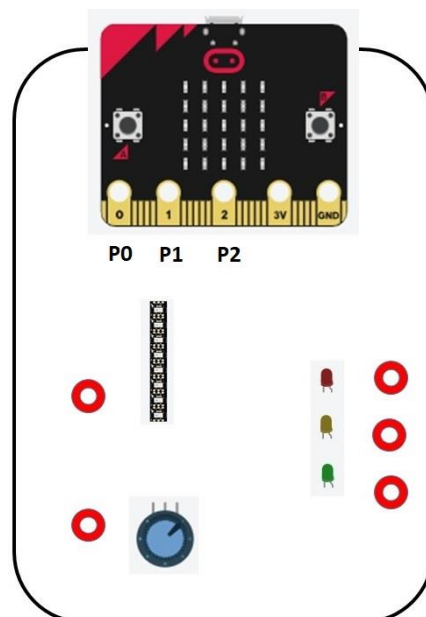


BASIC

kísérletező készlet (kézikönyv és példatár)



VÍGVÁRI
R E N D S Z E R H Á Z

Ezt a készletet elsősorban azoknak ajánlom, akik már ismerkedtek a micro:bit programozásával, ill. használatával.

Elsősorban 5. és 6. évfolyamon használom a tanulókkal.

A gyermekek hamar megünnak egy felületet, ha az nem kínál fel különböző tevékenységeket, illetve nem elégíti ki más és más igényeiket. A gyermekek figyelme hamar ellankad akkor is, ha csak hasonló tevékenységeket lehet végezni, és viszonylag hamar ki lehet ismerni a lehetőségek határait.

A **micro:bit** egy olyan eszköz, amely a programozás oktatásán túl tökéletesen alkalmas a fizikai számítástechnika tanítására is

A fizikai számítástechnika tanítása során a megvalósítható projektek száma **korlátlan!** Itt csak bemutató jelleggel ismertetek néhány kapcsolást és feladatot, de tervezem, hogy a kollégákkal folyamatosan bővítsük a feladatbázist. **Nagyon nehéz rábeszélni a biztos alapok alapos megértésére a felhasználókat, de higgyétek el nekem, hogy csak ez után tudtok építkezni.**



Physical
Computing

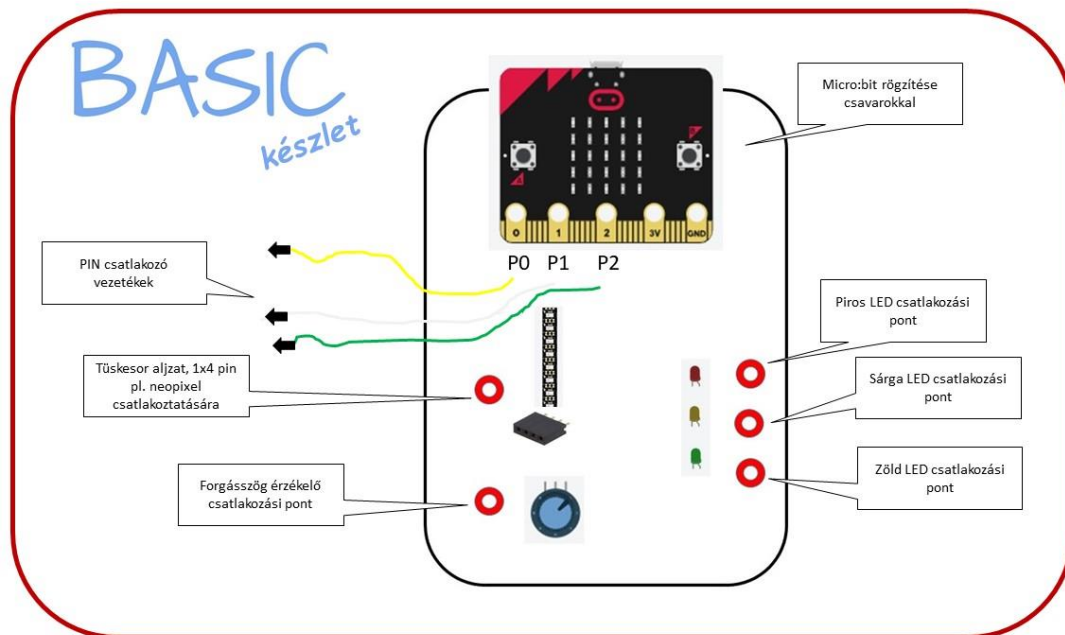
Már a kifejezés is eléggé ijesztő fizikai számítástechnika, de mit is jelent ez valójában?

Szűkebb értelemben olyan interaktív rendszerek létrehozását jelenti hardverek és szoftverek segítségével, melyek képesek érzékelni a világban létrejövő jeleket és reagálni is tudnak rá, és **lehetővé teszi a tanulók számára, hogy konkrét, kézzelfogható projekteket dolgozzanak ki a való világból, amelyek a tanulók képzeletéből fakadnak.**

A készlet és a gyakorlatok célja, hogy a felhasználó hozzájuthasson minimális elektronikai alapokhoz, jobban megismerjék a micro:bitet amelynek a ki és bemenetei alkalmasak arra hogy vezéreljenek eszközöket, és legyen képes néhány egyszerű alkalmazás létrehozására.

Az oktatás szükségessége nehezen vitatható, hiszen a háztartásokban jelenleg is számos olyan elektronikus eszköz működik, amely processzorvezérelt technológiára épül, a szórakoztató és kommunikációs elektronikától a fejlettebb háztartási gépekig. Ezeknek az eszközöknek az aránya egyre nő, és a jövő generációjának **a gombok nyomogatásán túlmutató kompetenciákkal kell rendelkeznie**, hogy ne váljon mindez a „varázslat” és „misztikum” eszközévé, hanem az egyszerű eszközhasználó is lássa és tudja a mögöttes algoritmusok emberi kreativitásban rejlő működését.

A Basic készlet felépítése:



Ez egy nagyon egyszerű eszköz, ez is volt a cél egyszerű és olcsó használható eszközt létrehozni.



Rögzítsük a Micro:bitet a menetes hüvelyekhez, riasztó lehet a csavarhúzó és a lógó vezetékek, de nem szokott problémát okozni.

Csatlakoztassuk a micro:bitet USB kábelen keresztül a számítógépünkhöz ez a feltétele, hogy majd le tudjuk tölteni a programunkat. Természetesen nem csak ezért csatlakoztatjuk, hanem a tápegység (áramellátás) szerepét is betölti ez a kapcsolat (később, ha majd a programunkat letöltöttük, akkor használhatunk akár powerbankot vagy 2 db elemet is, így válik hordozhatóvá az alkalmazásunk).

Ha megtörtént a csatlakozás ez azt is jelenti, hogy a GND (föld) és a 3V-os PIN között valóban megjelenik a 3,3V (3V3) feszültség.

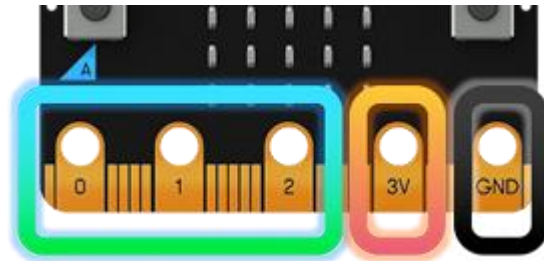
Nem veszélyes ez?

Nem veszélyes, hiszen annyira pici az áramerősség, ami teljesen elhanyagolható az emberi szervezetre, de ha egy vezetékkel ezt összekötöm, jó eséllyel tönkre megy a micro:bitem, tehát tessék odafigyelni.

Alapok

PIN -ek

Mi az?



élcsatlakozó (breakout)

A BBC micro:bitnek 20 külső csatlakozása van a készülék perem csatlakozóján, amit "PIN" -eknek nevezünk, amelyekhez általában programokkal használható elektronikus alkatrészek csatlakoztathatók

Öt nagy csatlakozási pont van, amelyek a 0, 1, 2, 3V és GND lemezek lyukaira van csatlakoztatva eszközünk.

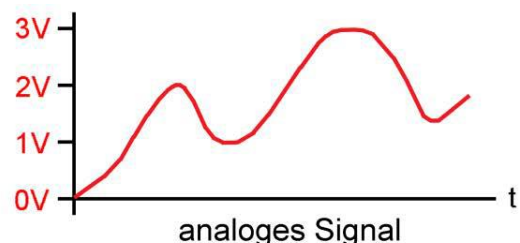
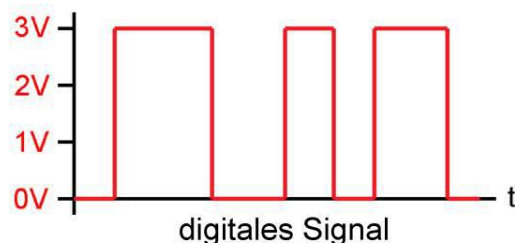
Ez a három pin, programtól függően, digitális és analóg be- és kimenetként is használható

Analóg és digitális

Az elektronikus alkatrészek digitális vagy analóg jeleket használnak. Egy digitális jel azt jelenti, hogy a komponens bemenete vagy kimenete 3V3 vagy 0V (nincs harmadik!). A digitális információ olyan, mint egy bináris - ez lehet a két érték közül az egyik. A gombok vagy a kapcsolók jó példák a digitális bemenetekre.

Az analóg komponensek fogadják a bemenetet vagy a kimenetet 0V-tól 3V3-ig terjedő feszültségtartományból. A mikrokontroller tartalmaz egy ADC-t (analóg-digitális átalakító), hogy ezeket az analóg jeleket a mikrokontroller által használható értékké alakíthassa.

A mikro: bitnek 10 bites ADC-ja van, így az analóg bemenet 0 - 1023 tartományba kerül. A potenciométerek és a fényérzékelők jó példák az analóg jeleket használó komponensekre.



Tehát fontos, hogy példákat szolgáltatassak a digitális be-ki menetre és az analóg be-ki menetre, viszont ha ezeket az egyszerű példákat megértjük, olyan készségre teszünk szert, amellyel már komoly robotok programozásának és eszközökkel való bővítésének is nekiállhatunk.

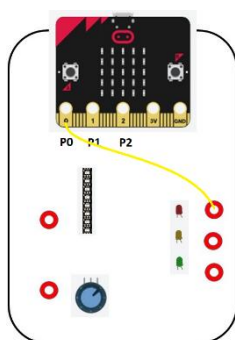
Digitális kimenet

1.feladat: A piros LED villogtatása

Terv: A LED villogása az egyik alapkísérlet („Helló világ!”).

A mikrokontrollereket így a micro:bitet nem arra találták ki, hogy közvetlenül lámpákat kapcsolgassunk, de a digitális (igaz) esetén 3V3 feszültség éppen elég, hogy a LED világítson, így jól lehet szemléltetni. (pl. motorokat nem tudunk közvetlenül meghajtani, de egy motorvezérlő segítségével már tudjuk működtetni, ill. szabályozni a sebességét és mozgásának irányát).

Kapcsolási rajz:

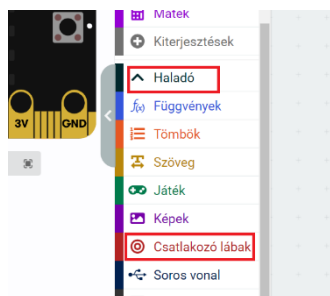


- digitális igaz (1) esetén a kiválasztott PIN-en mért érték 3,3 V
- digitális hamis (0) esetén a kiválasztott PIN-en mért érték 0V!

Gondoljunk bele már két PIN-en akár négy különböző jelet tudunk küldeni,

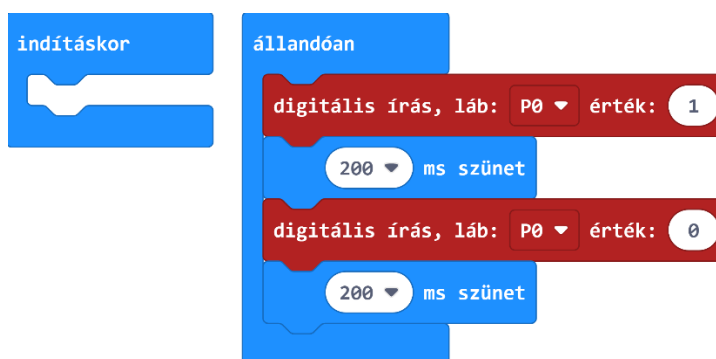
A piros LED a „P0”-n váltakozva be- és kikapcsol végtelen sokszor azaz állandóan.

Programkód:



Szükségünk lesz olyan utasításra ahol elérhetők a csatlakozó lábak.

Haladó\Csatlakozó lábak



Eredmény tesztelése:

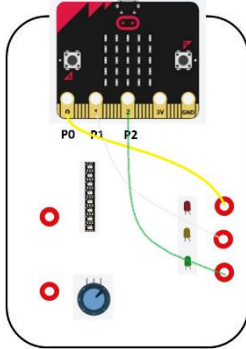
A kód letöltése után, láthatjuk, hogy a PIN0-ra csatlakoztatott piros LED villog, körülbelül 0,2 másodperces intervallummal.

A piros helyett csatlakoztathatjuk a sárga vagy a zöld LED-et, de változtathatjuk a PIN-eket is.

2.feladat: A három LED villogtatása

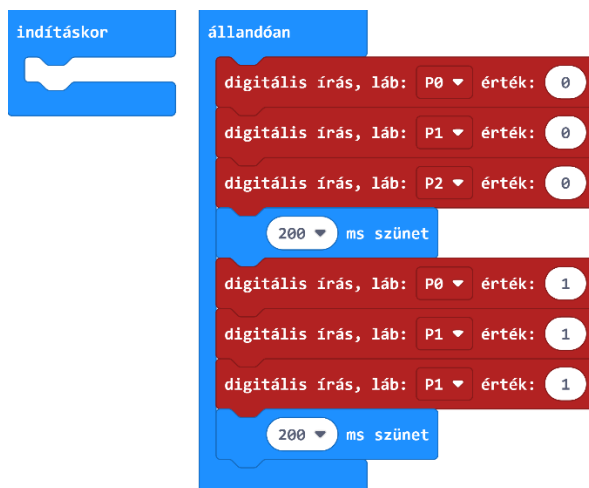
Terv: Egyszerre is lehet használni akár mind a három LED-et

Kapcsolási rajz:



Ha voltak kételyeink a teljesítménnyel kapcsolatban, most eloszlik hiszen mind a három LED világítani fog.

Programkód:



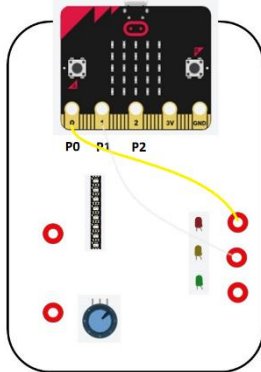
Eredmény tesztelése:

A kód letöltése után, láthatjuk, hogy a három LED villog, körülbelül 0,2 másodperces intervallummal. Természetesen változtathatjuk a szünet értékét is.

3.feladat: Két LED felváltva villog

Terv: Egyszerre kettő LED-et fogunk használni amelyek felváltva világítanak.

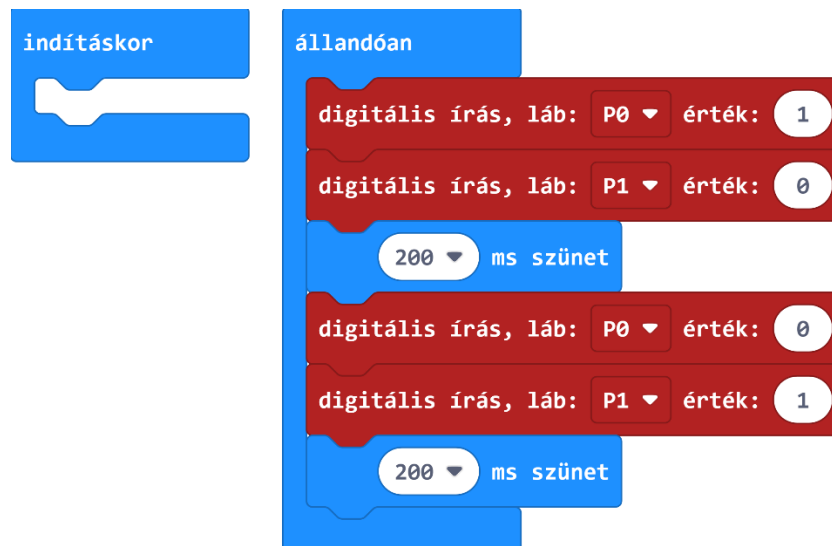
Kapcsolási rajz:



P0-ra kötjük a piros, P1-re kötjük a sárga LED-et

(Fontos lehet, hogy a banándugókat fogjuk meg és ne a vezetéket húzzuk, mert utóbbi tevékenységgel előbb utóbb kilazul a vezeték)

Programkód:



Eredmény tesztelése:

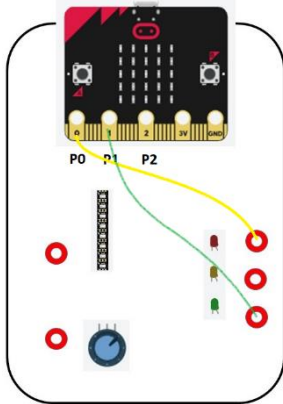
Mindig csak az egyik világít, itt is érdemes játszani azzal, hogy változtatjuk a szünet értékét.

(1 sec=1000 msec)

4.feladat: Gyalogos lámpa

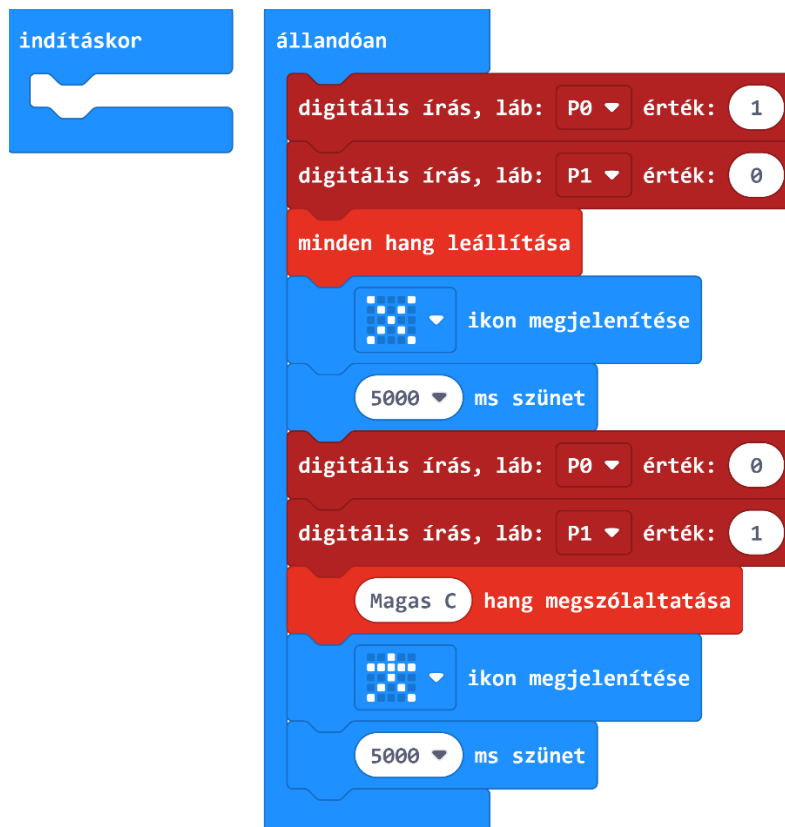
Terv: Kettő LED-et fogunk használni amelyek felváltva világítanak.

Kapcsolási rajz:



A gyalogos lámpán csak két LED van, piros és zöld.
Kiegészítjük a programot vizuális és hang élménnyel.

Programkód:



Eredmény tesztelése:

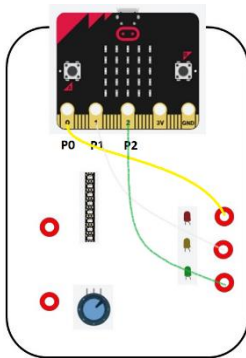
Ha a piros lámpa világít a micro:bit LED mátrixán megjelenik egy X felirat tetszőleges idő múlva, zöldre vált megjelenik a mozgó emberke és hanggal is jelezzük a szabad átkelést.

5.feladat: Háromfényes közlekedési jelzőlámpa

Terv: A forgalmas útkereszteződésekben gyakran jelzőlámpák irányítják a járművek forgalmát. A lámpatestbe építve egymás alatt helyezkednek el a jelzőlámpák. Felülről lefelé haladva ezek piros, sárga, zöld színűek.

piros (állj!), piros és sárga együtt (állj, felkészülés a haladásra), zöld (mehetsz!).

Kapcsolási rajz:



Programkód:



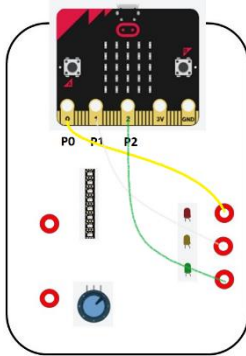
Eredmény tesztelése:

Piros - Piros/Sárga – Zöld – Sárga és ez ismétlődik.

6.feladat: Futófény

Terv: A LED-ek egymás után (200 ms) gyulladjanak fel.

Kapcsolási rajz:



Programkód:



Eredmény tesztelése:

Érdeemes változtatni a sorrenden pl. a zöld után ismét a sárga jelenjen meg.





Digitális bemenet

Ismét csatlakoztassuk a micro:bitet USB kábelen keresztül a számítógépünkhöz ez a feltétele, hogy majd le tudjuk tölteni a programunkat.

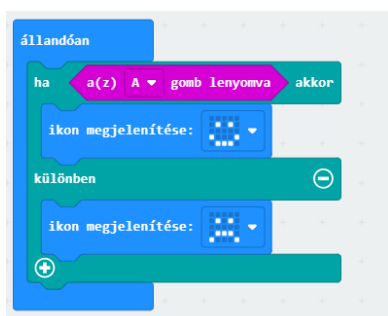
Újabb izgalmas rész következik, ahol a bemenetet vizsgáljuk, mégpedig azt, hogy az adott PIN-en van jel (3V3), vagy nincs jel (0V).

1.feladat: Digitális bemenet az „A” nyomógombon

A micro:bit tartalmaz négy kapcsolót A és B gomb a touch logo és hátul a reset gomb, most az A jelű gombot fogjuk használni.

Terv: ha az A gombot lenyomjuk akkor jelenjen meg a , ha nincs lenyomva akkor a  jelenjen meg. Valószínűleg már csináltuk is ezt a feladatot csak nem lett hangsúlyozva a digitális bemenet kifejezés.

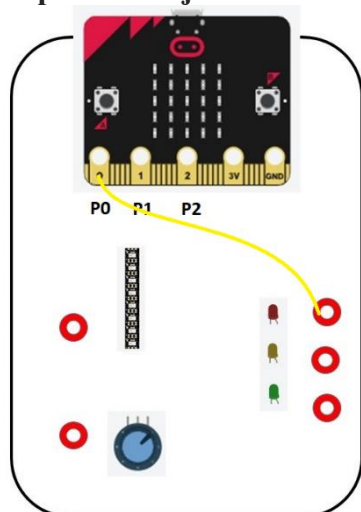
Programkód:



2.feladat: Az A és B nyomógombokkal vezérelt LED

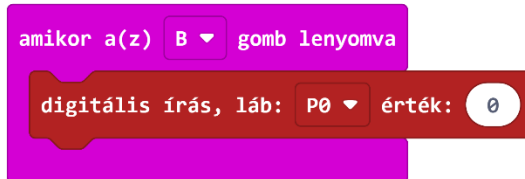
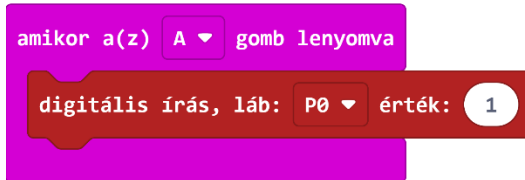
Terv: Az „A” gomb lenyomása után világítson a piros LED, a „B” gomb lenyomása után ne világítson a piros LED.

Kapcsolási rajz:



P0-át összekötjük a Piros Led-el

Programkód:



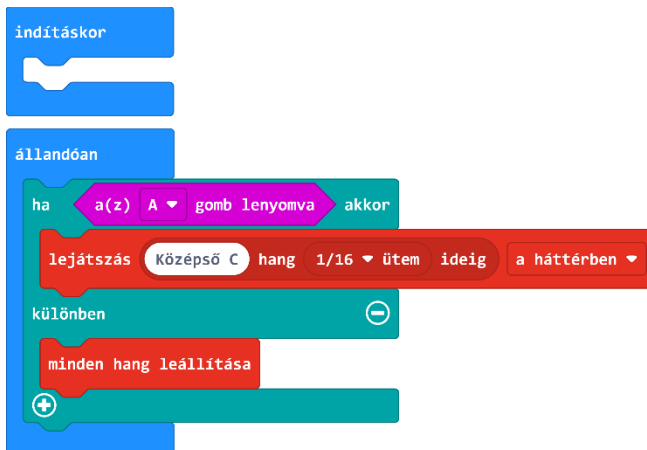
Eredmény tesztelése:

Érdeemes más kombinációt és több LED-et bevonni új feladatokba.

3.feladat: Csengő nyomógomb

Terv: Az „A” gomb lenyomása közben szólaljon meg egy hang, az elengedés után szűnjön meg a hang.

Programkód:



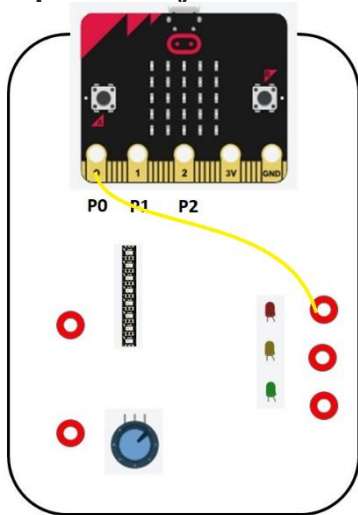
Eredmény tesztelése:

Érdeemes más kombinációt és több LED-et bevonni az új feladatokba.

4.feladat: Kapcsolás egy nyomógombbal

Terv: Az „A” gomb lenyomását követően világítson a piros LED, újbóli lenyomásra ne világítson.

Kapcsolási rajz:



P0-át összekötjük a Piros Led-el

Programkód:

```
indításkor
jel legyen 0

állandóan
ha a(z) A gomb lenyomva és jel = 0 akkor
  digitális írás, láb: P0 érték: 1
  jel legyen 1
+
ha a(z) A gomb lenyomva és jel = 1 akkor
  digitális írás, láb: P0 érték: 0
  jel legyen 0
+
```

Létrehozunk egy változót, ennek az értéke határozza meg, a kapcsolási sorrendet.

Első lenyomás „A” gomb lenyomva? = „igaz” és a jel értéke 0 következmény, hogy a LED világít, a változó értékét 1-re változtatjuk.

Második lenyomás a jel értéke 1, tehát a LED nem világít, a változót 0-ra állítjuk.

Logikailag jónak tűnik ez a megoldás. de ...

Eredmény tesztelése:

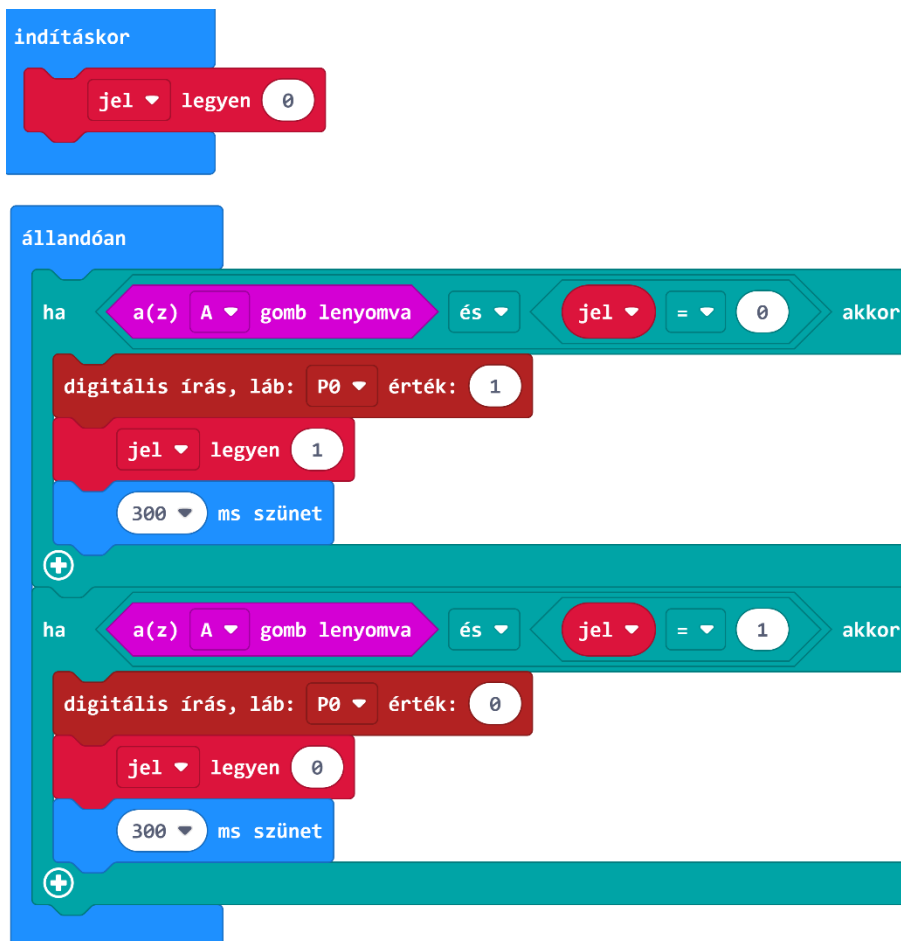
Az „A” gomb folyamatos lenyomásakor picit világít a piros LED, hát nem ezt vártuk.

Pergésmentesítés

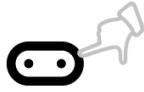
Ha a programban az állandóan ciklus gyorsan fut, akkor egy olyan nem kívánatos jelenség is felléphet, hogy egy gombnyomást **több gombnyomásnak detektál (túlszámol)** a program, pedig csak egyet szeretnénk. Ha nincs késleltetés a ciklusban, akkor többször beolvashatja a digitális bemenetet amíg le van nyomva a gomb. Egy valódi kapcsolónál ezen kívül egy másik jelenség is felléphet az ún. **pergés** vagy prellezés. A kapcsoló vagy nyomógomb belsejében található rugalmas mechanikai érintkezők az átkapcsolás során többször nekiütődhetnek és visszapatthanak a stabil kontaktus kialakulását megelőzően. Az pergés viszonylag gyorsan lezajlik általában 50 ms-on belül.

Lehetséges hardveresen és **szoftveresen** is pergésmentesíteni. A **pergésmentesítés** rész lényege, hogy a gomb **lenyomása után valamennyit késleltetni kell**, hogy a pergés jelenség megszűnjön és csak utána végezzük el az adott feladatot.

Kiegészítjük a programot „300 ms szünet” várakozással, így már jól működik a program.

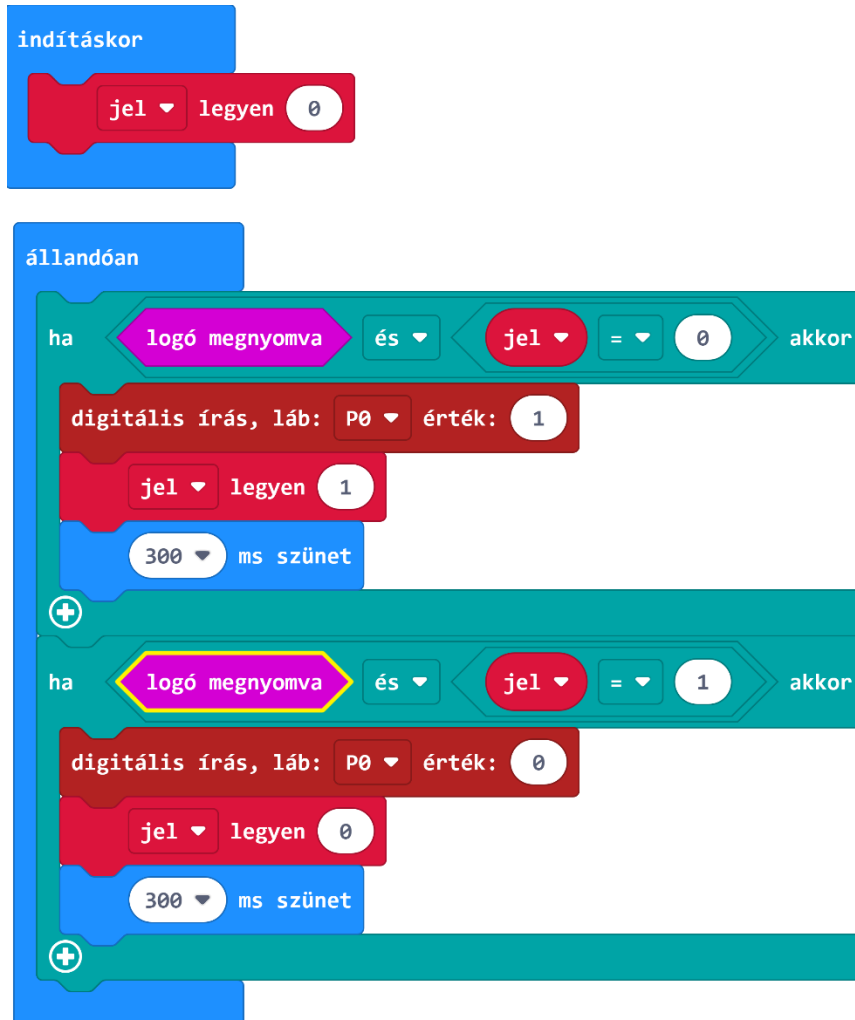


5.feladat: Kapcsolás



(touch logo) érintőkapcsolóval

Terv: Az touch logo érintését követően világítson a piros LED, újbóli lenyomásra ne világítson. (Az érintőkapcsolós megoldás gyakori az okos házakban, sőt egyre inkább sztenderd!)



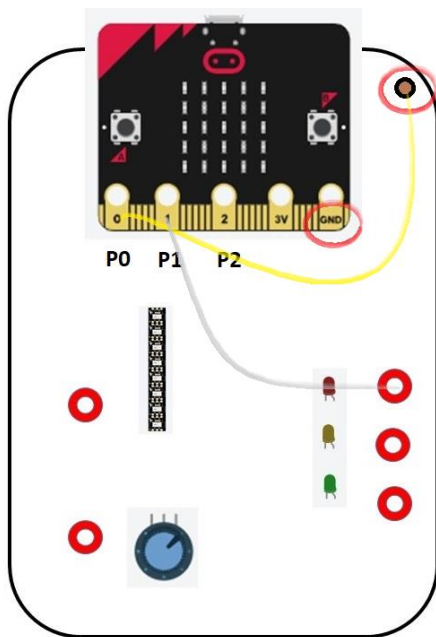
6.feladat: Kapcsolás a GND-vel

Terv: Azt vizsgáljuk, hogy csatlakozik-e az adott PIN a GND-hez, ha igen, akkor világítson a LED.

A GND a Ground rövidítése. Az elektronikus és elektromos áramkörökben az áram közös visszatérési útjára utal az elektromos forráshoz, és így lehetővé teszi az áramkör befejezését.

Ez a funkció akkor működik a legjobban, ha a micro:bit elemet használ az áramellátáshoz, az USB-kábel helyett.

Kapcsolási rajz:



Két ponton is el tudjuk érni a GND-t, az egyik a felirat és a hozzátartozó csavar a másik a micro:bittől jobbra található csavar (ide külön kivezettem, mert szükség lesz az érintős játék használatakor).

A Bemeneteknél találunk egy elég érdekes parancsot `a P0 láb lenyomva`, hogy lehet egy lábat lenyomni, érdekes a fordítás, helyesen így hangzik: földelt? Az a kérdés, hogy a kiválasztott láb érintkezik-e a GND (föld)-el.

Nagyon aranyos kísérlet amikor amikor a kapcsoló helyett egy osztálynyi tanuló fogja meg egymás kezét, mi pedig vizsgáljuk, hogy igaz-e az állítás

Programkód:



Eredmény tesztelése:

Ha P0-ra kötött banándugóval megérintjük a fent említett pontok egyikét a LED világít.

Ha bal kezünkkel megfogjuk a banándugót és jobb kezünkkel érintjük meg a pontok valamelyikét szintén világít a LED. Az emberi test kiválóan vezeti az elektromosságot, de lehet szemléltetni a vezető, nem vezető anyag fogalmát is.

Analóg kimenet

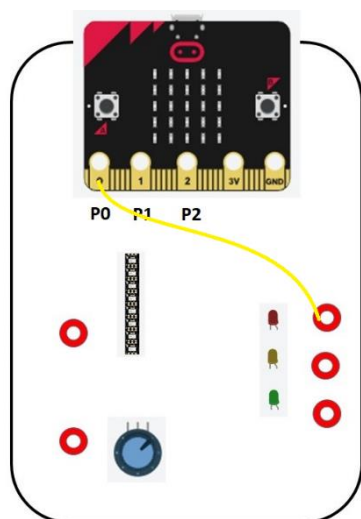
Az **analóg** komponensek fogadják a bemenetet vagy a kimenetet 0V-tól 3V3-ig terjedő feszültségtartományból. A mikrokontroller tartalmaz egy ADC-t (analóg-digitális átalakító), hogy ezeket az analóg jeleket a mikrokontroller által használható értékekké alakíthassa.

A mikro: bitnek 10 bites ADC-ja van, így az analóg bemenet 0 - 1023 tartományba kerül.

1. feladat: A piros LED fényerősségének változtatása

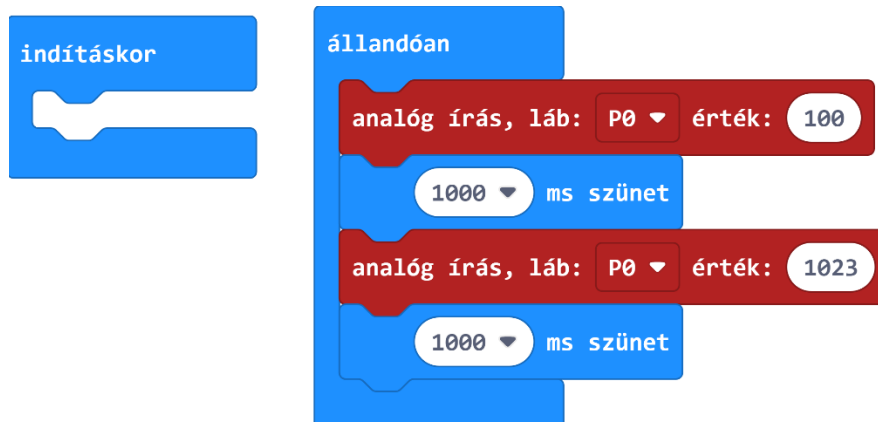
Terv: Programból változtatjuk az analóg kimenet értékét, ezzel befolyásoljuk a LED fényerősségét .

Kapcsolási rajz:



P0-át összekötjük a Piros Led-el

Programkód:



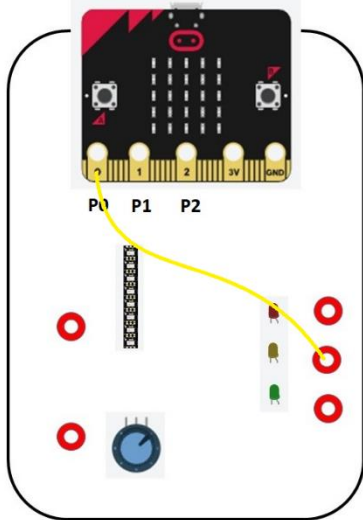
Eredmény tesztelése:

Változik a fényerősség, lélegzik a LED.

2. feladat: Gyertyafény imitálása

Terv: Változtassuk a sárga LED fényerősségét véletlenszerű értékkel.

Kapcsolási rajz:



Programkód:



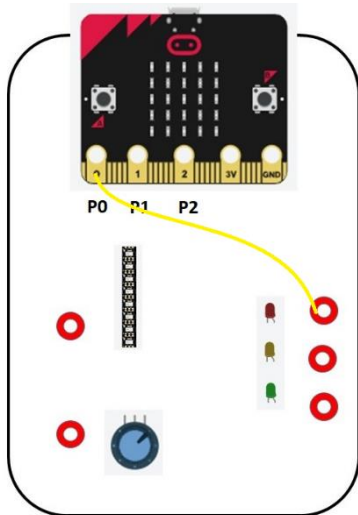
Eredmény tesztelése:

Változik a fényerősség, lélegzik a LED.

3. feladat: Erősödő – gyengülő fényhatás

Terv: Fokozatosan erősödjön majd gyengüljön a fényerősség.

Kapcsolási rajz:



Programkód:

```
indításkor  
  
állandóan  
ciklus index 0-tól 10 -ig  
ismételd  
  analóg írás, láb: P0 érték: index * 10  
  1000 ms szünet  
ciklus index 0-tól 10 -ig  
ismételd  
  analóg írás, láb: P0 érték: 10 - index * 10  
  1000 ms szünet
```

Eredmény tesztelése:

A beállított szünetnek megfelelően változik a fényerősség.

Analóg bemenet

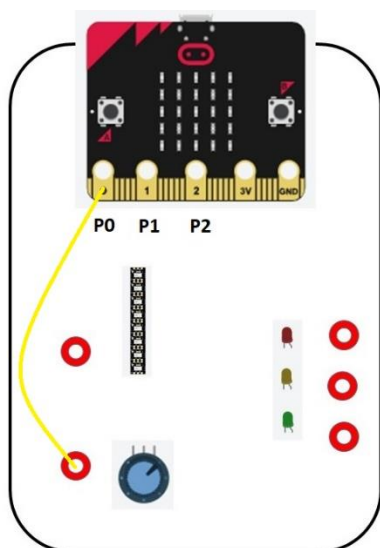
1. feladat: Forgásszög érzékelő értékeinek megjelenítése

A forgásszög érzékelő (potenciométer) egy változtatható feszültségű ellenállás (a készletben 10 k Ω). A belőle nyert változó adatokat fogjuk kiolvasni, a potenciométer tengelyének elfordításával az ellenállás értékét változtathatjuk a középső és mindkét szélső kivezetése között, ilyenkor a középső kivezetés feszültség értéke változik.

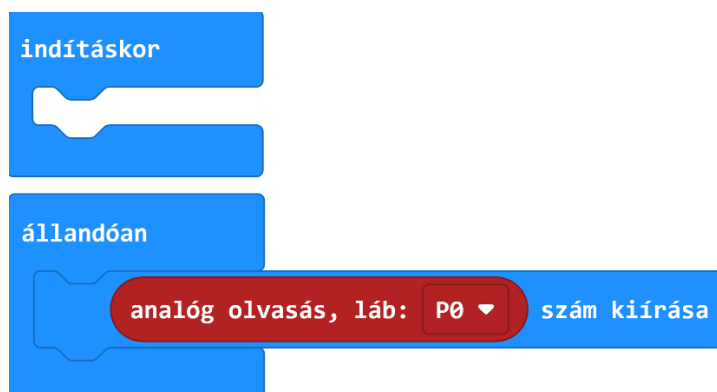
Az analóg bemenet kiolvasáshoz az analóg olvasás, utasítást fogjuk használni, ami egy számértéket ad vissza 0 és 1023 között (a használt feszültséghez arányosítva, pl. 3,3V = 1023; 0V = 0).

Terv: Megjelenítjük a micro:biten az analóg értékeket.

Kapcsolási rajz:



Programkód:



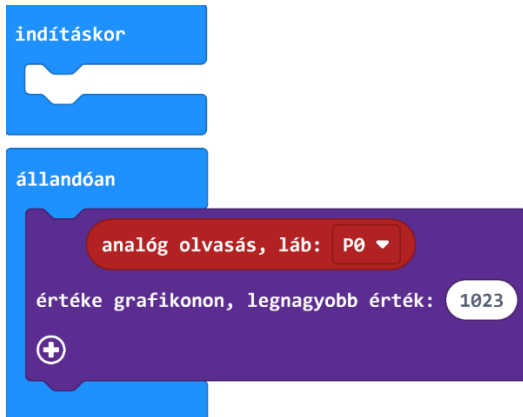
Eredmény tesztelése:

A mátrix kijelzőn olvashatók a mért értékek, látható hogy a forgatás hatására változnak az értékek. Tessék lassan tekerni a potenciométert, mert a kiírás sajnos lassan megy (pl. ezért használunk OLED vagy LCD kijelzőket). Olvassuk le a legkisebb és legnagyobb értéket.

2. feladat: Forgásszög érzékelő értékeinek grafikus megjelenítése

Terv: A 0 és 1023 közötti érték grafikus megjelenítése

Programkód:



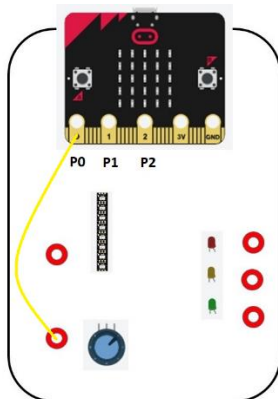
Eredmény tesztelése:

Nagyon látványos és gyors megjelenítés.

3. feladat: Hangmagasság szabályozás potenciométerrel

Terv: A forgatás során különböző értékeket kapunk (0 és 1023 között) ezt használjuk a hangmagasság beállításához.

Kapcsolási rajz:



Programkód:

a.)



b.)

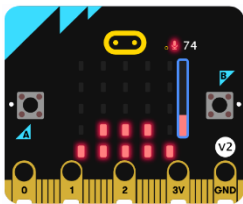


Eredmény tesztelése:

A forgásszög érzékelő elforgatásával változik a hangmagasság.

4. feladat: Hangerősség (mikrofon) értékeinek grafikus megjelenítése

Terv:



Ha használom a hangerő utasítást a mikrofon automatikusan bekapcsol, szolgáltat egy analóg értéket (0-255) amely nagysága a hangerősség nagyságától függ.

Természetesen az értéket is ki lehetne írni, de a többjegyű számok megjelenítése sokáig tart, így a grafikus megjelenés látványosabb.

Programkód:



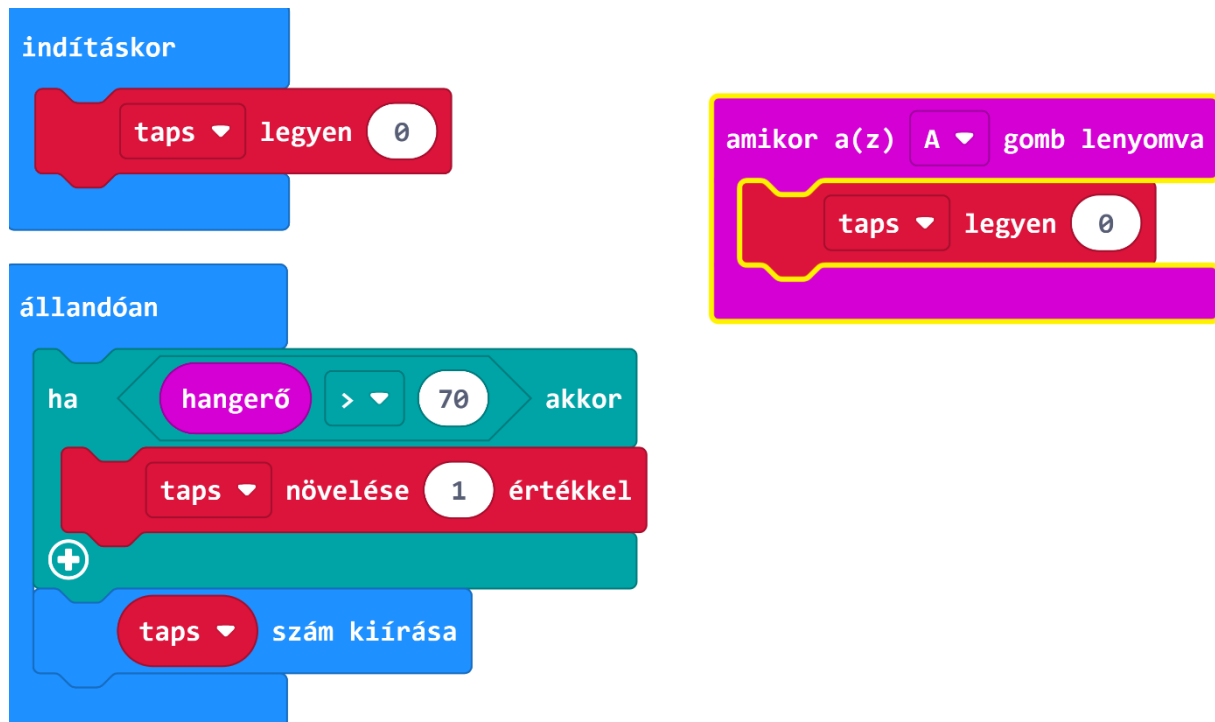
Eredmény tesztelése:

Nagyon látványos ekva lézer (Equalizer)

5. feladat: Tapsolás számláló

Terv: Szeretném megszámolni a tapsolások számát (nagyobb hangerő érték!)

Programkód:



Hozzunk létre egy taps nevű változót, amelynek értékét majd megjelenítjük

Eredmény tesztelése:

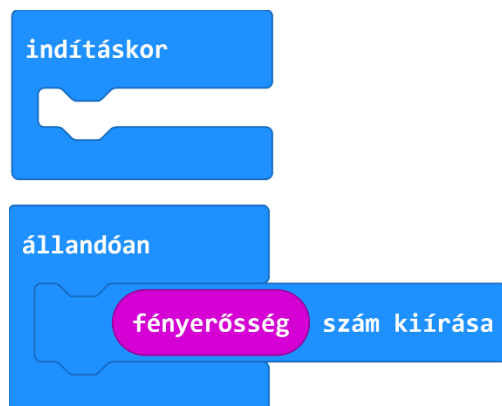
Nálam a 70-es értékkel működött, természetesen ez nagyban függ a környezeti zajtól.

6. feladat: Fényerősség megjelenítése

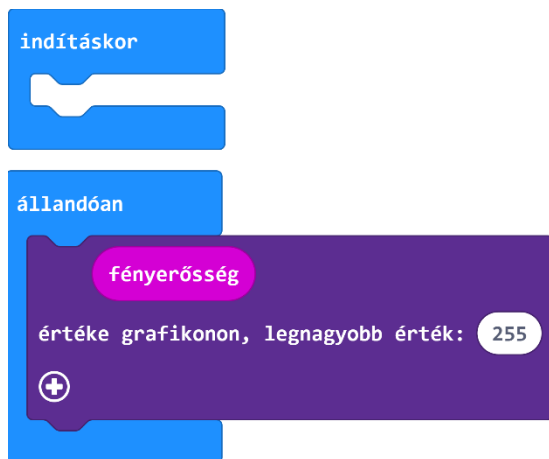
Terv: A micro:bit képes mérni a fényerősséget, nem túl pontos, az 5*5-ös LED mátrixon történik a mérés, tehát takarjuk le ill. világítsuk meg telefontal és vizsgáljuk meg az értékeket.

Programkód:

a.)



b.)



Eredmény tesztelése:

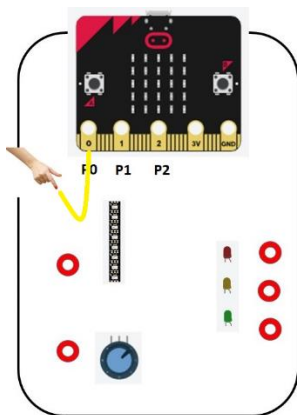
Általában szükség van egy erősebb fényforrásra, a LED-el történő fényérés ennyit tud, ha pontosabban szeretnénk akkor célszerű fényerősség szenzort használni.

7. feladat: Kapacitív vagy rezisztív?

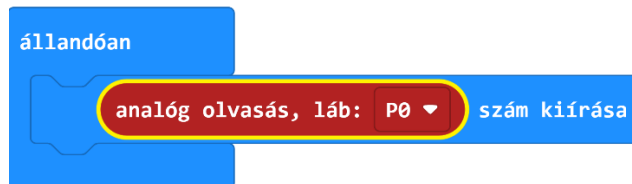
Terv: Megjelenítem a P0 analóg értékét (ahol is 0-át várok!), majd megérintem a kezemmel és figyelem, hogy lesz-e változás.

- **A kapacitív** érintésérzékelés az, amikor a micro:bit észleli a PIN vagy a vezető felület (például az érintéslogó) kapacitásának változását, amikor az ujjad hozzáér vagy nagyon közel kerül hozzá.
- **A rezisztív érzékelés** érzékeli az áram áramlását a GND érintkezőhöz. A testednek csatlakoznia kell a GND-hez (akár megérintésével, akár egy vezeték csatlakoztatásával), miközben megérinti a bemeneti érintkezőt, hogy létrejöjjön egy áramkör a két érintkező között.

Kapcsolási rajz:



Programkód:



Eredmény tesztelése:

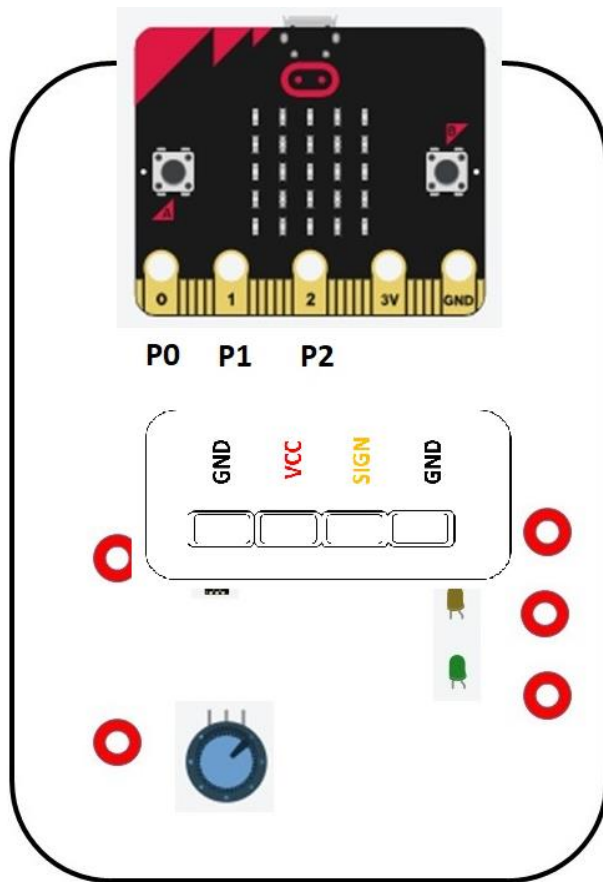
Hát nem 0-át mutat a kijelző, nálam ~645 és ha megérintem 1023 (ez nagyban változhat!), újabb lehetőség ez a kapacitív kapcsoló létrehozására.



Azt gondolom, hogy átvettük a **digitális** ki és bemenet, az **analóg** ki és bemenet témákat.

Ezen ismeretek birtokában nagyon komoly kísérleteket végezhetünk, és komoly **alkalmazásokat** építhetünk. Nem okoz majd gondot a kapható robotok működésének és programozásának megértése.

A 4-es csatlakozó:



A kísérletező készletünk tartalmaz egy négyes csatlakozási pontot is, amelyhez tetszőlegesen rendelhető (csatlakoztatható) a PIN0, PIN1 vagy PIN2.

A csatlakoztatási pont lábkiosztása a képről leolvasható, elsősorban úgy lett kialakítva, hogy csatlakoztatható legyen a neopixel sor (a készlet eleme), de nyilván szenzorok vagy pl. szervó motor (3V!) is csatlakoztatható.

A **Neopixel** az Adafruit nevű elektronikai vállalat által használt név és termék, egy olyan típusú RGB LED számára, amelyet különböző módon csomagolnak (alak és a ledék száma).



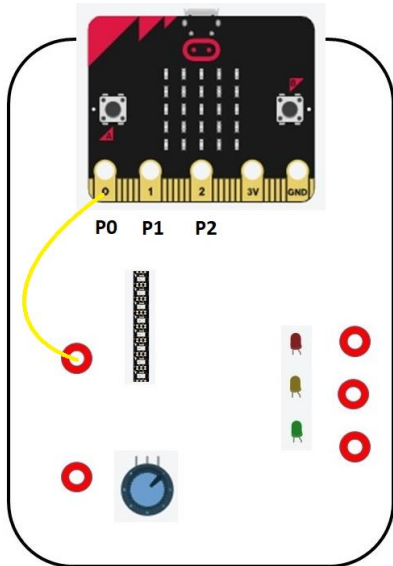
Közvetlenül a micro:bitről 8 LED-et tudsz táplálni, de ennél többet is lehet vezérelni (célszerű külső áramforrást igénybe venni). Úgy tervezték, hogy 5V-n használjuk, de működtethető 3V3 (3,3 V) mikrokontrollerrel is, mint a micro:bit.

Legérdekesebb jellemzője az egyetlen IO vezérlés (egyetlen vezetéken küldöm azt a jelet, amely meghatározza, hogy melyik vagy melyik és milyen színnel jelenjenek meg) és a végtelen kaszkádcsatlakozás (szinte végtelen sok elem összefűzhető).

1. feladat: Kapcsoljuk be az RGB ledek

Terv: Az A gomb megnyomása után világítson piros fényel, a B gomb megnyomása után ne világítson.

Kapcsolási rajz:



Csatlakoztassuk pl. a 0-s PIN-t, de mint fentebb írtam bármelyik is lehet, csak akkor a programban arra kell hivatkozni.

Az eszköz használatához szükségünk van a Neopixel kiterjesztésre!

Programkód:

```
indításkor
strip legyen NeoPixel at pin P0 with 8 leds as RGB (GRB format)

amikor a(z) A gomb lenyomva
strip show color red
strip show

amikor a(z) B gomb lenyomva
strip show color black
strip show

állandóan
```

Automatikusan létrehoz egy strip (szalag, sáv) nevű változót amit meg is változtathatunk, be kell állítani a kapcsolódó PIN-t és a ledek számát.

Érdekes a fekete szín, ezzel kapcsoljuk le a világítást.

Eredmény tesztelése:

Rendkívül erős a LED-ek fénye, célszerű használni a fényerősség utasítást.

```
indításkor
strip legyen NeoPixel at pin P0 with 8 leds as RGB (GRB format)
strip set brightness 30
```

2. feladat: Rendőrségi villogó

Terv: A piros kék felváltva villog.

Programkód:

```
indításkor
strip legyen NeoPixel at pin P0 with 8 leds as RGB (GRB format)
strip set brightness 30

állandóan
strip show color red
strip show
200 ms szünet
strip show color blue
strip show
200 ms szünet
```

3. feladat: Irányjelző (mai autókon)

```
indításkor
strip legyen NeoPixel at pin P0 with 8 leds as RGB (GRB format)
strip set brightness 30

állandóan
ciklus index 0-tól 7 -ig
strip set pixel color at index to orange
strip show
75 ms szünet
strip clear
strip show
100 ms szünet
```

4. feladat: Oda vissza futófény

indításkor

- strip legyen NeoPixel at pin P0 with 8 leds as RGB (GRB format)
- strip set brightness 30

állandóan

ciklus index 0-tól 7 -ig

ismételd

- strip set pixel color at index to blue
- strip show
- 50 ms szünet
- strip clear
- strip show
- 50 ms szünet

ciklus index 0-tól 7 -ig

ismételd

- strip set pixel color at 7 - index to blue
- strip show
- 50 ms szünet
- strip clear
- strip show
- 50 ms szünet

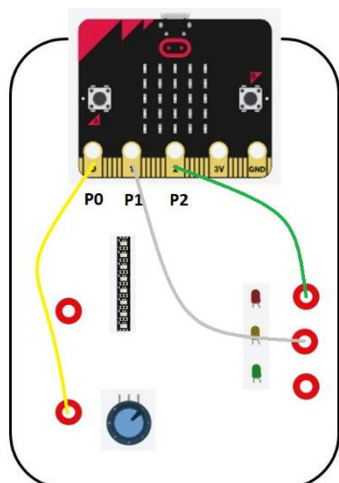
Ebből a néhány példából is látható, hogy nagyon sok érdekes feladatot lehet létrehozni.

Vegyes feladatok

1. feladat: Két LED villogás sebességének szabályozása forgásszög érzékelővel

Terv: Piros és sárga LED-ek felváltva világítanak és ennek a sebességét szabályozzuk a forgásszög érzékelővel.

Kapcsolási rajz:

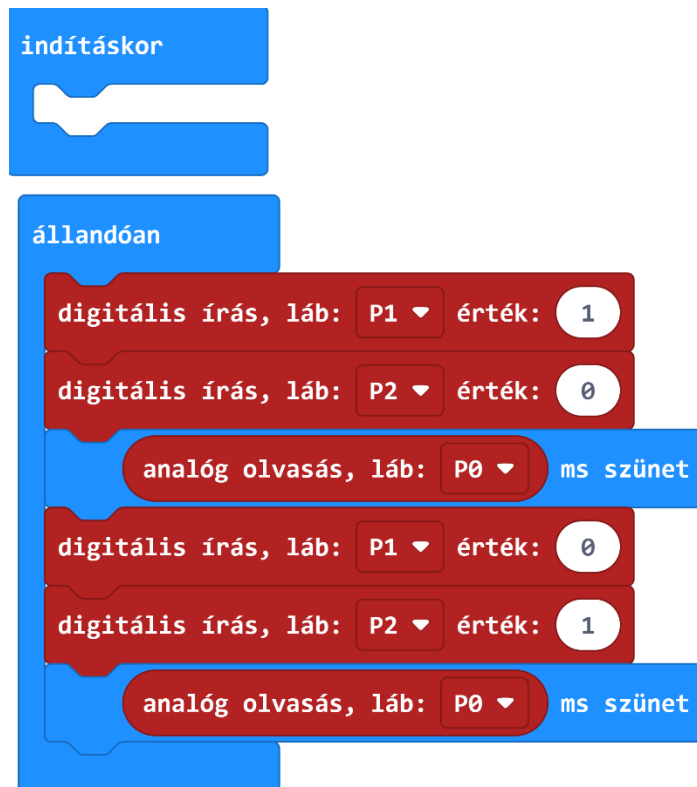


P0 - forgásszögérzékelő

P1 – sárga LED

P2 . piros LED

Programkód:



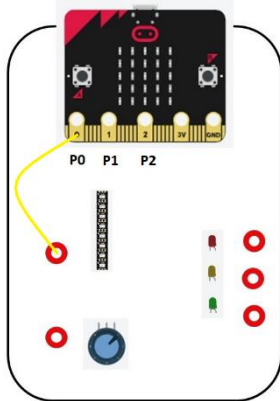
Eredmény tesztelése:

A szünet értékének változtatásával (0-1023!) befolyásoljuk a villogás sebességét.

2. feladat: Alkonyat kapcsoló (utcai világítás)

Terv: Fényerősség bizonyos értékénél kapcsoljon be a világítás.

Kapcsolási rajz:



Csatlakoztassuk a neopixelt!

Programkód:

```
indításkor
  strip legyen NeoPixel at pin P0 with 8 leds as RGB (GRB format)

állandóan
  fényerősség
  értéke grafikonon, legnagyobb érték: 255
  ha fényerősség < 150 akkor
    strip show color white
    strip show
  különben
    strip show color black
    strip show
```

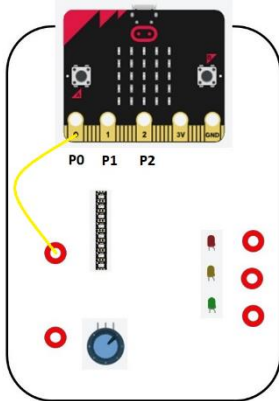
Eredmény tesztelése:

A nappali fény értékének elérése céljából használjuk a telefon lámpáját vagy vigyük közel az ablakhoz.

3. feladat: Taps kapcsoló

Terv: Ha tapsolunk (páratlan) világítson a lámpa, újabb taps esetén ne világítson a lámpa (páros).

Kapcsolási rajz:



Csatlakoztassuk a neopixelt!

Programkód:

```
indításkor
strip legyen NeoPixel at pin P0 with 8 leds as RGB (GRB format)
szam legyen 0

állandóan
  hangerő
  értéke grafikonon, legnagyobb érték: 255
  ha hangerő > 120 akkor
    szam növelése 1 értékkel
  ha szam / 2 maradéka = 1 akkor
    hívás Lámpa fel
  különben
    hívás Lámpa le

függvény Lámpa fel
  strip show color white
  strip show
  500 ms szünet

függvény Lámpa le
  strip show color black
  strip show
  500 ms szünet
```

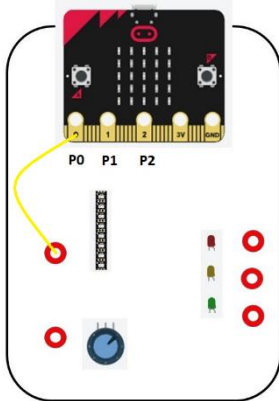
Eredmény tesztelése:

A hangerősség küszöb számát valószínűleg kell majd módosítani, nálam ezzel működött, de hát meghatározó a környezet.

4. feladat: Futófény sebességének szabályozása forgásszög érzékkel.

Terv: Az elforgatás mértékével változzon a világító LED mozgásának a sebessége. (Vigyázat, nagyon nagy volt a sikere a gyerekek körében, akár függőséget is okozhat!)

Kapcsolási rajz:



Csatlakoztassuk a neopixelt!

Programkód:

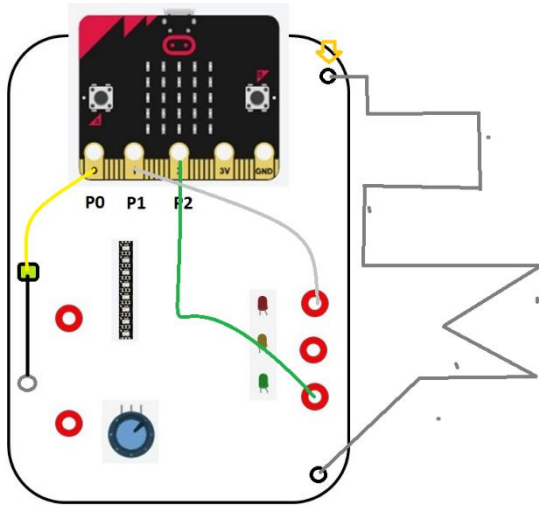
```
indításkor
  strip legyen NeoPixel at pin P0 with 8 leds as RGB (GRB format)
  strip set brightness 30

állandóan
  szün legyen analóg olvasás, láb: P1
  ciklus index 0-tól 7 -ig
    ismételd
      strip set pixel color at index to blue
      strip show
      szün ms szünet
      strip clear
      strip show
      szün ms szünet
  ciklus index 0-tól 7 -ig
    ismételd
      strip set pixel color at 7 - index to blue
      strip show
      szün ms szünet
      strip clear
      strip show
      szün ms szünet
```

5. feladat: Érintős játék hibaszámlálóval.

Terv: Az ismert érintős játék ahol is számoljuk az érintések számát ami maximum 9 lehet.

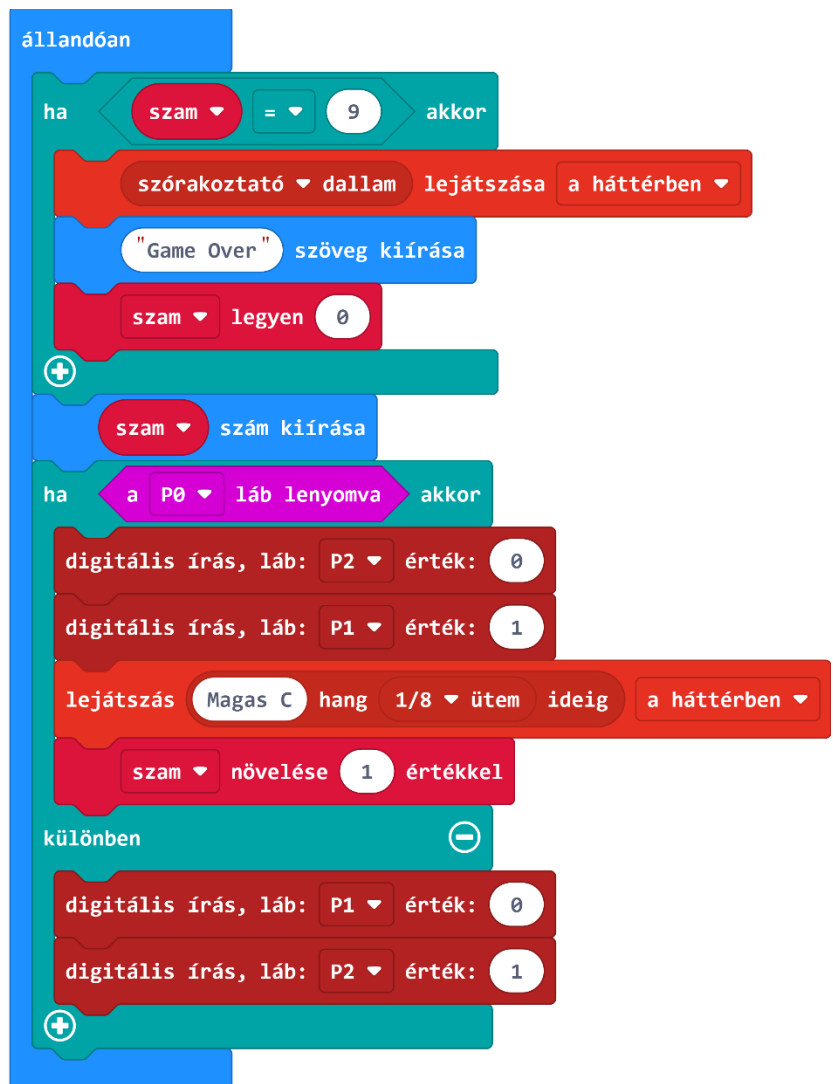
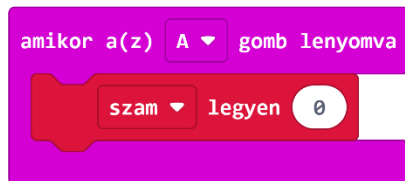
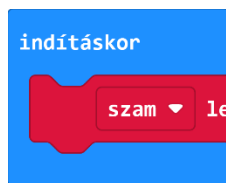
Kapcsolási rajz:



A jobb oldali, alátéttel ellátott csavarokhoz rögzítsük az általunk megálmodott (célszerű lapos vagy kombinált fogót használni) pályát a sárga nyíl helyén csatlakozik a GND-vel.

A P0-ra kötjük (csatlakoztatjuk) a karikával ellátott érzékelőt.

P1 és P2 az értékelő fények a zöld folyamatosan világít ill. hiba esetén a piros váltja.



Eredmény tesztelése:

Érintkezéskor hangjelzés, és a piros LED világít. Számoljuk az érintéseket, ha eléri a kilencet (a tíz megjelenése annyira lelassítaná) vége a játéknak.

Az A gomb megnyomásával lehet nullázni, az újratekéshez.

TO BE CONTINUED...

Sok sikert kívánok, és ha valami problémába ütköztünk, vagy csak megszeretnétek osztani az elkészült projekteteket forduljatok bizalommal hozzám:

vigvari.gyorgy@vigvari.hu e-mailon.