

Órai feladat

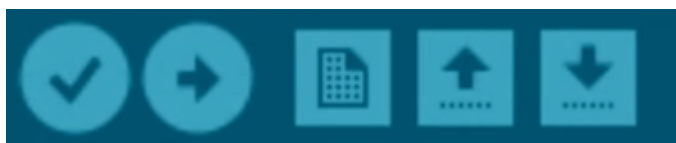
1. feladat ismétlő videó: https://www.youtube.com/watch?v=ZLcKTGI_xHY

- a. Egészítsd ki az Arduino Uno és Nano műszaki paramétereit:

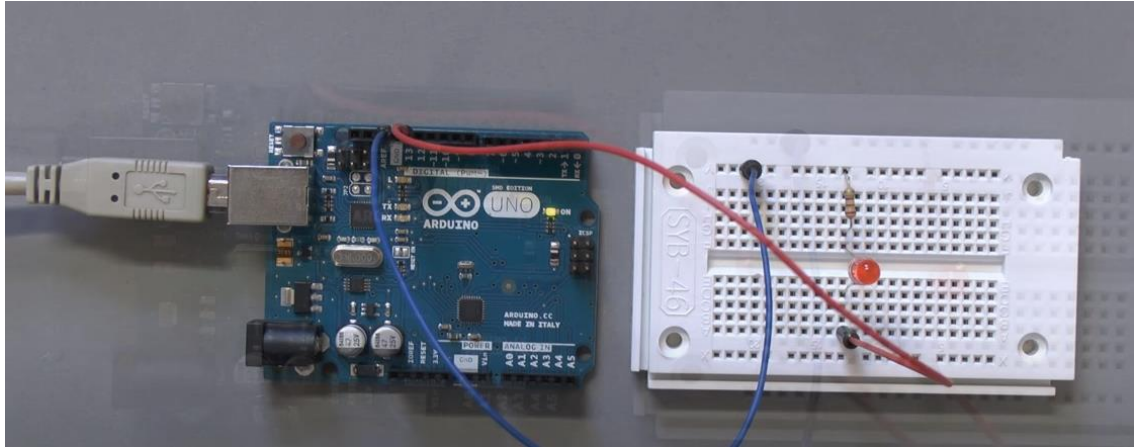
PARAMÉTER	ÉRTÉK
Digitális ki- és bemenetek száma	14, ebből 6 pwm képes
Analóg bemenetek száma	6
Maximális áramerősség I/O lábanként	40ma
Programmemória	32KiB, ebből 0,5 KiB foglalát
Adatmemória	2KiB
EEPROM memória mérete	1KiB
Használt mikrovezérlő	ATmega328 16MHz órajellel

14, ebből 6 pwm képes
6
40ma
32KiB, ebből 0,5 KiB foglalát
2KiB
1KiB
ATmega328 16MHz órajellel

- b. Mi az online fejlesztő környezet elérési útvonala? <http://arduino.cc>
c. A Sketch kezelőfelületen a felső menüszalag alatt található menü funkcióit magyarázd el!



- i. ellenőrzés
 - ii. feltöltés
 - iii. új projekt
 - iv. megnyitás
- d. Egy LED lámpát szeretnénk villogtatni. Hogyan kell bekötni a LEDet? (melyik lábát hova kötöd?)
Fontos az ellenállás helye az áramkörben? (Válaszod indokold!)
rövid negatív, hosszú pozitív(kimeneti porthoz legközelebb)
nem fontos az ellenállás helye



e. Írd le a LEDvillogtató program teljes kódját, magyarázd a működését!

const int pin = 13; változó létrehozása

void setup () {

pinMode (pin, OUTPUT); a 13-as portot deklaráljuk, mint kimeneti port

}

void loop () {

digitalWrite (pin, HIGH); a 13 portnak a feszültségét 5 voltra emeljük, melynek hatására felvillan
delay(1000); késleltetés mely vár a következő sor végre hajtásával

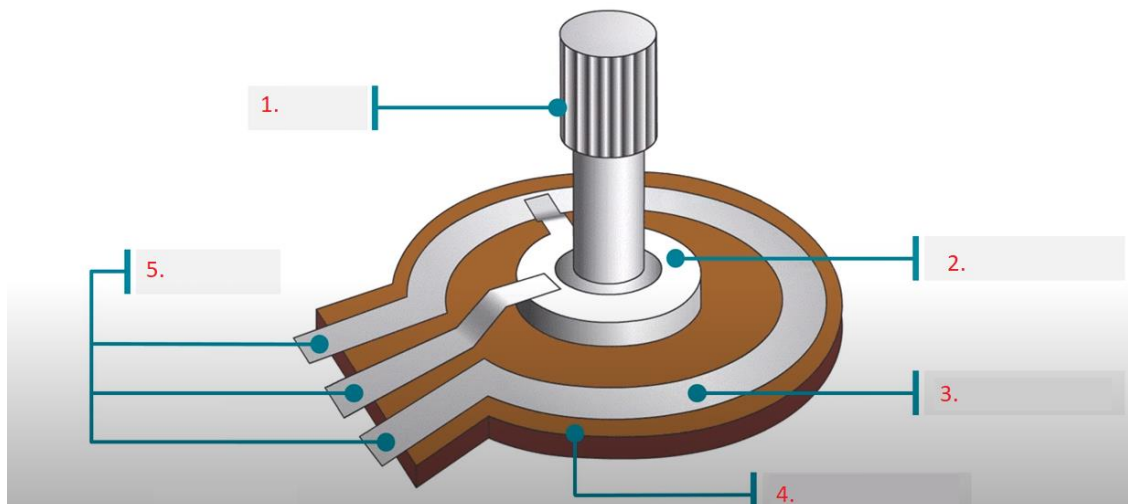
digitalWrite(pin, LOW); a 13 portnak a feszültségét 0 voltra csökkentjük, melynek hatására elalszik
delay(1000); késleltetés mely vár a következő sor végre hajtásával, mely jelen esetben a loop
függvény eleje lesz

}

2. feladat: <https://www.youtube.com/watch?v=-F0vYCWxz0I>

a. Mit nevezünk potenciométernek? Nevezd meg a részeit

Potenciométer felépítése



1. tengely 2. csúszó érintkező 3. ellenállás réteg 4. szigetelt hordozó 5. kivezetések

b. A potencio méter érték kiolvasásához, milyen Arduino összekapcsolás szükséges?

első láb 5volto pin,középső A0, utólsót földeljük

c. Írd le, magyarázd a program működését.

const int potPin = A0;

int potValue = 0;

void setup () {

Serial.begin(9600);

név, osztály: Barsi Péter Szf1A

pinMode(potPin, INPUT) az A0-as portot deklaráljuk, mint bemeneti port

}

void loop () {

portValue = analogRead(potPin); az analog porton lévő feszültséget olvasuk melyet át alakít digitális jellé, és úgy adja át a változónak

Serial.println(portValue); ki írja az éppen aktuális értéket

}

3. feladat: Soros monitor, Soros plotter, potencia mérővel.

https://www.youtube.com/watch?v=BUOiV0G7WmI&list=PLWCAEXWst8J5_CsLPfYDS8QbpIcSICfIO&index=2

a. Mire használjuk a soros monitort, milyen beállítási lehetőségeink vannak?

dibugolásra

b. Mire használjuk a soros plottert?

grafikus megjelenítés, diagram

4. feladat: Digitális jelszintek: <https://www.youtube.com/watch?v=x7xIJ8GHa-M>

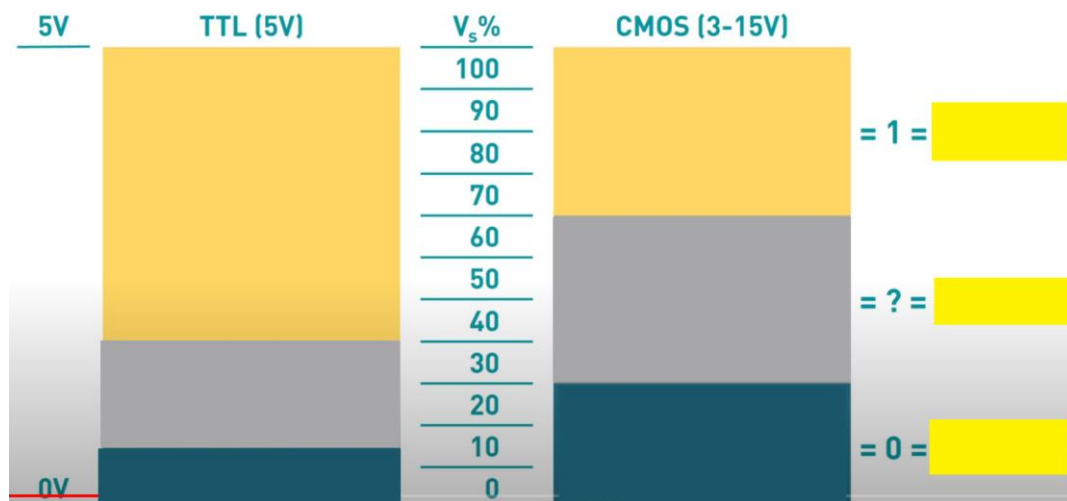
a. Mi az a GPIO periféria?

általános célú ki és bemenetek

b. Egészítsd ki az ábrát!

tTL: magas 2.0V/kétértelmű/alacsony 0.8V

COMS: magas 3.5V/kétértelmű/alacsony 1.5V



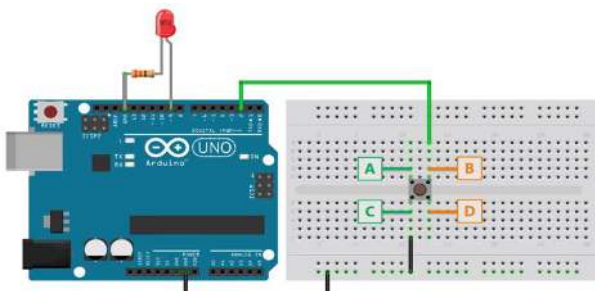
c. Mit neveznek szint illesztésnek?

összehangolja a különböző jelszintekkel dolgozó eszközök magas és alacsony értékeit

d. Hogyan kell bekötni a nyomógombokat?

egyik felét a földre, a másik felét az alduino egyik digitális portjára

e. Illeszd be a nyomógombbal vezérelt LED bekötési rajzát!



f. Magyarázd el a program működését!

név, osztály: Barsi Péter Szf1A

a digitalRead vagy high vagy low értékkel tudott visszatérni, és ennek függvényében változik a led lámpa feszültség értéke

Változók:

```
const int ledPin = 13;
const int buttonPin = 2;
int buttonState = 0;
```



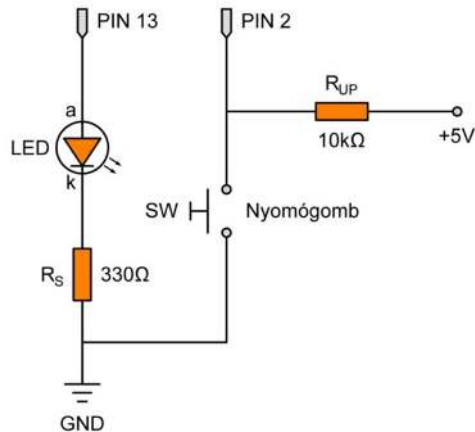
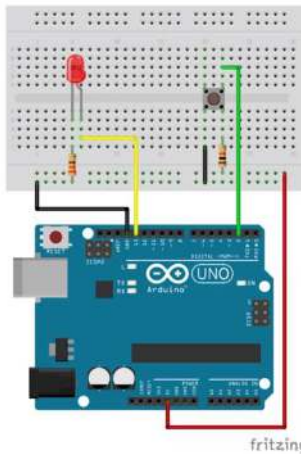
```
void setup()
{
  Serial.begin(9600);
  pinMode(ledPin, OUTPUT);
  pinMode(buttonPin, INPUT);
}
```

```
void loop()
{
  buttonState = digitalRead(buttonPin);

  Serial.println(buttonState);

  if (buttonState == HIGH) {
    digitalWrite(ledPin, HIGH);
  }
  else {
    digitalWrite(ledPin, LOW);
  }
}
```

- g. Mi az a felhúzó ellenállás mire használják? Készíts képet az új kapcsolásról, amiben szerepel már a felhúzó ellenállás!



a lebegő láb kiküszöbölésére való

- h. Mikor beszélünk élő 0-ról, valódi nulláról?
ha a mérendő fizikai jel 0 értékéhez 0 kimeneti jel tartozik
- i. Másold be a gombnyomásra történő led világítás be- /kikapcsolását vezérlő programkódot! Magyarázd a kód működését!

Változók:

```
int ledPin = 13;
int buttonPin = 2;
int buttonState = LOW;
int lastButtonState = LOW;
```

```
void setup()
{
  Serial.begin(9600); // Soros kapcsolat
  pinMode(ledPin, OUTPUT); // GPIO kimenet
  pinMode(buttonPin, INPUT); // GPIO bemenet
}
```

```
void loop()
{
  buttonState = digitalRead(buttonPin); // Gomb új értéke

  if (buttonState == LOW && lastButtonState == HIGH)
  {
    // LED KI/BE kapcsolása
    digitalWrite(ledPin, !digitalRead(ledPin));
  }

  lastButtonState = buttonState; // Gomb régi értéke = új
}
```

- j. Mi az a pergés? Hogyan tudjuk megszüntetni a pergést?(több módszert is ismertess!)

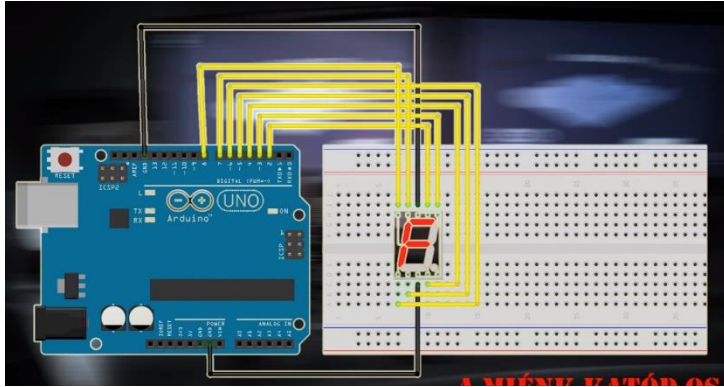
név, osztály: Barsi Péter Szf1A

1-5ms gyors nyit-zár, hardveres: kondenzátor a nyomógommbal párhuzamosan, smith trigger, software-es: gomb megnyomás megtörtént-e ha igen akkor várjon 5ms-ot és ha még mindig be van nyomva akkor biztos hogy meg van nyomva

5. feladat: A 7 szegmenses kijelző és a szervómotor

<https://www.youtube.com/watch?v=zbpmMu70CHY&list=PLWCAEXWsT8J5CsLPfYDS8QbplcSICfIO&index=3>

- a. Illeszd be a 7 szegmenses kijelző használt programot, kapcsolási rajzot hozzá. Magyarázd a működését!



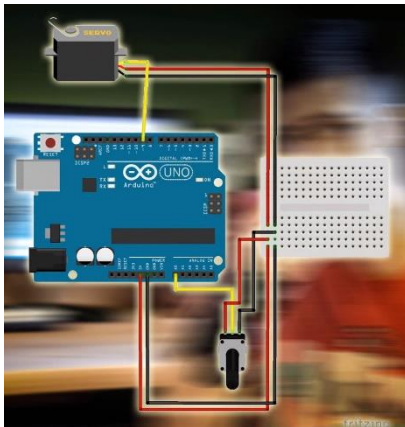
```
int pinA = 2;
int pinB = 3;
int pinC = 4;
int pinD = 5;
int pinE = 6;
int pinF = 7;
int pinG = 8;

void setup() {
  pinMode(pinA, OUTPUT);
  pinMode(pinB, OUTPUT);
  pinMode(pinC, OUTPUT);
  pinMode(pinD, OUTPUT);
  pinMode(pinE, OUTPUT);
  pinMode(pinF, OUTPUT);
  pinMode(pinG, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  //0
  digitalWrite(pinA, HIGH);
  digitalWrite(pinB, HIGH);
  digitalWrite(pinC, HIGH);
  digitalWrite(pinD, HIGH);
  digitalWrite(pinE, HIGH);
  digitalWrite(pinF, HIGH);
  digitalWrite(pinG, LOW);
  delay(1000);
  //1
  digitalWrite(pinA, LOW);
  digitalWrite(pinB, HIGH);
  digitalWrite(pinC, HIGH);
  digitalWrite(pinD, LOW);
  digitalWrite(pinE, LOW);
  digitalWrite(pinF, LOW);
  digitalWrite(pinG, LOW);
  delay(1000);
  //2
  digitalWrite(pinA, HIGH);
  digitalWrite(pinB, HIGH);
  digitalWrite(pinC, LOW);
  digitalWrite(pinD, LOW);
  digitalWrite(pinE, HIGH);
  digitalWrite(pinF, LOW);
  digitalWrite(pinG, LOW);
  delay(1000);
  //3
  digitalWrite(pinA, HIGH);
  digitalWrite(pinB, HIGH);
  digitalWrite(pinC, LOW);
  digitalWrite(pinD, HIGH);
  digitalWrite(pinE, HIGH);
  digitalWrite(pinF, LOW);
  digitalWrite(pinG, LOW);
  delay(1000);
  //4
  digitalWrite(pinA, HIGH);
  digitalWrite(pinB, LOW);
  digitalWrite(pinC, HIGH);
  digitalWrite(pinD, HIGH);
  digitalWrite(pinE, LOW);
  digitalWrite(pinF, LOW);
  digitalWrite(pinG, LOW);
  delay(1000);
  //5
  digitalWrite(pinA, HIGH);
  digitalWrite(pinB, LOW);
  digitalWrite(pinC, HIGH);
  digitalWrite(pinD, LOW);
  digitalWrite(pinE, HIGH);
  digitalWrite(pinF, LOW);
  digitalWrite(pinG, LOW);
  delay(1000);
  //6
  digitalWrite(pinA, HIGH);
  digitalWrite(pinB, LOW);
  digitalWrite(pinC, LOW);
  digitalWrite(pinD, HIGH);
  digitalWrite(pinE, HIGH);
  digitalWrite(pinF, HIGH);
  digitalWrite(pinG, LOW);
  delay(1000);
}
```

az egyes fény led sort a kijelzőben kapcsolgatjuk

- b. Illeszd be a potméterrel vezéreltservo motor használt programot, kapcsolási rajzot hozzá. Magyarázd a működését. (Knob)



```
#include <Servo.h>

Servo myservo; // create servo object to control a servo

int potpin = 0; // analog pin used to connect the potentiometer
int val; // variable to read the value from the analog pin

void setup() {
  myservo.attach(9); // attaches the servo on pin 9 to the servo object
}

void loop() {
  val = analogRead(potpin); // reads the value of the potentiometer (value between 0 and 1023)
  val = map(val, 0, 1023, 0, 180); // scale it to use it with the servo (value between 0 and 180)
  myservo.write(val); // sets the servo position according to the scaled value
  delay(15); // waits for the servo to get there
}
```

a potmeter állásának megfelelő szöget vesz fel a motor is

név, osztály: Barsi Péter Szf1A

- c. Illeszd be a potméterrel vezéreltservo motor használt programot, kapcsolási rajzot hozzá. Magyarázd a működését. (Sweep)

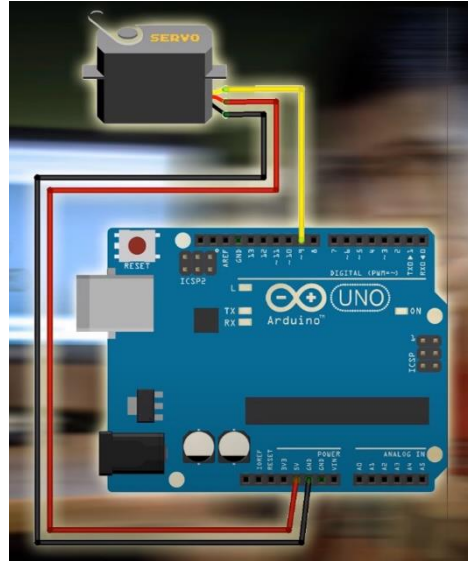
```
#include <Servo.h>

Servo myservo; // create servo object to control a servo
// twelve servo objects can be created on most boards

int pos = 0;    // variable to store the servo position

void setup() {
  myservo.attach(9); // attaches the servo on pin 9 to the Arduino
}

void loop() {
  for (pos = 0; pos <= 180; pos += 1) { // goes from 0 to 180 in steps of 1 degree
    // in steps of 1 degree
    myservo.write(pos); // tell servo to go to position
    delay(15);           // waits 15ms for the servo to reach the position
  }
  for (pos = 180; pos >= 0; pos -= 1) { // goes from 180 to 0 in steps of 1 degree
    myservo.write(pos); // tell servo to go to position
    delay(15);           // waits 15ms for the servo to reach the position
  }
}
```

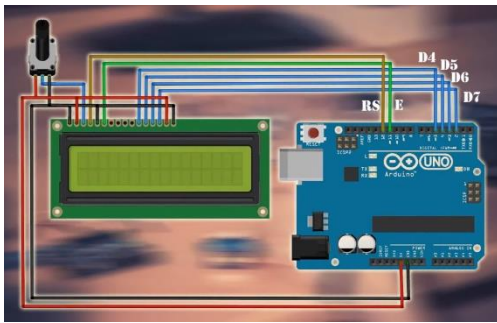


egy ciklusban növeljük a motorforgási szögét 180-ig majd egy másikban ugyanezt csak visszafele

6. feladat: LCD kijelző: <https://www.youtube.com/watch?v=bRauxEo3SuY>

a.

Ismertesd a két bemutatott program bekötését és működését!



```
#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup()
{
  lcd.begin(16, 2);
  lcd.clear();
}

void loop()
{
  lcd.clear();
  lcd.print("Hello World");

  delay(2500);

  lcd.clear();
  lcd.setCursor(0,1);
  lcd.print("kurzor beall.");

  delay(2500);

  lcd.clear();
  lcd.blink();
  lcd.print("kurzor villog");
}
```

```
delay(2500);

lcd.noBlink();
lcd.clear();
lcd.setCursor();
lcd.print("Alulvon. kurzor");

delay(2500);

lcd.clear();
lcd.noCursor();
lcd.print("nincs kurzor");

delay(2500);
```

ennél folyamatosan többféle dolog jelenik meg és a cursor-t is megjelenítjük

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

byte newChar[8] = {
  B00000,
  B11111,
  B00001,
  B11111,
  B00000,
  B11110,
  B00100,
  B11111
};

void setup() {
  lcd.begin(16, 2);
  lcd.clear();
  lcd.createChar(1, newChar);
}

void loop() {
  lcd.setCursor(0,0);
  lcd.write(1);
  lcd.write(1);
}
```

itt egy byte tömb elemei szerint létrehozott betűt íratunk ki