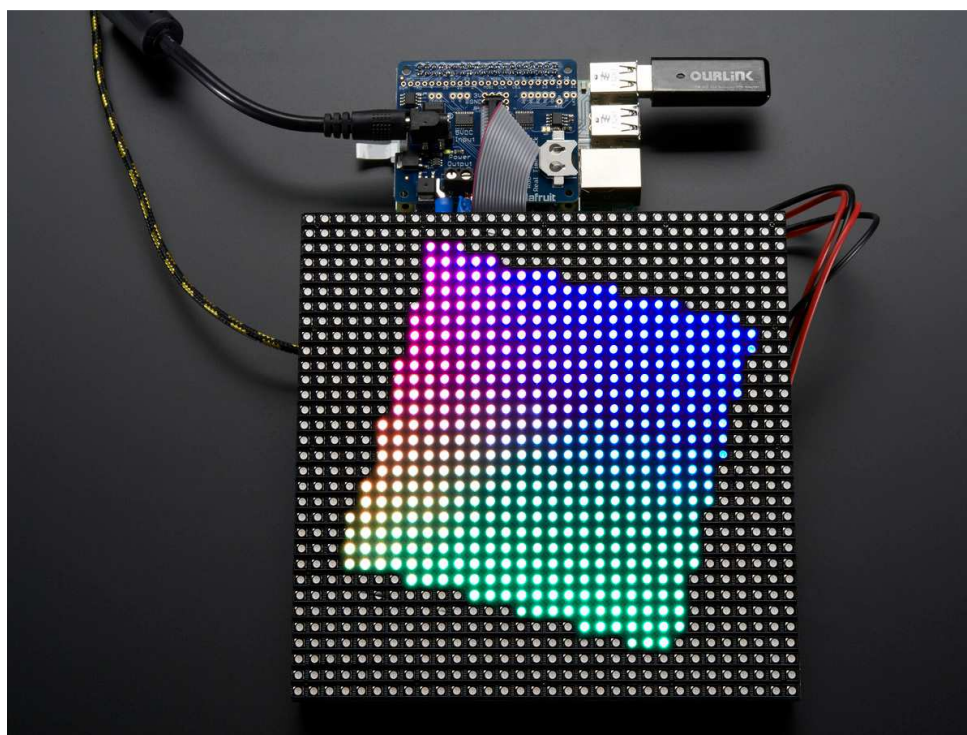
Codul sursa din acest document este licentiat

Public-Domain

Esti liber sa distribui acest document prin orice mijloace consideri (email, publicare pe website / blog, printare, sau orice alt mijloc), atat timp cat nu aduci nici un fel de modificari acestuia. Codul sursa din acest document poate fi utilizat in orice fel de scop, de natura comerciala sau nu, fara nici un fel de limitari.

Real Time Pi 3

Chiar dacă realizarea unui ceas electronic este o sarcină simplă care se poate realiza utilizând cam orice platformă programabilă (inclusiv o placă Arduino) utilizarea unui unui ecran de dimensiuni mari format din leduri RGB pentru afișare este o sarcină puțin mai dificilă. Dificultatea provine atât din necesarul de memorie cât și din necesarul de putere de prelucrare. Tocmai din aceste motive, în cadrul lecției de față, ne propunem să prezentăm implementarea unui ceas / calendar utilizând o platformă Raspberry Pi și o placă Adafruit RGB Matrix + Real Time Clock – placă ce reprezintă combinația ideală pentru proiectul nostru deoarece conține atât componenta de RTC cât și componenta de comandă pentru matrici RGB.



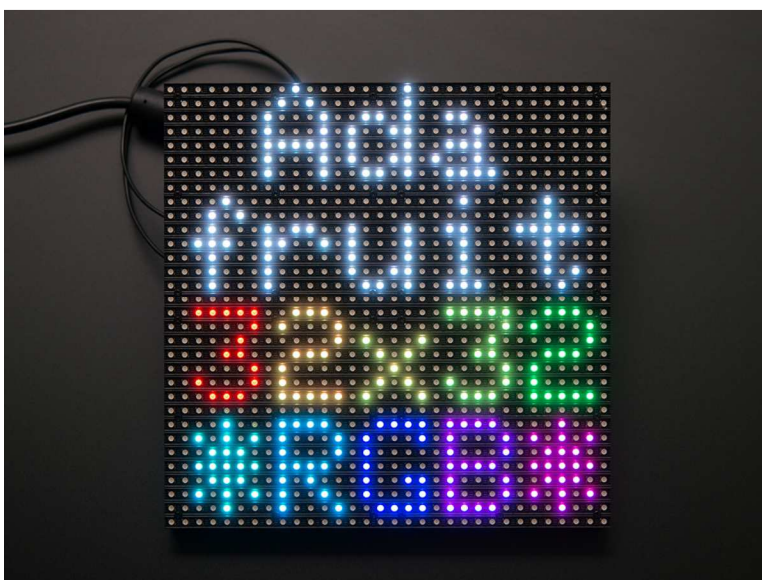
<https://www.robofun.ro/raspberry-pi-si-componente/adafruit-rgb-matrix-hat-rtc-for-raspberry-pi-mini-kit>

Pentru mai multe informații legate de punerea în funcțiune și utilizarea plăcii Adafruit RGB Matrix + Real Time Clock se poate consulta și:

Adafruit RGB Matrix + Real Time Clock HAT for Raspberry Pi

<https://learn.adafruit.com/adafruit-rgb-matrix-plus-real-time-clock-hat-for-raspberry-pi>

Atenție!!! Pentru buna funcționare a modulului RTC este nevoie de o baterie de 12mm (de exemplu CR1225) ce nu este inclusă în kit. Placa Adafruit RGB Matrix nu vine cu afișajul RGB inclus, se poate utiliza unul sau se pot insera mai multe module Adafruit Matrice LED RGB 32x32:



<https://www.robofun.ro/electronice/led/matrice-led-rgb-32x32>

Pentru mai multe informații despre inserierea și alimentarea corectă a acestor module LED RGB se poate consulta și următorul material:

32x16 and 32x32 RGB LED Matrix HUNDREDS of pixels of eye-blasting LED glory!

<https://learn.adafruit.com/32x16-32x32-rgb-led-matrix/>

Pentru implementarea funcționalității propuse vom efectua două operații: vom configura placa Raspberry Pi să utilizeze circuitul RTC DS1307 prezent pe placa Adafruit și vom scrie programul ce afișează data și ora pe afișajul matriceal RGB.

Configurarea plăcii Raspberry Pi pentru a utiliza circuitul RTC DS1307 este similară cu configurarea prezentată în prima parte a seriei de lecții Real Time Pi. Se va instala suportul software pentru comunicația I2C:

```
sudo apt-get install -y python-smbus
sudo apt-get install -y i2c-tools
```

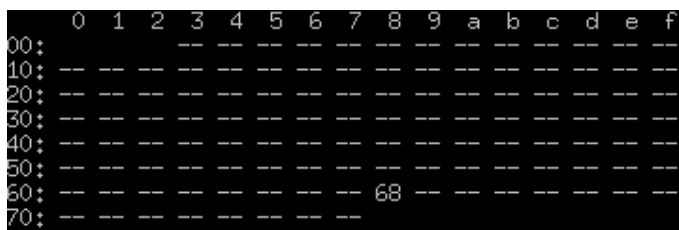
și se va activa comunicația I2C utilizând utilitarul *raspi-config*:

```
sudo raspi-config
```

Pentru a verifica conexiunea între placa de dezvoltare și componenta RTC se poate utiliza următoarea comandă:

```
sudo i2cdetect -y 1
```

ce va indica adresa I2C a modulului conectat: 0x68 pentru DS1307.



	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
00:				--	--	--	--	--	--	--	--	--	--	--	--	--
10:	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
20:	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
30:	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
40:	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
50:	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
60:	--	--	--	--	--	--	--	68	--	--	--	--	--	--	--	--
70:	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Dacă conexiunea între modulul RTC și placa de dezvoltare este în regulă putem configura fișierele de inițializare a sistemului de operare pentru a utiliza modulul RTC.

În fișierul */boot/config.txt* se adaugă următoarea linie:

```
dtoverlay=i2c-rtc,ds1307
```

Se va înlătura serviciul sistemului de operare ce emulează ceasul hardware:

```
sudo apt-get -y remove fake-hwclock
sudo update-rc.d -f fake-hwclock remove
```

în fișierul */lib/udev/hwclock-set* se vor comenta următoarele linii

```
#if [ -e /run/systemd/system ] ; then
```

```
# exit 0
#fi
```

și se va reporni sistemul:

```
sudo reboot
```

După repornire se poate verifica funcționarea corectă a modulului RTC și a mecanismelor specifice sistemului de operare prin următoarele comenzi:

- Prima inițializare a modulului RTC se va face prin intermediul comenzii:

```
sudo hwclock -D -r
```

- Sincronizarea timpului sistemului cu modulul RTC se va face prin comanda (scrierea datei sistem în memoria modulului RTC):

```
sudo hwclock -w
```

- Consultarea informațiilor din modulul RTC se va face prin comanda:

```
sudo hwclock -r
```

După setarea inițială a modulului RTC sistemul Raspberry Pi se va sincroniza automat la fiecare repornire cu informațiile stocate în memoria internă a modulului RTC – la fiecare repornire ora și data se vor inițializa cu ajutorul modulului RTC care păstrează informațiile și în lipsa alimentării cu energie electrică.

Pentru a implementa funcționalitatea de ceas (afișarea datei și orei pe ecranul LED RGB) se va pleca de la proiectul:

RGB LED Matrix and matrix hat + RTC From Adafruit

<https://www.youtube.com/watch?v=7iGx401DEs0>

Codul sursă a acestui proiect se poate descărca de la adresa:

<https://drive.google.com/open?id=0ByaorJ43ehItV3laTHdjenR2bms>

Acest proiect utilizează biblioteca BiblioPixel ce permite realizarea de animații pentru mai multe tipuri de afișaje:

<https://github.com/maniacallabs/bibliopixel/>

Utilizarea acestei biblioteci necesită instalarea următoarelor componente:

```
sudo pip install BiblioPixel
sudo pip install pillow
sudo pip install spidev
sudo pip install pyserial
```

După instalarea componentelor software și după copierea codului sursă a proiectului se va edita fișierul *clock.py* și se va putea observa codul necesar implementării funcționalității de ceas (funcția *ShowDateTime*) scris în limbajul Python:

```
global image
global draw
blink=0
for n in range(1,seconds*2):
    font = ImageFont.truetype('font/Generica Condensed
                               Std Bold.otf', 17)
    fonts = ImageFont.truetype('font/pixelmix.ttf', 8)
    if blink == 0:
        if mode == 0:
            for n in range(1, 9):
                draw.line((1, n, 30, n),fill=colors.Black)
                draw.text((4,1),datetime.now().strftime('%H'),
                           font=fonts, fill=Timecolor)
                draw.text((18,1), datetime.now().strftime('%M'),
                           font=fonts, fill=Timecolor)
```

<https://www.robofun.ro/forum/>

```

draw.text((16,1), ':', font=fonts, fill=Timecolor)
draw.text((3,11), datetime.now().strftime('%d/%m'),
          font=fonts, fill=Datecolor)
draw.text((6,21), datetime.now().strftime('%20%y'),
          font=fonts, fill=Yearcolor)

blink=1
elif mode == 1:
for n in range(1, 16):
draw.line((1, n, 30, n), fill=colors.Black)
draw.text((1,2), datetime.now().strftime('%H'),
          font=font, fill=Timecolor)
draw.text((17,2), datetime.now().strftime('%M'),
          font=font, fill=Timecolor)
draw.text((15,0), ':', font=font, fill=Timecolor)
draw.text((3,21), datetime.now().strftime('%d/%m'),
          font=fonts, fill=Datecolor)

blink=1
else:
if mode == 0:
for n in range(1, 9):
draw.line((1, n, 30, n), fill=colors.Black)
draw.text((4,1), datetime.now().strftime('%H'),
          font=fonts, fill=Timecolor)
draw.text((18,1), datetime.now().strftime('%M'),
          font=fonts, fill=Timecolor)
draw.text((3,11), datetime.now().strftime('%d/%m'),
          font=fonts, fill=Datecolor)
draw.text((6,21), datetime.now().strftime('%20%y'),
          font=fonts, fill=Yearcolor)

blink=0
elif mode == 1:
for n in range(1, 16):

```

```
draw.line((1, n, 30, n), fill=colors.Black)
draw.text((1,2), datetime.now().strftime('%H'),
          font=font, fill=Timecolor)
draw.text((17,2), datetime.now().strftime('%M'),
          font=font, fill=Timecolor)
draw.text((3,21), datetime.now().strftime('%d/%m'),
          font=fonts, fill=Datecolor)

blink=0
matrix.SetImage(image.im.id, 0, 0)
time.sleep(0.5)
```

Bineînțeles, pe lângă funcționalitatea de ceas / calendar, ansamblul format din platforma Raspberry Pi și placa Adafruit Matrice LED RGB permite implementarea de funcționalități diverse (inclusiv proiectul dat ca exemplu adaugă funcționalitatea de stație meteo). Se pot explora diverse dezvoltări ulterioare parcurgând și următoarele proiecte:

RGB Matrix In Architectural Model

<https://blog.adafruit.com/2015/04/28/rgb-matrix-in-architectural-model-arttuesday/>

Raspberry Pi RGB LED Matrix Webapp

Control RGB LED panels from a Web application using Raspberry Pi.

<https://learn.adafruit.com/raspberry-pi-rgb-led-matrix-webapp>

DIY LED Video Wall

Tile together multiple RGB LED panels to create a giant LED wall!

<https://learn.adafruit.com/adafruit-diy-led-video-wall/>