

# Szervomotorok

Agócs Norbert és Nagy Dániel

2020. szeptember 8.

# Tartalom

- 1 Ismétlés
- 2 Ismétlő feladatok
- 3 Programozás alapok: A for ciklus
- 4 Feladatok
- 5 Arduino alapok: SG90 Micro Servo
- 6 Szótár

# Tartalom

- 1 Ismétlés
- 2 Ismétlő feladatok
- 3 Programozás alapok: A for ciklus
- 4 Feladatok
- 5 Arduino alapok: SG90 Micro Servo
- 6 Szótár

# Mit tanultunk eddig?

## Hardware:

- Arduino alaplap
- Breadboard
- LED
- Ellenállás
- Gomb
- Ultrahang szenzor

## Software

- Változó létrehozása
- Alapműveletek ( $=$ ,  $+$ ,  $-$ ,  $/$ ,  $*$ )
- Alapvető parancsok
  - `Serial.begin(9600)`
  - `Serial.println(" ")`
  - `pinMode()`
  - `digitalWrite()`
  - `delay()`
  - `digitalRead()`
  - `pulseIn()`
- Elágazás: `If() ...else`

# Ismétlő teszt

Hány állítás igaz?

- A LED-eket közvetlenül az Arduinoba köthetjük
- Az ellenállások méretét a színeik alapján lehet eldönteni
- Az ultrahangos szenzoron TRIGGER a kimenet és ECHO a bemenet
- A gombok lenyomását a `digitalWrite()` függvénnyel érzékeltük
- Az `int` egész számokat, a `float` tört számokat tárol.

# Ismétlő teszt

Hány állítás igaz?

- x A LED-eket közvetlenül **nem** köthetjük az Arduinoba, **kell hozzá ellenállás is**
- Az ellenállások méretét a színeik alapján lehet eldönteni
- Az ultrahangos szenzoron TRIGGER a kimenet és ECHO a bemenet
- x A gombok lenyomását a `digitalRead()` (~~`digitalWrite()`~~) függvénnyel érzékeltük
- Az `int` egész számokat, a `float` tört számokat tárol.

3 igaz

# Tartalom

- 1 Ismétlés
- 2 Ismétlő feladatok
- 3 Programozás alapok: A for ciklus
- 4 Feladatok
- 5 Arduino alapok: SG90 Micro Servo
- 6 Szótár

# Feladat I

Mit ír ki a kód?

```
void setup() {  
  Serial.begin(9600);  
}  
  
void loop()  
{  
  int a = 5;  
  Serial.println("a");  
  Serial.println(a);  
}
```



# Feladat I

Mit ír ki a kód?

```
void setup() {  
    Serial.begin(9600);  
}  
  
void loop()  
{  
    int a = 5;  
    Serial.println("a");  
    Serial.println(a);  
}
```

Megoldás:

- `Serial.println("a");` az "a" karektert írja ki
- `Serial.println(a);` az a változó értékét írja ki

# Feladat II

Mit ír ki a kód?

```
void setup() {  
    Serial.begin(9600);  
}  
  
void loop(){  
    int alma = 10;  
    int korte = -10;  
    int eper = 5;  
    if(alma < korte){  
        Serial.print("eper");  
    }  
    else{  
        Serial.print(eper);  
    }  
}
```

# Feladat II

Mit ír ki a kód?

```
void setup() {  
  Serial.begin(9600);  
}  
  
void loop(){  
  int alma = 10;  
  int korte = -10;  
  int eper = 5;  
  if(alma < korte){  
    Serial.print("eper");  
  }  
  else{  
    Serial.print(eper);  
  }  
}
```

**Megoldás:** A feltétel `alma < korte` hamis, ezért az `else`-be írt rész fut le. Az `eper` változó értéke 5, tehát a megoldás 5.

# Feladat III

Mit csinálhat a kód?

```
int LED = 2;
int GOMB = 3;
void setup() {
    pinMode(LED, OUTPUT);
    pinMode(GOMB, INPUT);
}

void loop() {
    int GOMB_ALLAPOT = digitalRead(GOMB);
    if (GOMB_ALLAPOT == HIGH) {
        digitalWrite(LED, HIGH);
    }
    else {
        digitalWrite(LED, LOW);
    }
}
```

## Feladat III

Mit csinálhat a kód?

```
int LED = 2;
int GOMB = 3;
void setup() {
  pinMode(LED, OUTPUT);
  pinMode(GOMB, INPUT);
}

void loop() {
  int GOMB_ALLAPOT = digitalRead(GOMB);
  if (GOMB_ALLAPOT == HIGH) {
    digitalWrite(LED, HIGH);
  }
  else {
    digitalWrite(LED, LOW);
  }
}
```

**Megoldás:** Ha lenyomjuk a 3-as PIN-re kapcsolt gombot, akkor felvillan a 2-es PIN-re kapcsolt LED.

# Feladat IV

Mit csinálhat a kód?

```
int LED = 2;
int GOMB = 3;
void setup() {
  pinMode(LED, OUTPUT);
  pinMode(GOMB, INPUT);
}

void loop() {
  int GOMB_ALLAPOT = digitalRead(GOMB);
  digitalWrite(LED, GOMB_ALLAPOT);
}
```

# Feladat IV

Mit csinálhat a kód?

```
int LED = 2;
int GOMB = 3;
void setup() {
  pinMode(LED, OUTPUT);
  pinMode(GOMB, INPUT);
}

void loop() {
  int GOMB_ALLAPOT = digitalRead(GOMB);
  digitalWrite(LED, GOMB_ALLAPOT);
}
```

**Megoldás:** Ha lenyomjuk a 3-as PIN-re kapcsolt gombot, akkor felvillan a 2-es PIN-re kapcsolt LED.

# Tartalom

- 1 Ismétlés
- 2 Ismétlő feladatok
- 3 Programozás alapok: A for ciklus**
- 4 Feladatok
- 5 Arduino alapok: SG90 Micro Servo
- 6 Szótár



# for loop/ciklus

## Mikor használjuk?

Akkor használjuk ha egy utasítást egy előre ismert alkalommal szeretnénk hogy lefusson.

## Példák

- Az első 25 szám kiiírása
- Az első 10 prímszám megkeresése
- 1500 darab mérés elvégzése
- 1500 adat kiátlagolása

# Példa

Az első 25 szám kiírása

```
void setup() {  
    Serial.begin(9600);  
}  
  
void loop() {  
    for(int i = 1; i < 25; i = i + 1)  
    {  
        Serial.println(i);  
    }  
}
```

**Magyarázat:** Az `int i = 1` a kezdeti érték, `i < 25` a végső érték vagy kilépési feltétel, `i = i + 1` az `i` érték növelése.

# Példa

## 1000 mérés elvégzése

```
int SzenzorPin = 3;
void setup() {
    Serial.begin(9600);
    pinMode(SzenzorPin, INPUT);
}

void loop() {
    for(int meresSzam = 0; meresSzam < 1000; meresSzam =
        meresSzam + 1)
    {
        int adat = digitalRead(SzenzorPin)
    }
}
```

**Magyarázat:** Az `int meresSzam = 0` a kezdeti értéke a számlálónak, `meresSzam < 1000` a végső érték eddig megyünk, aztán a `meresSzam = meresSzam + 1` megnöveli a `meresSzam`-ot, miután elvégeztünk egy mérést

# For ciklus/loop

## Létrehozás

```
for(kezdeti érték; feltétel; számlálónövelés)
```

# For ciklus/loop

## Létrehozás

```
for(kezdeti érték; feltétel; számlálónövelés)
```

## kezdeti érték

A ciklusváltozó első értéke. pl: `int i = 1` vagy `int meresSzam = 0`

# For ciklus/loop

## Létrehozás

```
for(kezdeti érték; feltétel; számlálónövelés)
```

## kezdeti érték

A ciklusváltozó első értéke. pl: `int i = 1` vagy `int meresSzam = 0`

## feltétel

A ciklus addig tart amíg ez a feltétel teljesül. pl: `i < 25` vagy  
`meresSzam < 1000`

# For ciklus/loop

## Létrehozás

```
for(kezdeti érték; feltétel; számlálónövelés)
```

## kezdeti érték

A ciklusváltozó első értéke. pl: `int i = 1` vagy `int meresSzam = 0`

## feltétel

A ciklus addig tart amíg ez a feltétel teljesül. pl: `i < 25` vagy  
`meresSzam < 1000`

## számláló növelés

A ciklusváltozó értékét változtatjuk, ez kell ahhoz hogy a ciklus egyszer véget érjen. pl: `i = i+1` vagy `meresSzam = meresSzam + 1`

# For ciklus

**Létrehozása:** `for(kezdeti érték; feltétel; számláló növelés)`

- **Kezdeti érték:** A ciklusváltozó első értéke.
  - 1. argumentum
  - Példa: `int i = 1` vagy `int meresSzam = 0`
- **Feltétel:** A ciklus addig tart amíg ez a feltétel teljesül.
  - 2. argumentum
  - Példa: `i < 25` vagy `int meresSzam = 1000`
- **Számláló növelése:** A ciklusváltozó értékét változtatjuk, ez kell ahhoz hogy a ciklus egyszer véget érjen.
  - 3. argumentum
  - Példa: `i = i+1` vagy `meresSzam = meresSzam + 1`



# Tartalom

- 1 Ismétlés
- 2 Ismétlő feladatok
- 3 Programozás alapok: A for ciklus
- 4 Feladatok**
- 5 Arduino alapok: SG90 Micro Servo
- 6 Szótár

# Feladat I

Mit ír ki a kód?

```
void setup() {  
    Serial.begin(9600);  
}  
  
void loop() {  
    for(int j = 10; j < 20; j = j + 2)  
    {  
        Serial.println(j);  
    }  
}
```

# Feladat I

Mit ír ki a kód?

```
void setup() {  
    Serial.begin(9600);  
}  
  
void loop() {  
    for(int j = 10; j < 20; j = j + 2)  
    {  
        Serial.println(j);  
    }  
}
```

**Megoldás:** 10 12 14 16 18, 10-nél kezdődik hiszen `int j = 10` a kezdeti érték, a számlálót mindig 2-vel növeljük (`j = j + 2`) és a 20 már nincsen benne hiszen `20 < 20` már hamis.

# Feladat II

Mit ír ki a kód?

```
void setup() {  
  Serial.begin(9600);  
}  
  
void loop() {  
  for(long j = 1; j <= 128; j = j * 2)  
  {  
    Serial.println(j);  
  }  
}
```

# Feladat II

Mit ír ki a kód?

```
void setup() {  
    Serial.begin(9600);  
}  
  
void loop() {  
    for(long j = 1; j <= 128; j = j * 2)  
    {  
        Serial.println(j);  
    }  
}
```

**Megoldás:** 1 2 4 8 16 32 64 128, 1-nél kezdődik hiszen `int j = 1` a kezdeti érték, a számlálót mindig **megkétszerezünk** (`j = j * 2`) és a 128 még benne van hiszen `128 <= 128` még igaz.

# Feladat III

Mit ír ki a kód?

```
void setup() {  
    Serial.begin(9600);  
}  
  
void loop() {  
    int lepesKoz = 5;  
    for(int v = 20; v >= -2; v = v - lepesKoz)  
    {  
        Serial.println(v);  
    }  
}
```

# Feladat III

Mit ír ki a kód?

```
void setup() {  
    Serial.begin(9600);  
}  
  
void loop() {  
    int lepesKoz = 5;  
    for(int v = 20; v >= -2; v = v - lepesKoz)  
    {  
        Serial.println(v);  
    }  
}
```

**Megoldás:** 20 15 10 5 0, 20-nál kezdődik hiszen `int v = 20` a kezdeti érték, a számlálóból mindig lépésköznnyit vonunk ki, ami éppen 5 tehát  $v = v - 5$ ) és a 0 még benne van hiszen  $0 \geq -2$  még igaz viszont utána  $-5 \geq -2$  már nem igaz.

# Tartalom

- 1 Ismétlés
- 2 Ismétlő feladatok
- 3 Programozás alapok: A for ciklus
- 4 Feladatok
- 5 Arduino alapok: SG90 Micro Servo**
- 6 Szótár



# SG90 Micro Servo

## Alapok



### A szervomotor előnyei:

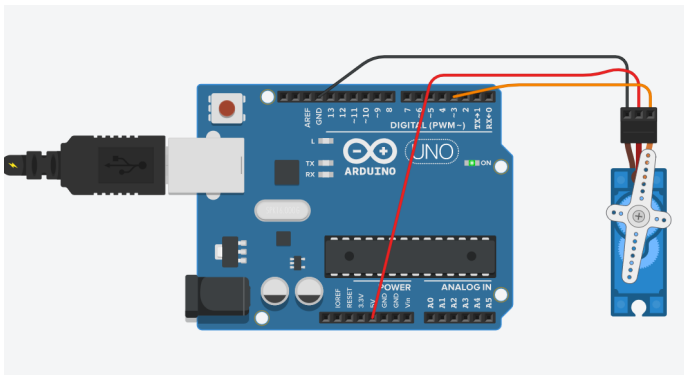
- Olcsó (kb. 500 ft)
- Egyszerű használni
  - Mindössze 3 kimenet
  - Közvetlenül Arduinora köthető
- $\pm 5^\circ$  pontosság

# SG90 Micro Servo

## Bekötés

### Bekötés:

- Piros - 5V
- Fekete/Barna - GND
- Narancssárga - PWM PIN, most 3 de lehet 3,5,6,9,10,11 bármelyike



# Függvények

```
Servo myservo;
```

A szervomotort kezelő objektum létrehozása myservo néven. Innentől minden függvény elé a myservo szót kell írni.

```
myservo.attach(PIN);
```

A szervomotor inicializálása, ezzel kell megadni hogy melyik PIN-re kötöttük be a Servot.

```
myservo.write(szög);
```

A szervomotor elfordítása a megadott szöghöz.

```
myservo.detach();
```

A szervomotor lecsatlakoztatása

**Figyelem** a myservo az objektum neve, ha máshogy nevezzük el, akkor más nevet kell a helyére. A Servo.h egy úgynevezett **könyvtár**.

# Függvények

A szervo irányításához szükséges függvények:

- `Servo myservo`: A szervomotort kezelő objektum létrehozása `myservo` néven.
  - Innentől minden függvény elé a `myservo` szót kell írni.
  - Példa: `myservo.függvény()`
- `myservo.attach(PIN)`: A szervomotor inicializálása, ezzel kell megadni hogy melyik PIN-re kötöttük be a Servot.
- `myservo.write(szög)`: A szervomotor elfordítása a megadott szöghöz.
- `myservo.detach()`: A szervomotor lecsatlakoztatása

**Figyelem** a `myservo` az objektum neve, ha máshogy nevezzük el, akkor más nevet kell a helyére. A `Servo.h` egy úgynevezett **könyvtár**.

# A szervomotor programozása

## Középre állítás

```
#include <Servo.h>
int ServoPin = 3;
Servo motor1;

void setup() {
  motor1.attach(ServoPin);
}

void loop() {
  motor1.write(90);
}
```

### Könyvtár beimportálása:

- Az első sorban az `include` azért kell, hogy a szervomotor működéséhez szükséges függvényeket elérjük.

**Feladat:** Írjuk át a 90-et és próbáljunk ki más szögeket.

# A szervomotor forgatása oda-vissza

```
#include <Servo.h>
int ServoPin = 3;
Servo motor1;

void setup() {
  motor1.attach(ServoPin);
}

void loop() {
  motor1.write(0);
  delay(2000);
  motor1.write(180);
  delay(2000);
}
```

# A szervomotor forgatása oda-vissza for ciklussal

```
#include <Servo.h>

int ServoPin = 3;
int ido = 5000; //ennyi ms legyen a teljes fordulat
Servo motor1;

void setup() {
  motor1.attach(ServoPin);
}

void loop() {
  for(int szog = 0; szog < 180; szog = szog + 1)
  {
    motor1.write(szog);
    delay(ido/180); //minden lépés után ido/180 várakozás,
                  //összes így pont ido-nyi várakozás lesz
  }
}
```

# Feladat

Oda vissza forgás megadott idő alatt



# Feladat

Oda vissza forgás megadott idő alatt

```
#include <Servo.h>
int ServoPin = 3;
int ido = 6000; //ennyi ms legyen a teljes fordulat
Servo motor1;

void setup(){
  motor1.attach(ServoPin);
}

void loop(){
  for(int szog = 0; szog < 180; szog = szog + 1){
    motor1.write(szog);
    delay(ido/360);
  }
  for(int szog = 180; szog > 0; szog = szog - 1){
    motor1.write(szog);
    delay(ido/360);
  }
}
```

# Tartalom

- 1 Ismétlés
- 2 Ismétlő feladatok
- 3 Programozás alapok: A for ciklus
- 4 Feladatok
- 5 Arduino alapok: SG90 Micro Servo
- 6 Szótár**

## Szavak I

r Sensor /-en	szenzor
r Aufnehmer /-	érzékelő
s Gerät /-e	eszköz, berendezés
messen +Akk.	vmi-t mérni
s Signal /-e	jel
r Thermistor /-en	termisztor
e Temperatur	hőmérséklet
r Fotowiderstand /-widerstände	fotorezisztor
e Lichtstärke /-n	fényerősség
r Schall /e Schälle	hang
r Ultraschall	ultrahang
r Ultraschallsensor	ultrahangszenzor
e Entfernung /-en	távolság
e Welle	hullám
s. verbreiten in+A	vmiben terjedni

## Szavak II

periodisch	periódikus
e Veränderung /-en	megváltozás
s Licht	fény
e Größe /-n	fizikai mennyiség
e Ausbreitungsgeschwindigkeit	terjedési sebesség
e Frequenz	frekvencia
e Wellenlänge	hullámhossz
vermeiden + Akk.	vmit elkerülni
emittieren + Akk.	vmit kibocsátani
aufnehmen + Akk.	vmit érzékelni
e Reflexion	visszaverődés
berechnen + Akk.	kiszámolni vmit
e Stromversorgung	áramellátás
e Einparkhilfe	tolatóradar