

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.

# Aurora *geomagnetic lights*

În cadrul lecției de față ne propunem să realizăm un sistem de lumini care să redea prin culoare valoarea câmpului magnetic al pământului – de aici și denumirea lecției și paralela cu luminile aurorei boreale.



Pentru implementare vom utiliza o placă de dezvoltare Polulu A-Star 32U4 Micro bazată pe microcontrolerul ATmega32U4 (la fel ca și Arduino Leonardo) și care are dimensiuni mult mai mici decât oricare altă placă similară (are o dimensiune de două ori mai mică decât Arduino Micro). Bineînțeles în cadrul proiectului se poate substitui placa de dezvoltare cu o placă echivalentă (Arduino Leonardo sau Arduino Micro) dar dimensiunea sistemului va fi ... mai mare (și prețul un pic ... mai mare).



<https://www.robofun.ro/platforme/avr/a-star-32u4-micro>

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

Pentru măsurarea câmpului magnetic al pământului vom utiliza senzorul digital I2C HMC5883L – magnetometru pe trei axe:



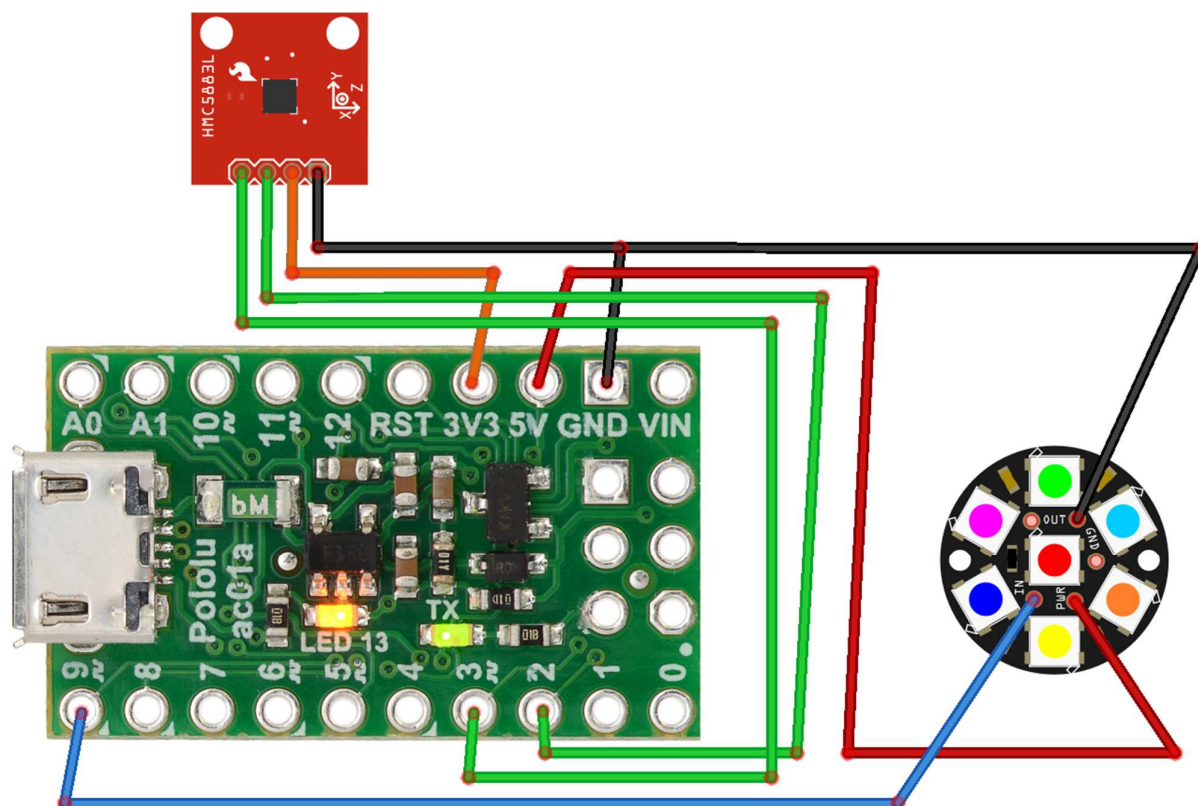
[https://www.robofun.ro/senzori/magnetic/magnetometru\\_3axe\\_HMC5883L](https://www.robofun.ro/senzori/magnetic/magnetometru_3axe_HMC5883L)

Pentru implementarea sistemului de lumini ne vom păstra într-o clasă de dimensiuni reduse și vom utiliza un ansamblu de 7 leduri RGB WS2812 5050 și anume Neopixel Jewel:



<https://www.robofun.ro/electronice/led/neopixel-jewel-7-x-ws2812-5050-rgb-led-with-integrated-drivers>

Schema de interconectare între placa de dezvoltare și celelalte două componente ale sistemului este:



Senzorul HMC5883L se va alimenta la 3.3V și va utiliza pinii 2 (SDA) și 3 (SCL) ai plăcii de dezvoltare. Neopixel Jewel se va alimenta la 5V și va utiliza pinul 9 al plăcii de dezvoltare (conectat la pinul IN).

Dimensiunea redusă a sistemului rezultat ne va conduce imediat cu gândul la o implemetare portabilă (o broșă sau un glob pentru pomul de crăciun) dar pentru a face acest lucru trebuie avute în vedere două aspecte extrem de importante:

- Cele 7 leduri 5050 degajă destul de multă căldură dacă sunt aprinse la maxim pentru o perioadă lungă de timp – trebuie lăsat spațiu suficient pentru răcire;
- Chiar dacă placa de dezvoltare acceptă pe pinul VIN tensiuni între 5.5V și 15V regulatorul intern poate debita un curent maxim de 100mA – nu poate alimenta Neopixel Jewel prin intermediul regulatorului intern al plăcii de dezvoltare – se recomandă utilizarea unei surse externe de 5V de minim 500mA.

Programul sistemului va utiliza următoarele biblioteci externe mediului Arduino IDE:

- Adafruit Unified Sensor Driver și Adafruit HMC5883L Driver pentru citirea magnetometrului;

[https://github.com/adafruit/Adafruit\\_Sensor](https://github.com/adafruit/Adafruit_Sensor)

[https://github.com/adafruit/Adafruit\\_HMC5883\\_Unified](https://github.com/adafruit/Adafruit_HMC5883_Unified)

- FastLed pentru comanda Neopixel Jewel.

<https://github.com/FastLED/FastLED>

În cadrul lecției vă propunem două exemple de program care să realizeze corelarea între citirile oferite de senzorul HMC5883L și comanda Neopixel Jewel dar sperăm ca acestea să reprezinte doar un punct de plecare în explorarea unor variante personale.

Primul exemplu de program va aprinde simultan toate cele 7 leduri ale Neopixel Jewel iar valorile câmpului magnetic al pământului pe cele trei axe vor influența cele trei componente de culoare (RGB – red/green/blue) ale tuturor ledurilor. Astfel se va realiza o corelație între valorile câmpului magnetic pe axele X, Y și Z și valorile R, G, B de comandă. Chiar dacă senzorul este capabil să raporteze valori între -800uT și 800uT ale câmpului magnetic corelarea se va face doar pe intervalul -60uT și 60uT – interval normal de achiziție.

```
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_HMC5883_U.h>
Adafruit_HMC5883_Unified mag =
    Adafruit_HMC5883_Unified(12345);

#include "FastLED.h"
#define NUM_LEDS 7
#define DATA_PIN 9
CRGB leds[NUM_LEDS];

void setup() {
    if(!mag.begin()) { while(1); }
```

```

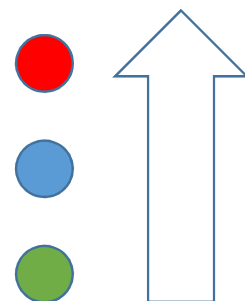
FastLED.addLeds<NEOPIXEL, DATA_PIN>(leds, NUM_LEDS);
}

void loop() {
    sensors_event_t event;
    mag.getEvent(&event);
    int r,g,b;
    r = map(event.magnetic.x, -60, 60, 0, 255);
    g = map(event.magnetic.y, -60, 60, 0, 255);
    b = map(event.magnetic.z, -60, 60, 0, 255);
    for (int i=0; i<NUM_LEDS; i++) {
        leds[i].setRGB( r, g, b);
        FastLED.show();
        delay(10);
    }
    FastLED.show();
    delay(1000);
}

```



Cel de al doilea exemplu de program va calcula unghiul față direcția nord (cap compas) și va indica direcția nord (la fel ca și un instrument de tip busolă). Pe Neopixel Jewel se vor aprinde doar trei leduri: cel din centru ce va indica intensitatea câmpului magnetic pe axa Z (neutilizată în calculul cap compas) pe culoarea albastră și alte două leduri diametral opuse vor da direcția - unul roșu (ce va indica nordul) și unul verde ce va indica sudul. Bineînțeles, corectitudinea geomagnetică a indicatorului rezultat constă și în alinierea corectă geometrică între senzorul magnetic și Neopixel Jewel.



```

#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_HMC5883_U.h>
Adafruit_HMC5883_Unified mag =
    Adafruit_HMC5883_Unified(12345);

```

```

#include "FastLED.h"
#define NUM_LEDS 7
#define DATA_PIN 9
CRGB leds[NUM_LEDS];

void setup() {
    if(!mag.begin()) { while(1); }
    FastLED.addLeds<NEOPIXEL, DATA_PIN>(leds, NUM_LEDS);
}

void loop() {
    sensors_event_t event;
    mag.getEvent(&event);

    float heading =
        atan2(event.magnetic.y, event.magnetic.x);
    float declinationAngle = 0.22;
    heading += declinationAngle;
    if(heading < 0) heading += 2*PI;
    if(heading > 2*PI) heading -= 2*PI;
    float headingDegrees = heading * 180/M_PI;

    int l1, l2;

    if ((headingDegrees>=0)&&(headingDegrees<60))
        { l1 = 1; l2 = 4; }
    if ((headingDegrees>=60)&&(headingDegrees<120))
        { l1 = 2; l2 = 5; }
    if ((headingDegrees>=120)&&(headingDegrees<180))
        { l1 = 3; l2 = 6; }
    if ((headingDegrees>=-60)&&(headingDegrees<0))

```

```

        { l1 = 4; l2 = 1; }
    if ((headingDegrees>=-120)&&(headingDegrees<=-60))
        { l1 = 5; l2 = 2; }
    if ((headingDegrees>=-180)&&(headingDegrees<=-120))
        { l1 = 6; l2 = 3; }

    int r,g,b;
    b = map(event.magnetic.z, -60, 60, 0, 255);

    for (int i=0; i<NUM_LEDS; i++) {
        leds[i] = CRGB::Black;
        FastLED.show();
        delay(10);
    }
    FastLED.show();

    leds[0].setRGB( 0, 0, b);
    leds[l1].setRGB( 255, 0, 0);
    leds[l2].setRGB( 0, 255, 0);
    FastLED.show();

    delay(1000);
}

```

Cele două programe au fost dezvoltate și testate cu Arduino IDE 1.6.12, Pololu A-Star 32U4 Drivers commit b0fc0c0, FastLED 3.1.3, Adafruit HMC5883 Unified 1.0.0 și Adafruit Unified Sensor 1.0.2.



Pentru mai multe surse de inspirație puteți trece în revistă și următoarele proiecte:

Neopixel Cosmic Turtle Necklace

<https://learn.adafruit.com/neopixel-led-magnetic-pendant-necklace>

Jewel Hair Stick

<https://learn.adafruit.com/jewel-hair-stick>

FLORA NeoGeo Watch

<https://learn.adafruit.com/flora-geo-watch>