

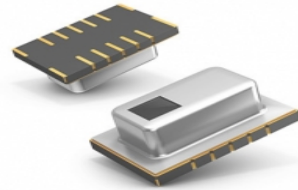


A Panasonic GRID-EYE szenzor 2. generáció használata – 3. rész

Cikkorozatunk első részében bemutattuk a Panasonic GridEye termoelem mátrix szenzort és annak felhasználási lehetőségeit, valamint írtunk arról, hogy hogyan lehet ezt a szenzort egy önállóan működő mini-hőkameraként felhasználni. A cikksorozat második részében a korábban megismert kiértékelő kit Arduino pajzsként való felhasználásával szimuláltunk egy gesztusvezérelt elektronikai rendszert (jelenlét érzékelésen alapuló vezérlés), és egy PC alapú mozgáskövető világításkapcsolási megoldást is bemutattunk. A sorozat harmadik, egyben befejező részében a GridEye szenzorral megvalósítunk egy önállóan futó Arduino Due alkalmazást, mely a mini hőkamera képét egy 8X8-as RGB LED mátrixon jeleníti meg, miközben a céltárgy mozgását egy szervó motorral igyekszünk követni.

A GRID-EYE szenzor

Mint azt az olvasó már tudja, a Panasonic GRID-EYE eszköze egy 8X8-as MEMS technológiára épülő hőelem mátrix, azaz 64 különálló szenzorral képes abszolút hőmérsékletet detektálni az objektum által kibocsátott infravörös sugárzás érzékelésével.



A Grid-EYE képes a hőmérséklet és a hőmérsékleti gradiens észlelésére és egyszerű, kifelbontású (8X8=64 pixeles) hőkép felvételére is.

A végtermék fejlesztés megkönnyítésére készített, a szenzort és soros kommunikációs interfészt tartalmazó kártya kialakítása alkalmassá teszi azt ARDUINO DUE miniszámítógéphez illeszkedő „shield” -ként való használatra.

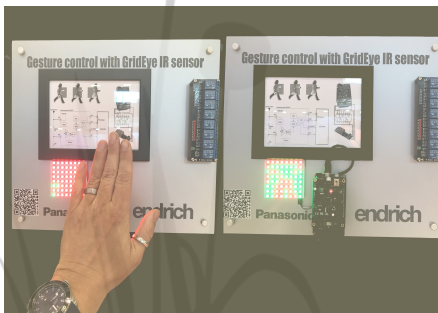
Fontos felhívunk a figyelmet arra, hogy a GridEye kiértékelő kit 3.3V-os kártya, így csak az Arduino DUE számítógépekkel működik. Az UNO-hoz már csak fizikai kialakítása miatt sem alkalmazható, a MEGA verziók, bár rendelkeznek a megfelelő csatlakozókkal melyek látszólag illeszkednek is a kártya

fizikai csatlakozásaihoz, azok 5V-os feszültség szintje miatt nem kompatibilisek a 3.3V-os GridEye Evaluation kit-el.



A feladat

A megvalósítandó szimulációs feladat során a GRID-EYE előtt mozgó objektum, jelen esetben az emberi kéz és annak közvetlen környezetének hőmérsékleti adatait az Arduino számítógéppel az I2C buszon keresztül a szenorból kiolvassuk. Szoftveres feldolgozás során kiemeljük a kéz hőképét, majd ezt egy a kereskedelemben kapható WS2812B meghajtóval rendelkező 8X8-as RGB LED mátrixon jelenítjük meg.



A lapon található egy termisztor is, mely a NYÁK környezeti hőmérsékletét méri

természetesen ennek adatára is szükség van. A kéz érzékelését az egyszerűség kedvéért úgy végezzük, hogy azon pixelek hőmérsékletadatait, melyek a szenzor környezeti hőmérsékleténél jelentősen nagyobb értékkel rendelkeznek piros, melyek pedig ezzel egyezők zöld színnel jelenítjük meg az RGB LED mátrixon. Az így kiemelt „piros folt” mozgását egy szervó motorral követjük.

Az Arduino DUE számítógép

Az Arduino Due az Atmel SAM3X8E ARM Cortex-M3 CPU-ra épülő mikrokontroller lap. Ez az Arduino család első 32-bit ARM magra épülő tagja. 54 digitális I/O porttal rendelkezik, melyek közül 12 PWM kimenetként is használható. Meghajthatunk vele például egy szervomotort, mely segítségével követhetjük a hőforrás mozgását. Mint említettük a korábbi verzióktól eltérően a DUE 3.3V-os kártya, így a GPIO portok maximális jelszintje is csak 3.3V. Emiatt a szervó motor meghajtásához kiviteltől függően esetleg gondoskodni kell a feszültségillesztésről (3.3V-ról 5V-ra), például a sorozat előző részében kifejtett módon, az SN74HC125N vonalmeghajtó alkalmazásával.

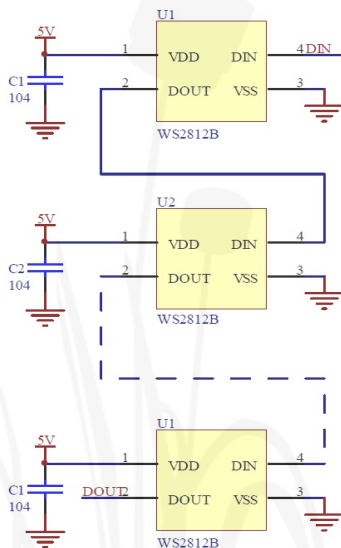
A WS2812B alapú RGB LED mátrix

A WS2812B 5050 RGB LED diódákból készített 8X8-as mátrix ideális kijelző a

GridEye szenzor számára, hiszen egy az egyben hozzárendelhetők a LED-ek a szenzor pixeleihez.

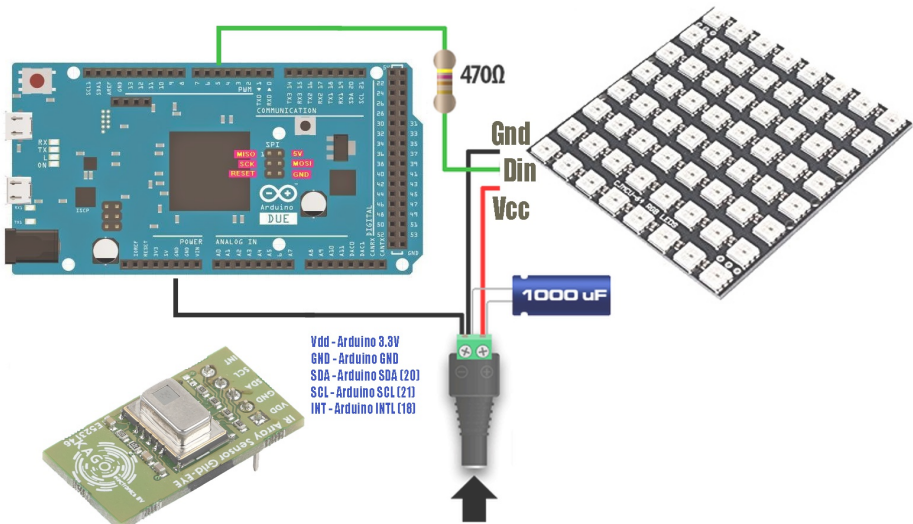


A WS2812B tartalmazza a címezhető LED drivert és az R,G,B chipeket is. Az egyes fénypontok összefűzhető LED szalaggá, vagy az általunk használt eszközben egy 64 potból álló fénymezővé. A panelen nem találunk előtétellenállásokat sem, mivel a WS2812B a fizikai chipeket impulzusszélesség modulált áramgenerátoros táplálással hajtja meg.



Az egyes LED-ek címzéséhez egy soros bit-stream küldése szükséges a lánc első WS2812B elemére. Az adatfolyam 3 byte-os RGB intenzitás értékeket tartalmaz a lánc minden meghajtója számára, így esetünkben az ARDUINO DUE $3 \times 8 \times 64 = 1536$ bitnyi adatot kell, hogy küldjön a 64 LED számára másodpercenként tízszer, mivel ez a GridEye szenzor mintavételezési frekvenciája (10 Hz). A lánc minden WS2812B eszköze továbbadja a Dout kimeneten keresztül az adatokat a következő láncem Din bemenetére, így minden LED megkapja a saját 24 bites színinformációját. Miután az adatok minden WS2812B-n rendelkezésre állnak, az Arduinonak szinkronizáló jelet kell kiküldeni, ezzel minden láncem egyidejűleg jeleníti meg a számára előírt színt. Az egyes színeket így a mért hőmérsékletértékeknek feleltethetjük meg. Egyszerűbb megoldás lehet, ha egy adott küszöbérték alatt zöld, afelett piros színnel érzékeltethetjük a kéz jelenlétét, mint ahogy a példánkban mi is tettük. Mivel az egyes LED-ek árama kb 3×20 mA, a teljes modulra vetítve az áramfelvétel kb 3.8A-re tehető maximális kivezérlésnél, tehát mindenképpen erős külső tápegységre van szükség.

A LED matrix meghajtására természetesen gyakorlott programozóként magunk is vállalkozhatunk, de erre nincsen szükség, mert az ARDUINO közösség



számára elérhetők olyan könyvtárak, mint például a Fastled Animation Library (<http://fastled.io/>), mely egyszerű interfészt biztosít nem csak a WS2812B, de más hasonló elven

működő modulok számára is (Neopixel, WS2801, WS2811, LPD8806, TM1809).

Az alábbi kódrészlet megmutatja milyen egyszerű a 64 egymáshoz fűzött LED



```
#include<FastLED.h>
#define NUM_LEDS 64
#define DATA_PIN 5

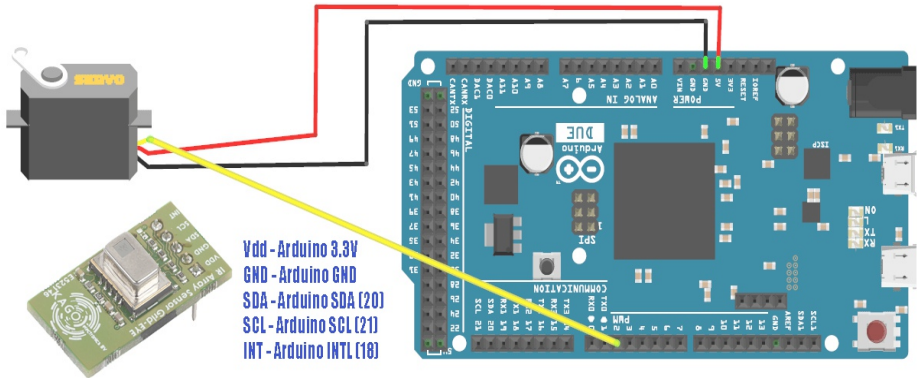
/* LED mátrix inicializálása*/
FastLED.addLeds<WS2812B, DATA_PIN, RGB>(leds, NUM_LEDS);

/* a 64 pixel színeinek beállítása tetszőleges értékekkel
/* Piros : redVALUE[i] Zöld:greenVALUE[i] Kék:blueVALUE[i]

for( int i = 0; i <= 63; i++ ) {
.
.
leds[i].setRGB( (int) ( redVALUE[i],greenVALUE[i],blueVALUE[i] );
.
.
}

FastLED.show();|
```

programozása az ARDUINO DUE egyik digitális kimenetén keresztül. A színek mért hőmérsékleti adatokhoz való igazítása a cikksorozat előző részében leírtak és némi fantázia segítségével könnyen elvégezhető. Természetesen a szerző minden szükséges segítséget megad az érdeklődők számára.



Szervó motor illesztése

A WS2812B 5050 RGB LED márix az előzőekben tárgyaltak alapján kiválóan alkalmazható a GridEye szenzorral mért hőmérsékleti adatok kijelzésére, azonban gesztusvezérlési alkalmazásokban a kijelzés nem elegendő, valamilyen vezérlésre és beavatkozásra is szükség van. Ennek modellezésére az ARDUINO számítógéphez egy szervó motort illesztünk és a mozgó objektum (kéz) hőképeinek követésére utasítjuk.

A DC szervó kivitelről (áramfelvételtől) függően vagy külső tápforrásról, vagy az Arduino lapról közvetlenül, esetleg feszültségillesztésen (3.3V → 5V)

keresztül táplálható, a pozicionálás pedig az Arduino Due egyik PWM (impulzusszélesség modulált) kimenetéről oldható meg. Szerencsére a bonyolult programozási algoritmusok megírása itt is elkerülhető, mindössze az ingyenesen elérhető SERVO könyvtárra van szükség.

Az következő oldalon lévő programrészlet bemutatja, hogy miként tudjuk a szervómotort 5 fokként pozicionálni a „hotspot” GridEye szenzoron megjelenő képeknek jobbra és balra mozgásának követésére.

Természetesen a fent bemutatott példák nem teljes programok, mindösszesen csak arra hivatottak, hogy megmutassuk milyen egyszerű is egy ilyen rendszer elkészítése.

Amennyiben az olvasó komolyabban érdeklődik a téma iránt, az Endrich budapesti irodájában bármikor lehetőség van személyes konzultációra és a működő modellek is megtekinthetők.



```
#include <Servo.h>
Servo myservo;          // Szervó objektum létrehozása

.
.
.

void setup() {
  myservo.attach(2);    // PWM2 a vezérlőport
  myservo.write(0);     // szervó alaphelyzet
}

.
.
.

// a "hőkép" jobbra mozog,
// a szervó 5 fokos lépéssel követi

pos = pos+5;
myservo.write(pos);
delay(100);

// a "hőkép" balra mozog,
// a szervó 5 fokos lépéssel követi

pos = pos-5;
myservo.write(pos);
delay(100);
```