

Mikrovezérlő Programozás

Tartalom

Az ötlet rövid leírása:	1
Hozzávalók és költségvetés	1
Működési elv	2
Kapcsolási rajz	3
Kód példa	3
Fejlesztési lehetőségek	4

Név: Sümegi Bence

Az ötlet rövid leírása:

Ebben a projektben egy LED fényerejét szabályozzuk az Arduino PWM (Pulse Width Modulation) technikájának segítségével. A PWM lényege, hogy a LED-hez kapcsolt digitális kimeneten nagyon gyorsan váltogatjuk a jelet be- és kikapcsolt állapotok között, és a világítás erősségét az határozza meg, hogy a bekapcsolt idő hány százalékát teszi ki az egy teljes ciklusnak. A LED így nem villog, hanem halványabban vagy erősebben világít, attól függően, hogy mennyi ideig van bekapcsolva a ciklus alatt.

Hozzávalók és költségvetés

Alkatrészlista költségvetéssel:

- DHT11 érzékelő - A hőmérséklet és páratartalom mérésére.
- Mikrokontroller (pl. Arduino, ESP32, Raspberry Pi stb.) - Az érzékelő adatainak feldolgozásához.
- Ellenállás (általában $10\text{k}\Omega$) - Az érzékelő működtetéséhez szükséges, és a DHT11 kimeneti adatát stabilizálja.

- Kábelezés - A DHT11 és a mikrokontroller összekötéséhez.
 - Breadboard (opcionális) - Az alkatrészek ideiglenes összekapcsolásához.
 - Tápellátás - Az érzékelő működtetéséhez szükséges áramforrás (általában 3.3V vagy 5V).
 - Szoftver és könyvtárak - A mikrokontrollerhez szükséges programozási környezet (pl. Arduino IDE) és megfelelő könyvtárak (pl. DHT.h könyvtár)
-

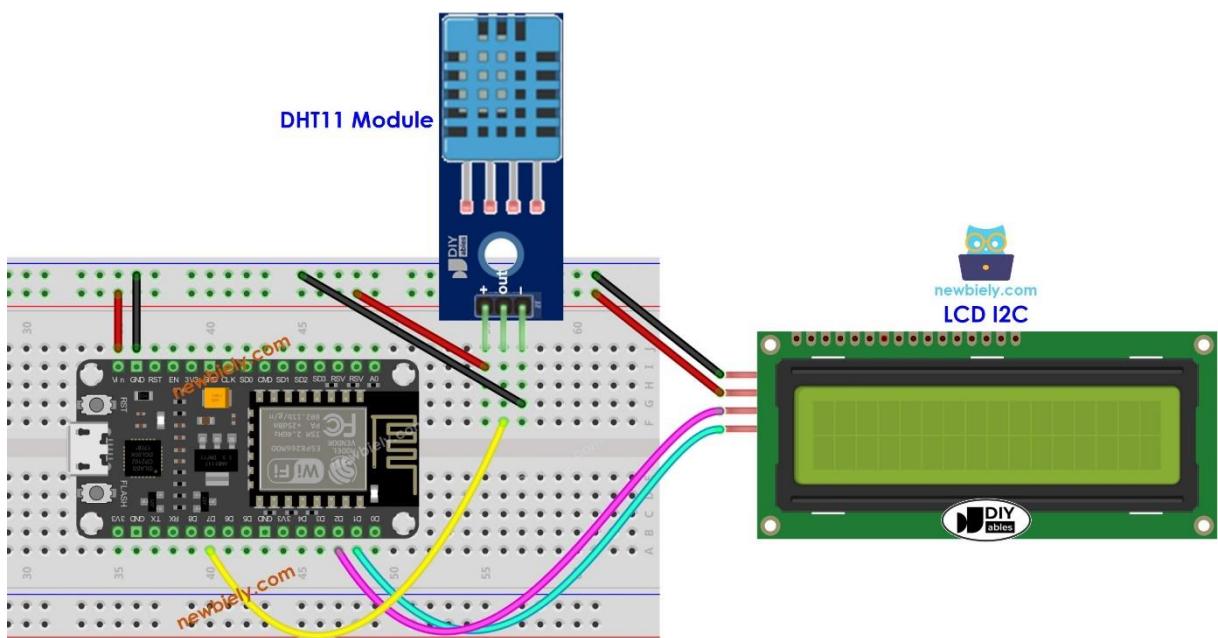
- DHT11 érzékelő: ~ 500 - 1 000 HUF
- Mikrokontroller (pl. Arduino Uno): ~ 3 000 - 5 000 HUF
- Ellenállás (10kΩ): ~ 20 - 100 HUF (általában egy csomagban több is van)
- Breadboard: ~ 1 000 - 2 000 HUF (opcionális)
- Kábelek: ~ 200 - 500 HUF (a legtöbb projekt készletben benne van)
- Tápellátás: Ha az Arduino USB-ről kapja az áramot, akkor nincs külön költség, de ha külső tápegység szükséges, akkor annak ára ~ 1 000 - 2 000 HUF.

Működési elv

A DHT11 egy digitális hőmérséklet- és páratartalom-érzékelő. A páratartalmat kapacitív érzékelővel mér, amelynek kapacitása a levegő nedvességtartalmával változik. A hőmérsékletet egy termisztor érzékeli, melynek ellenállása a hőmérséklet függvényében változik. Az érzékelő digitálisan küldi el az adatokat a mikrokontrollernek, amely dekódolja őket, és kisszámítja a hőmérsékletet és páratartalmat. A DHT11 0-50 °C közötti hőmérsékletet és 20-80% közötti páratartalmat mér, pontossága ±2 °C és ±5%.

Kapcsolási rajz

1. Kép DHT11



Forrás: newbiely.com

Kód példa

```
#include <DHT.h>
```

```
#define DHTPIN 2
```

```
#define DHTTYPE DHT11
```

```
DHT dht(DHTPIN, DHTTYPE);
```

```

void setup() {
    Serial.begin(9600);
    dht.begin();
}

void loop() {
    float h = dht.readHumidity();
    float t = dht.readTemperature();
    if (isnan(h) || isnan(t)) {
        Serial.println("Hiba!");
        return;
    }
    Serial.print("Páratartalom: ");
    Serial.print(h);
    Serial.print(" %\t");
    Serial.print("Hőmérséklet: ");
    Serial.print(t);
    Serial.println(" *C");
    delay(2000); // 2 másodperc várakozás
}

```

Fejlesztési lehetőségek

- Adattárolás:** SD kártya vagy EEPROM.
- Kijelzés:** LCD/OLED kijelző.
- Távfelügyelet:** Wi-Fi vagy Bluetooth kapcsolaton keresztül.
- Riasztás:** Hőmérséklet/páratartalom alapú riasztások.
- Több érzékelő:** Több DHT11 összekapcsolása.

Önreflexió

A Mikrokontrollerek tantárgy során sokat fejlődtem abban, hogy megértsem ezeknek a kis vezérlőegységeknek a működését. Egyre tisztábban látom, hogyan dolgozza fel a mikrokontroller a bemeneteket és hogyan irányítja a kimeneteket. A kódok eleinte bonyolultnak tűntek, de ahogy jobban megismertem a regiszterek és a perifériák szerepét, már nem csak követtem a példákat, hanem meg is értettem őket. Úgy érzem, most már magabiztosabban kezelem a feladatokat és jobban átlátom, hogyan kapcsolódik össze a hardver és a programozás.