

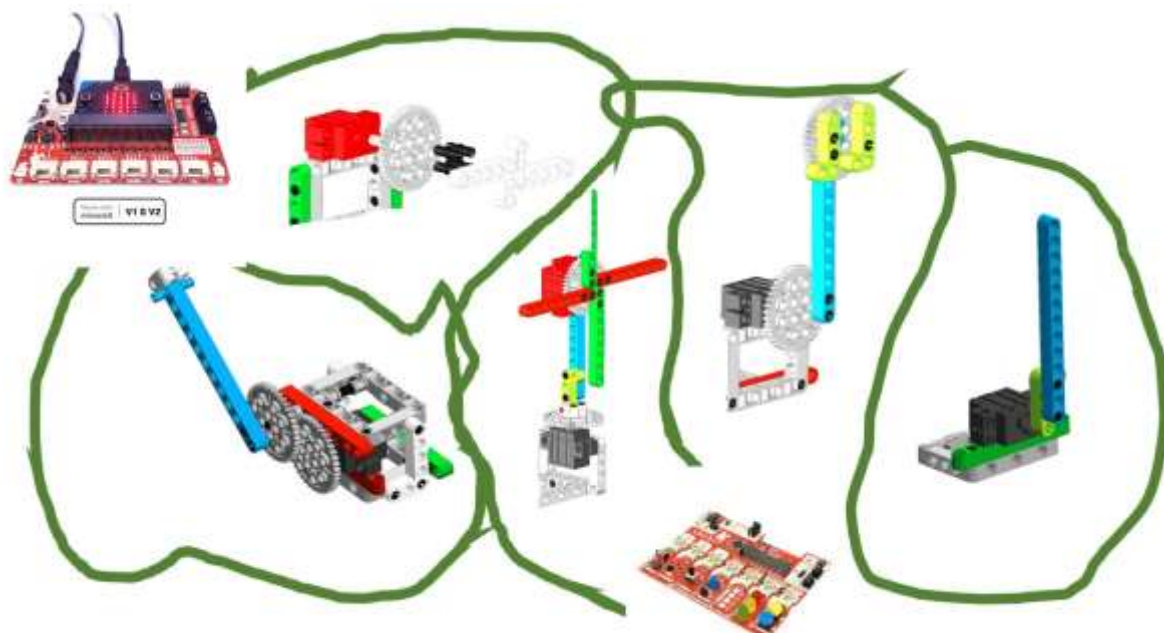
AVR Projekt építőkészlet (lego kompatibilis)

Kezelési utasítás és építési segédlet

Kezelési utasítás és építési segédlet

Ez a készlet elsősorban a reka:bithez lett optimalizálva, de természetesen jól lehet használni más eszközökkel is.

A reka:bit a Edu:bit kistestvére, sokkal kisebb tábla mérete tökéletes az iskolák és versenyek innovációs projektjeinek építéséhez.



Tartalom:

<i>A Reka:bit adatlap</i>	<i>2</i>
<i>A fények és jelentésük</i>	<i>3</i>
<i>NeoPixel 1 x 8 RGB LED</i>	<i>5</i>
<i>A DC motorok vezérlése.....</i>	<i>8</i>
<i>A szervo motorok vezérlése.....</i>	<i>12</i>
<i>Sorompó vezérlése.....</i>	<i>15</i>
<i>Állítható sebességű ventilátor</i>	<i>19</i>
<i>Állítható forgásirányú ventilátor.....</i>	<i>22</i>
<i>Forgásszög mérése.....</i>	<i>25</i>
<i>Távolságmérés.....</i>	<i>27</i>
<i>Hangszer készítése.....</i>	<i>25</i>
<i>Zene_Metronóm</i>	<i>31</i>
<i>Hajítógép.....</i>	<i>35</i>

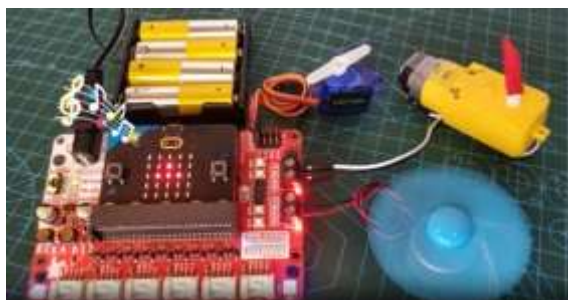
A Reka:bit adatlap:

A **REKA:BIT** az EDU:BIT kistestvére, sokkal kisebb táblamérete tökéletes iskolák és versenyek innovációs projektjeinek építésére.



Emellett a **REKA:BIT** be-/kikapcsolás jelzővel is rendelkezik, valamint beépített alacsonyfeszültség és túlfeszültségi LED-ekkel is rendelkezik, amelyek megfelelő figyelmeztetést adnak, ha bármilyen probléma adódna a tápellátással. A tápellátás történhet a számítógép USB csatlakozójából, de lehet hordozható is a 4db elemmel az elemtartóból.

A **REKA:BIT** társprocesszorral rendelkezik a többfeladatos feladatok hatékonyabb kezeléséhez. A **REKA:BIT** számára nem jelent problémát a zenelejátszás akár 4x szervomotor és 2x egyenáramú motor vezérlése közben, a micro:bit LED mátrix animálása és akár az RGB LED-ek különböző színekben történő megvilágítása is.



Ez a képesség már nagyon komoly projektek megvalósítására is alkalmas, de hogy egyszerűbb legyen a tervezés és megvalósítás, létrehoztunk egy lego kompatibilis néhány elemből álló készletet, amihez ötleteket is biztosítunk ebben a jegyzetben.



A reka:bit csatlakoztatható a 16*16-os alaplaphoz és marad hely építeni, de azt is lehet, hogy külön építünk.

1. A fények és jelentésük.

Csatlakoztassuk a reka:bitünket a számítógéphez, abban a pillanatban kigyulladnak bizonyos fények:



Információ a tápellátás erősségéről.

Micsoda segítség lehet ez, ha majd merülnek az elemek!



A kék led-ek a használt PIN állapotár mutatja, amelyek segítik a kódolást és a hibaelhárítást.

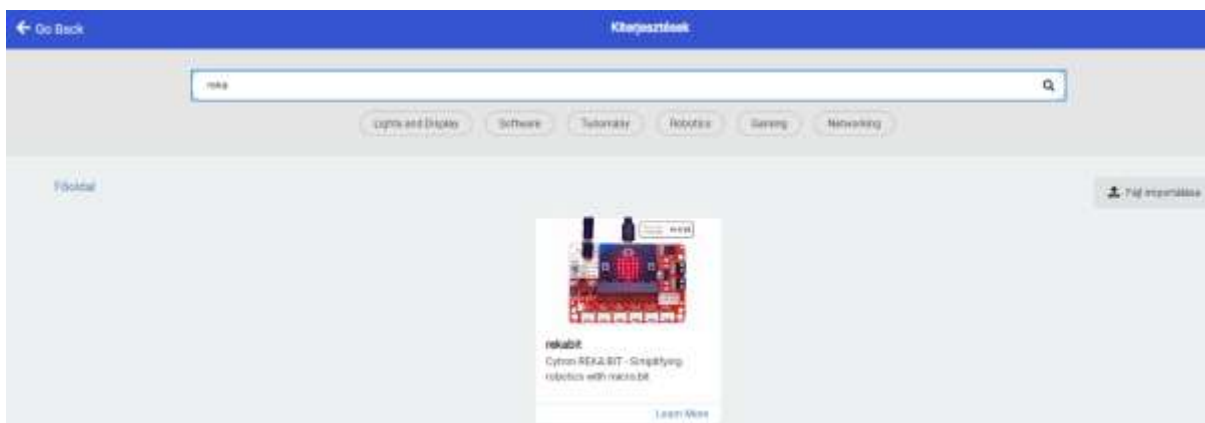
A piros led a motor meghajtóról szolgáltat információt, ebből négy darab van.

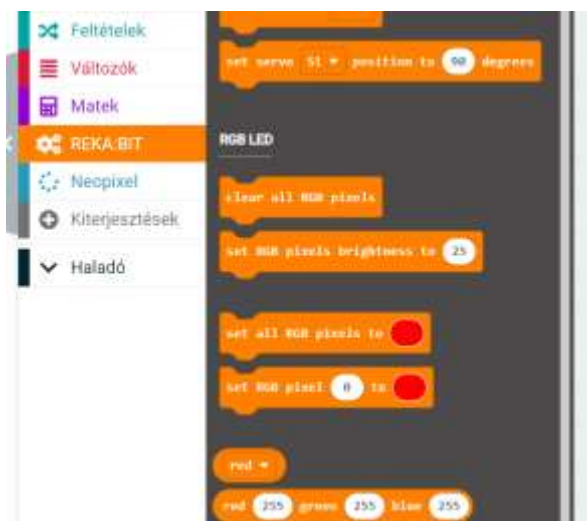
Ezeknek a LED-eknek a figyelésével könnyen ellenőrizhető a program és az áramkör kapcsolata.

Van még két szabadon felhasználható led-ünk (neopixel, zip, intelligens), ezek alapértelmezésben nem világítanak, de programozhatók, később értékes elemei lehetnek projektjeinknek.



Ha szeretnénk használni akkor, indítsuk el a makecode szerkesztőt és adjuk hozzá a reka:bit kiterjesztést.





Egyszerűbbé teszi a kódolást ez a kiterjesztés.

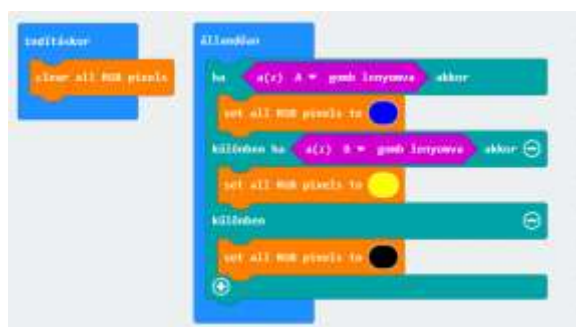
Amit szeretnénk használni az RGB LED-ek.



Indításkor töröljük a led-eket, majd villogtatjuk a zöld és piros színt.

(Fontos: a reka:bit legyen bekapcsolva, mert különben nem világít az RGB led!)

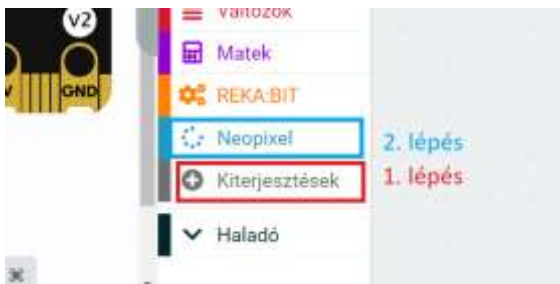
A későbbiekben majd látni fogjuk, hogy csatlakoztatható több neopixel led is, és azt is tudjuk vezérelni.



Ha az A gomb akkor kéken világít, ha B gomb akkor sárga.

NeoPixel 1 x 8 RGB LED

A készlet tartalmaz egy neopixel ledsort amit kétoldalas ragasztóval rögzítettem egy lego elemre *(természetesen ragaszthatjuk máshova is!)*



Ha szeretnénk használni az eszközt, akkor +Kiterjesztések és válasszuk a Neopixel bővítményt (a reka:bit parancsai csak a saját ledjeit képesek vezérelni)

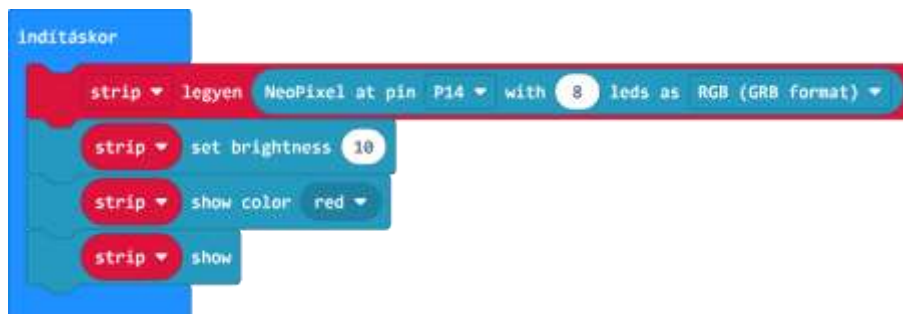
Csatlakoztassuk a P14-re , a strip nevű változót hagyhatjuk, de át is nevezhetjük.



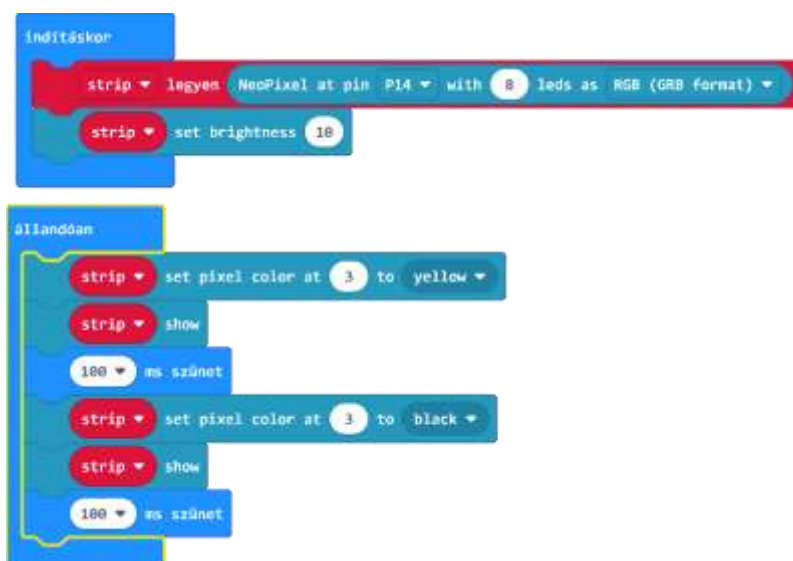
Felkapcsoljuk az összes ledet piros színnel, rettenetes élmény, nagyon erős a fény,



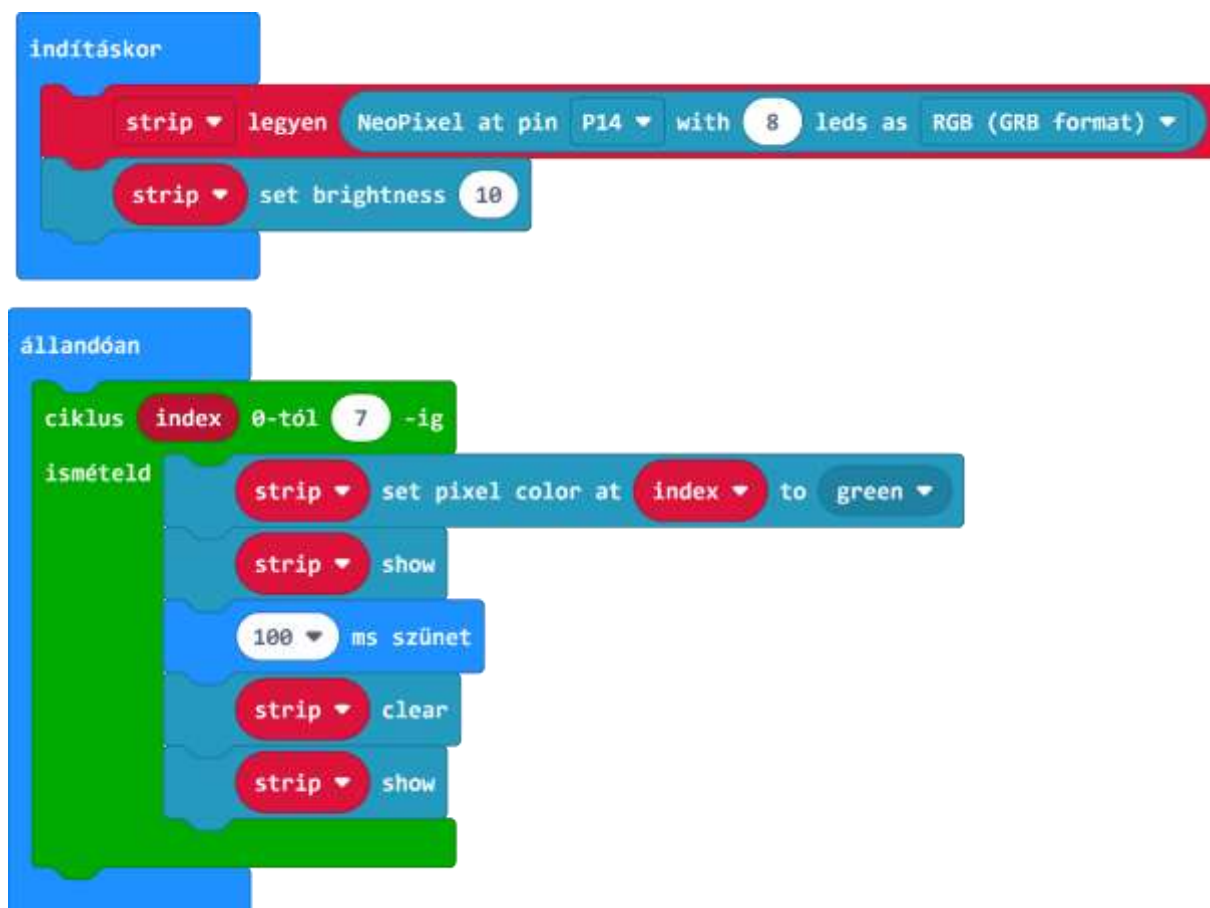
ezért csökkentjük a fényerősséget.



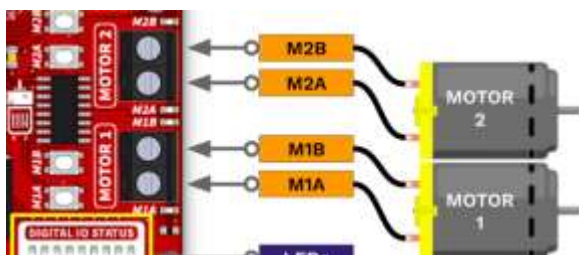
Természetesen lehet a 8 ledet akár külön egyesével is címezni:



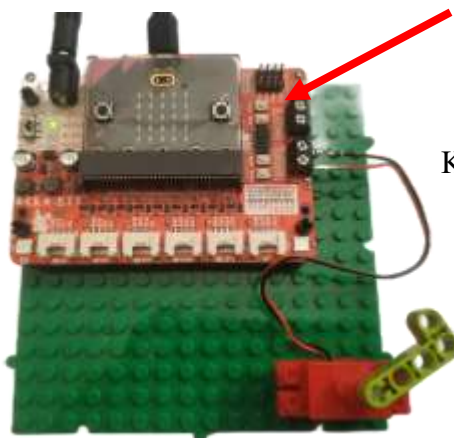
Egy futófény, nagyon sok lehetőség rejlik az eszközben és feldobja a projektjeinket.



2. A DC motorok vezérlése.

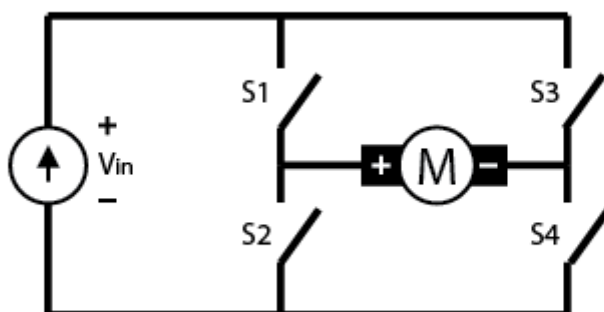


Kettő darab DC (egyenáramú) motort tudunk vezérelni (meghajtani) lehet kézi vezérléssel ill. programból is

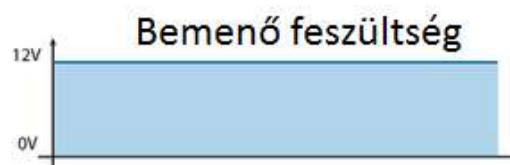


Kézi vezérlés esetén a két csatlakozóhoz van négy darab gomb amelyekkel a forgásiránya állítható be. Tartozik négy darab led is, amely mutatja a forgásirányt. Ha csatlakoztattuk a motort akkor ezeket csak meg kell nyomni.

A forgásirány szabályozásához csak a motoron át vezetett áram irányát kell megváltoztatnunk, és ennek leggyakoribb módja a H-híd használata. A H-Bridge-áramkör négy kapcsolóelemet, tranzisztort vagy MOSFET-et tartalmaz. Ha két külön kapcsolót aktiválunk egyszerre, megváltoztathatjuk az áram irányát, ezáltal megváltoztathatjuk a motor forgási irányát is.

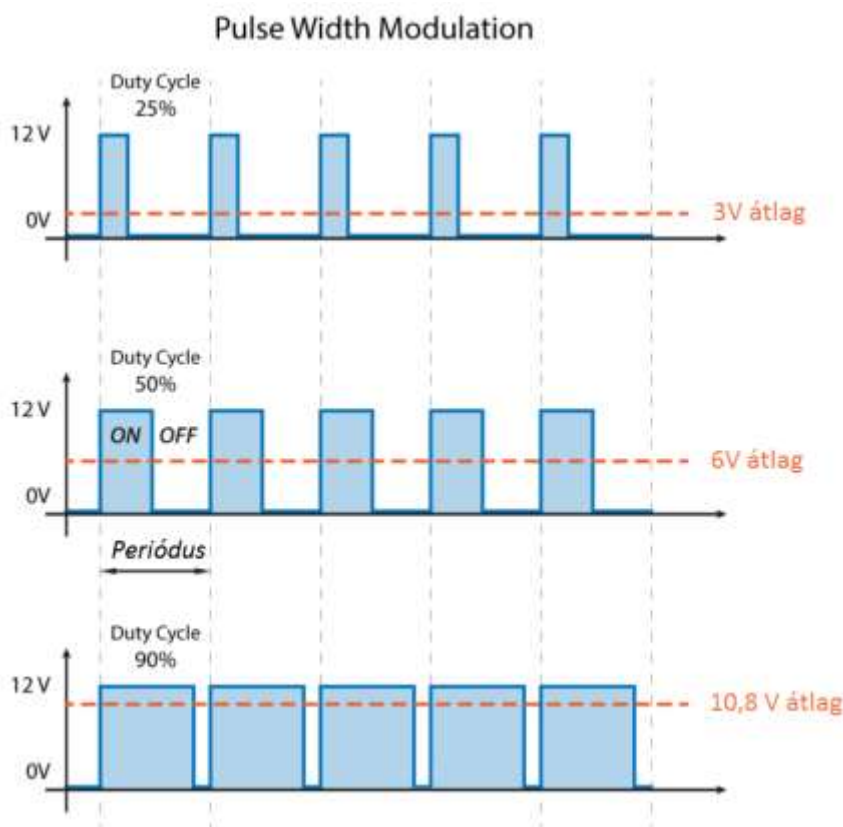


Egy DC motor sebességét/fordulatszámát egyszerűen szabályozhatjuk úgy, hogy a bemenetén változtatjuk a bemeneti feszültséget, valamint PWM jel segítségével. De mi is az a PWM jel?



PWM DC motor vezérlés

A PWM (Pulse Width Modulation) vagy másképp nevezve impulzus szélesség moduláció olyan technika, amely lehetővé teszi számunkra, hogy az elektromos eszközhöz tartozó feszültség átlagértékét gyors sebességgel bekapcsolva és kikapcsolva állítsuk be. Az átlagos feszültség függ a működési ciklustól (Duty Cycle), vagy attól az időtartamtól, ameddig a jel be van kapcsolva, szemben a jel kikapcsolásával egy periódus alatt.



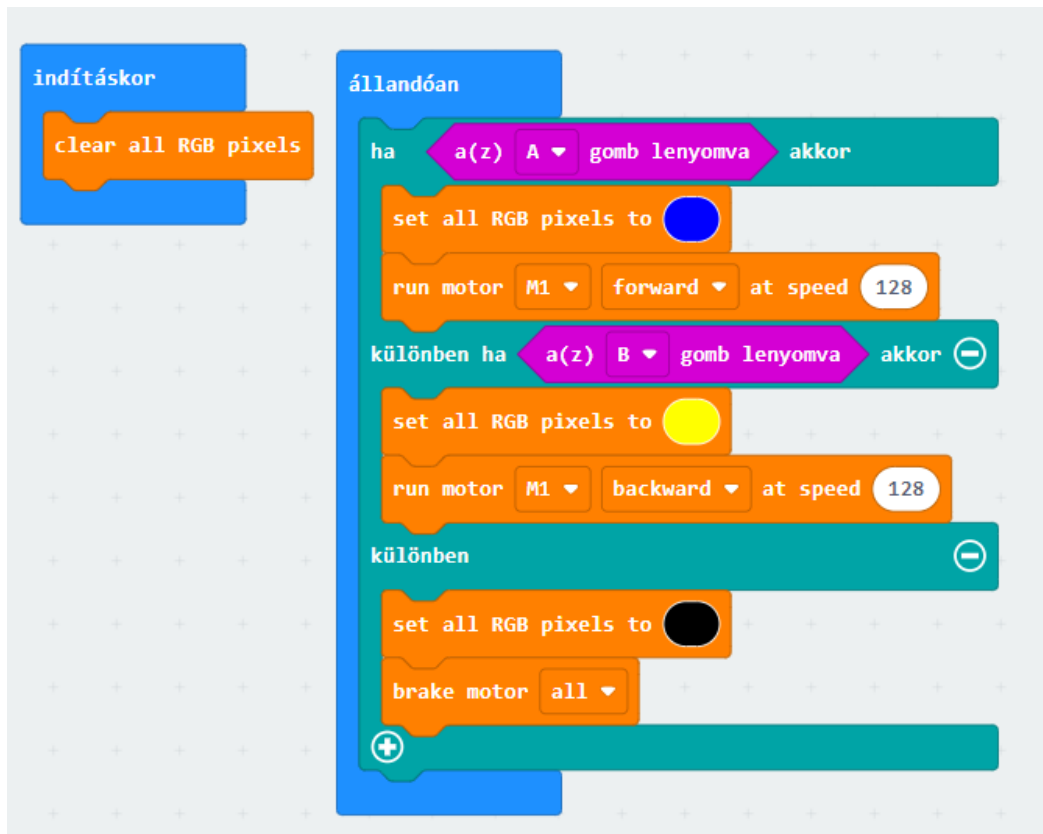
Tehát ha ezt a két metódust (PWM, H-híd) kombinálni tudjuk, akkor teljes uralmunk lesz a DC motor felett. A reka:bit biztosítja ezt a lehetőséget.



Az első utasítás leállítja a kívánt motort, vagy mind a kettőt.

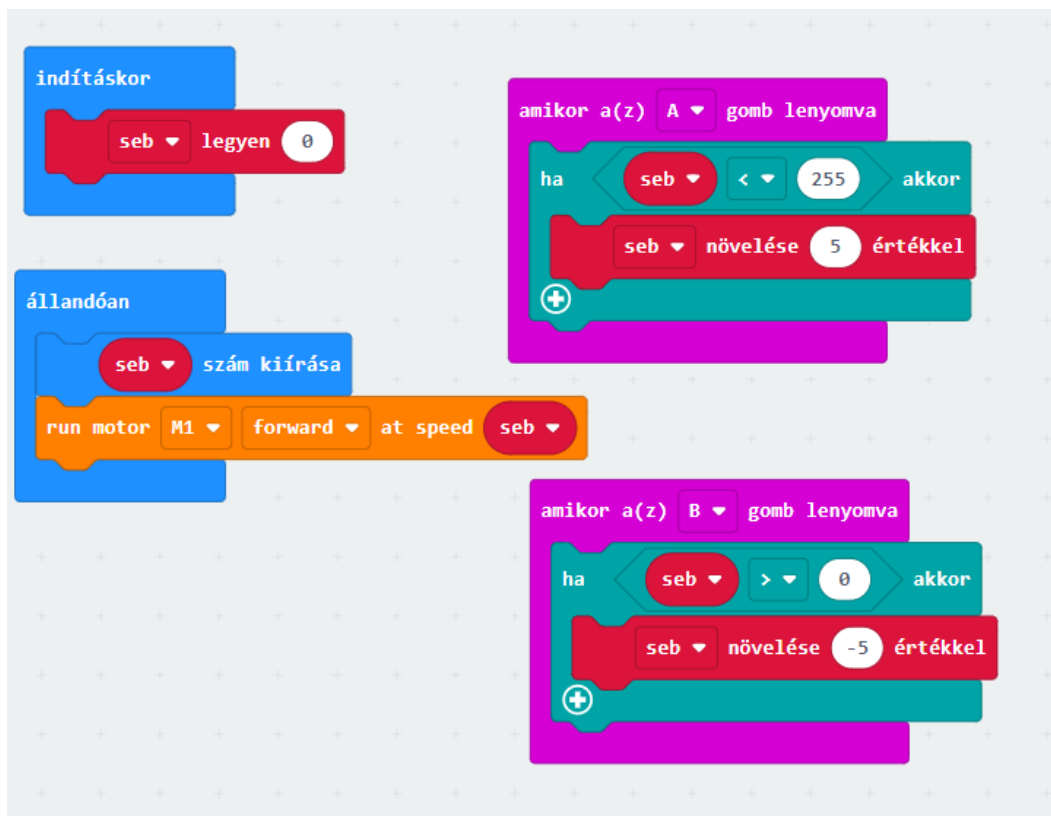
A második utasítás elindítja, ahol is lehet változtatni a forgásirányt ill. a fogás sebességét is.

A led kezelésére megírt alkalmazást bővítsük ki a motor vezérlésére szánt utasításokkal:

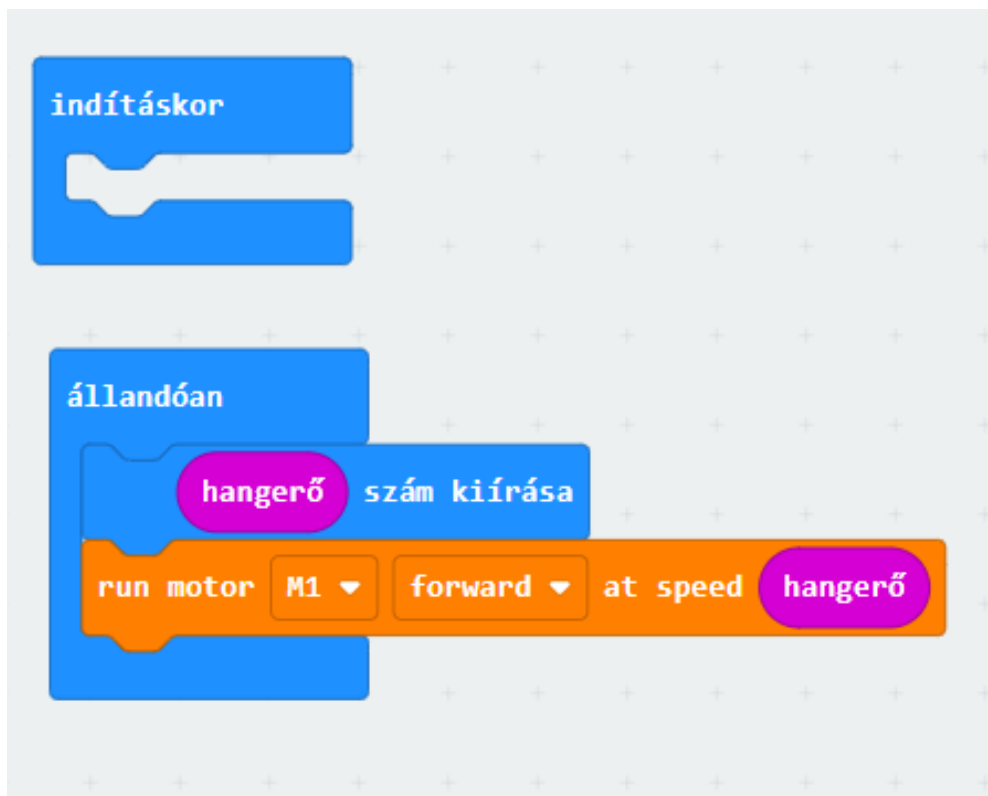


Egy másik példa a sebesség növelésére, ill. csökkentésére:

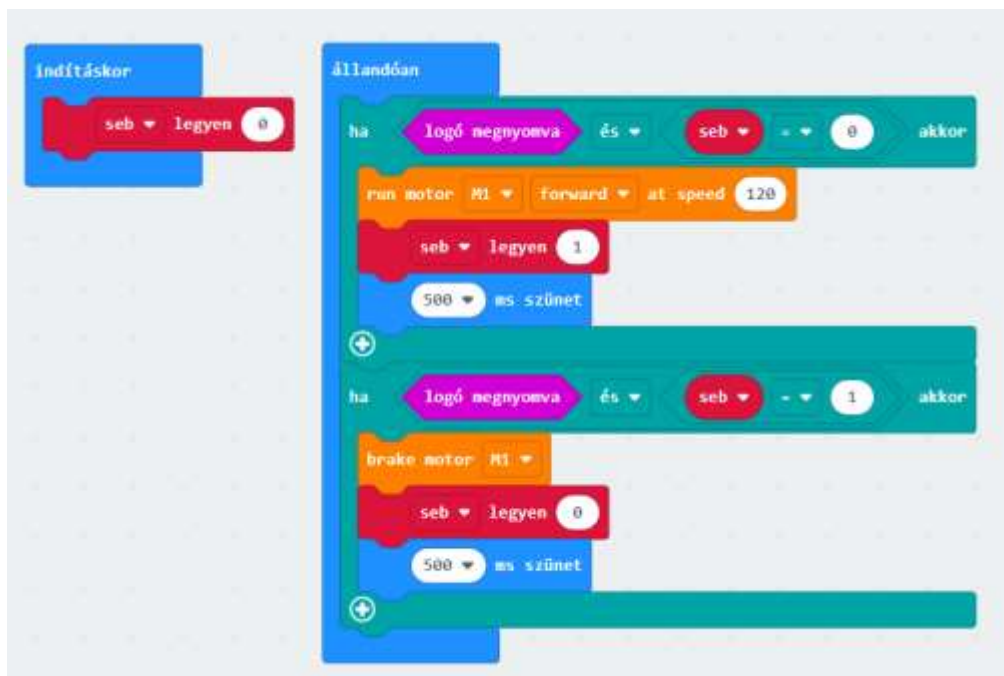
(a seb nevű változót hozzuk létre!)



Ha rendelkezünk micro:bit V2-el, akkor hang vezérelt motor:



Egy kapcsolós vezérlés:



3. A szervó motorok vezérlése.

Az 1960-as években a Yaskawa innovatív egyenáramú szervó motorok sorozatát fejlesztette ki, majd 1983-ban piacra dobták az AC szervomotort, ami új hullámot indított el az autópárházban. Napjainkban, közel 40 évvel később, pedig a gyári automatizálás majdnem 100%-a szervó motort használ.

Mi az a szervó?

Eddig a motorjaink csak pörögtek. Erre is sokszor van szükség, de ettől még gyakrabban van szükség arra, hogy egy motor valamit egy *adott szögben elmozdítson*.

Gondold csak végig, hogy Neked is vannak mozgó részeid, de nincs olyan élőlény az univerzumban, aki bármi részét folyamatosan *forogatni* tudná. Az emberek minden egyes ízülete egy adott kicsi tartományú elfordulásra képes. Amikor ülsz, akkor 90 fokban behajlik a térdizületed, amikor felállsz, akkor pedig kiegyenesedik. Esetleg guggoláskor akár 180 fokban is be tudod ezt az ízületet hajlítani, és ez a 0..180 fokos elfordulás teljesen elegendő mindenre.

Próbáld meg kinyújtani a kezéd, és megvizsgálni, hogy a csuklód mekkorát tud fordulni! Nekem úgy 220 fok jött ki: lehet a tenyerem felül, és a tenyerem alul, és egy kicsit kifacsarva tovább fordítva a tenyerem kifelé néz. Tehát ez az ízület ennyit tud fordulni.

Ha azt kérik tőled, hogy tartsd a kezéd *tenyérrel felfelé*, akkor az agyad a tenyered mostani állapotából kitalálja, hogy merre felé és mennyit is kellene még fordítani a csuklódnak, hogy a tenyered legyen felfelé. Ez remek dolog: hiszen ez teszi lehetővé, hogy dolgokat meg tudjunk fogni, hogy tudjunk sétálni, táncolni!

Az ipari robotokban pontosan ezeket az ízületeket modellezzik egy olyan motorral, ami:

- ✓ lassú de erős elfordulásra képes, ahelyett, hogy pörögne
- ✓ komoly driver van hozzá
- ✓ tudja, hogy most hol van
- ✓ tudja, hogyan jut el leggyorsabban a kívánt elfordulási pozícióba

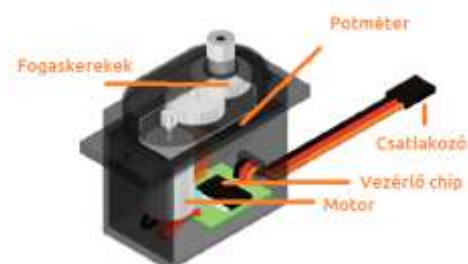
Modell-szervó

Az ipari szervó-hajtások szuper-pontosak, és szuper-drágák. Nekünk sokkal kevesebb elvárásunk van, nem baj, ha a motor nem tud 0.01 fok pontossággal mozdulni, viszont minden más szempont az fontos. Nem akarunk fogaskerekekkel meg driver áramkörökkel vacakolni.

Ezért gondos kezek ezt az egész dolgot összerakták egy hetyke kis dobozkába, ami így néz ki:



Ha belekukkantunk, ott van benne minden:



Ha egy szervót szétszedünk, ilyesmiket találunk benne:



A szervó-hajtás azt jelenti, hogy a mikrokontroller csak parancsot ad, hogy hova kell fordulni, és az elektronika elintézi a motorral, a fogaskerekekkel, és a motor tengelyének a mérésével, hogy a motor a lehető leggyorsabban a kívánt helyre forduljon.

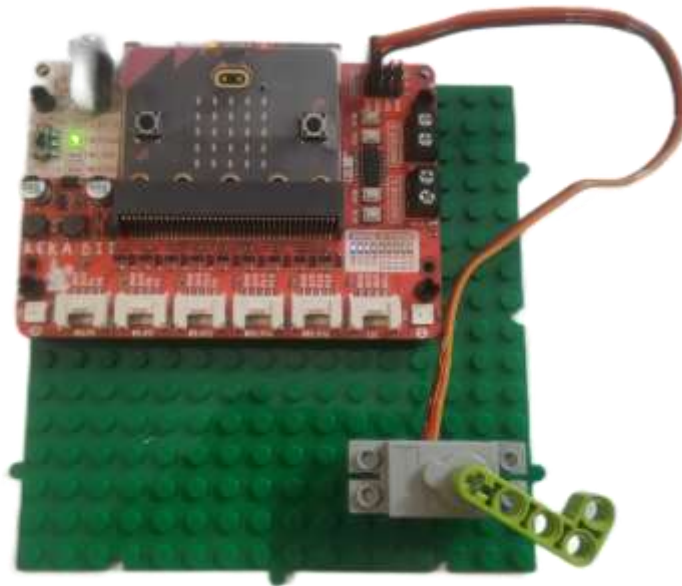
Azt hogy most hol van (és hogy elérte-e a kívánt helyet) egy potméter érzékeli. (A potméter egy forgatható ellenállás: az elfordításával arányos feszültséget fog érzékelni az elektronika)

Többféle servo motort használunk 180^0 - 270^0 - 360^0 utóbbit szokás folyamatos szervónak is nevezni.

A mi készletünkben egy 270^0 -os szervó motort találunk, de a reka:bit kiterjesztése miatt csak 0 - 180^0 tartományban fogunk használni. Viszont akár négy darabot is használhatunk egyszerre, hurrá akár egy robotkar is vezérelhető már ennyi szervóval.

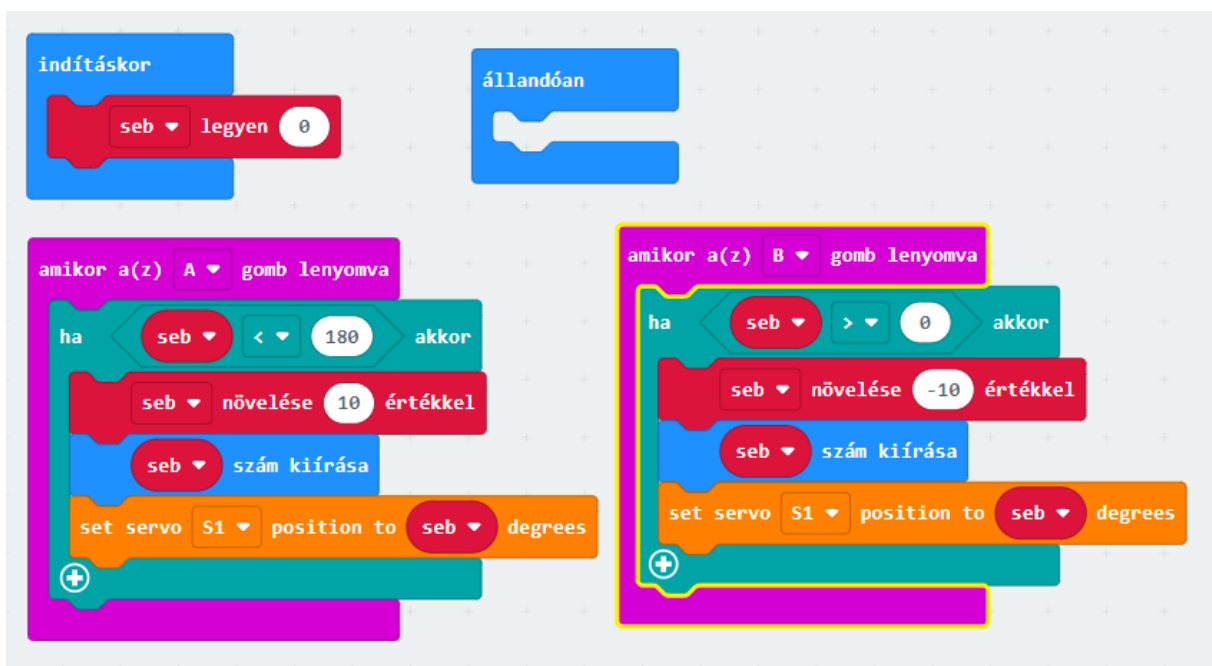


A csatlakoztatásnál különösen ügyeljünk a helyes színsorrendre, fordítva károsíthatjuk a motort.



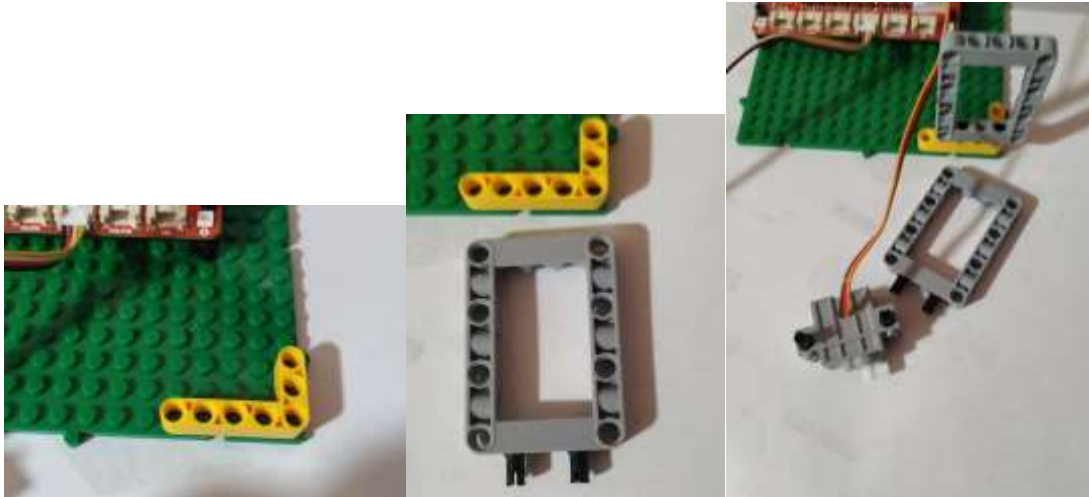
A következő programban A és B gombokkal (kapcsolókkal) mozgatjuk a szervót 10 fokenként. (A mi szervó motorunk 270 fokos, tehát az utasítás hatására többet fog mozdulni, viszont gyönyörűen csatlakozik a lego elemekhez, tehát ha pontosan akarunk dolgozni be kell szerezni egy 180 fokos szervót, vagy át kell számolni.)

Hozzunk létre egy seb változót és próbáljuk ki az értékeket:



4. Sorompó vezérlése.

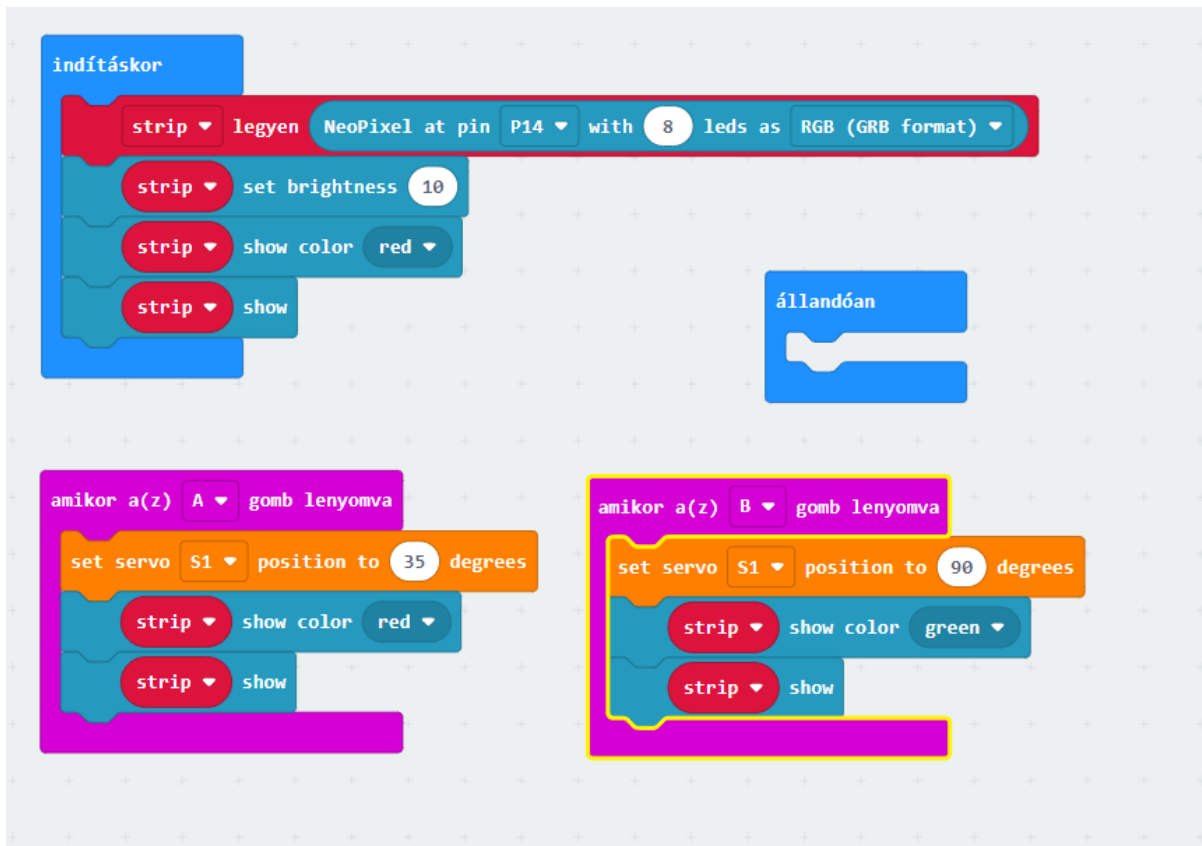
Építsünk egy sorompót, az egyszerűsége törekedtem, ill. annak igazolására, hogy megépíthető a rendelkezésre álló alkatrészekből:



Felhasználjuk a neopixel ledsort amit kétoldalas ragasztóval rögzítettem és a PIN14-re csatlakozik, a servo motor az S1-re, de természetesen ezek más PIN-re is köthetők.

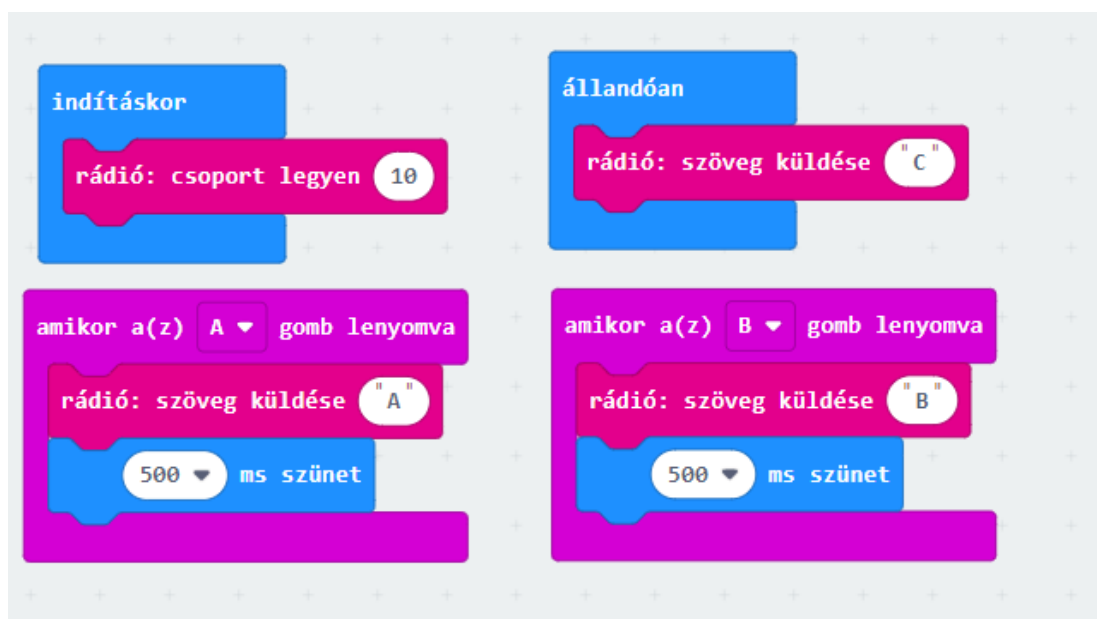
(A sorompó mérete esetleg egy szívószállal kiegészítve egészen komoly méreteket ölthet!)

A reka:bitbe helyezett micro:bittel tudjuk vezérelni az A ill. B gomb megnyomásával amikor is változik a szervó helyzete és a neopixel színe.

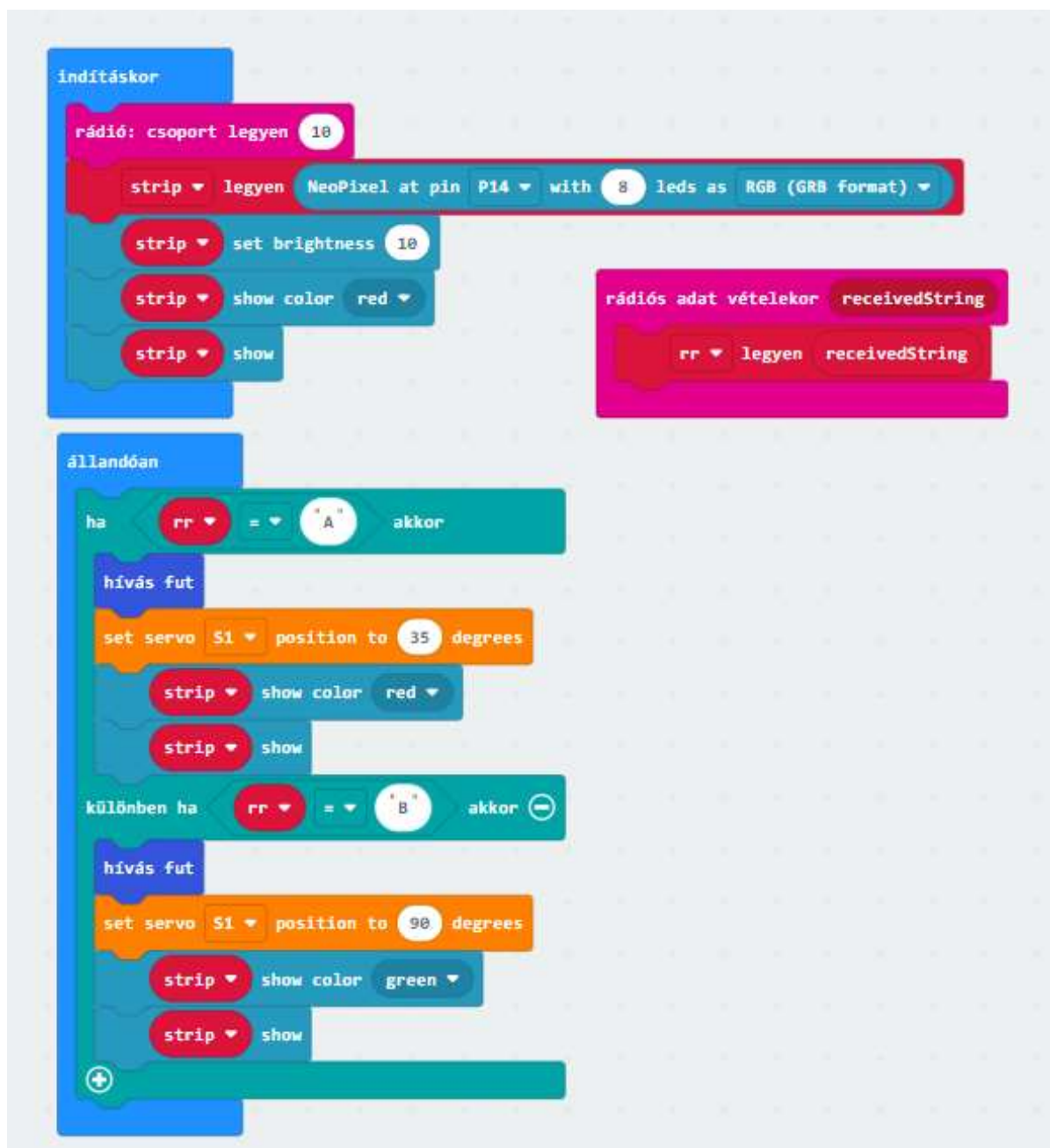


Ez a modell teljesen jól működik, ha van egy portás bácsi aki nyomogatja a gombokat, talán életszerűbb, ha a gépjárműben van egy távirányító és azzal tudom vezérelni a sorompó működését.

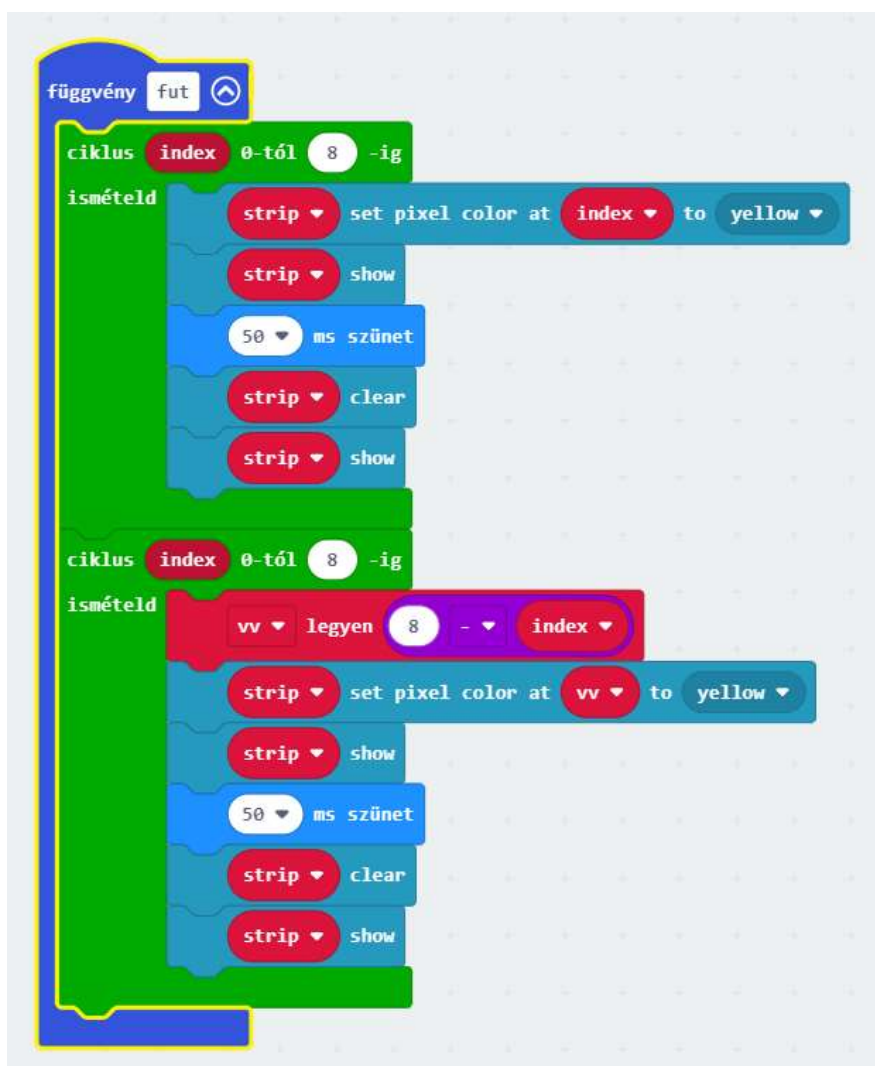
A sorompó távirányítója (rádiós kapcsolat!), ahol küldhettünk volna számot is, de így érdekesebbnek gondoltam:



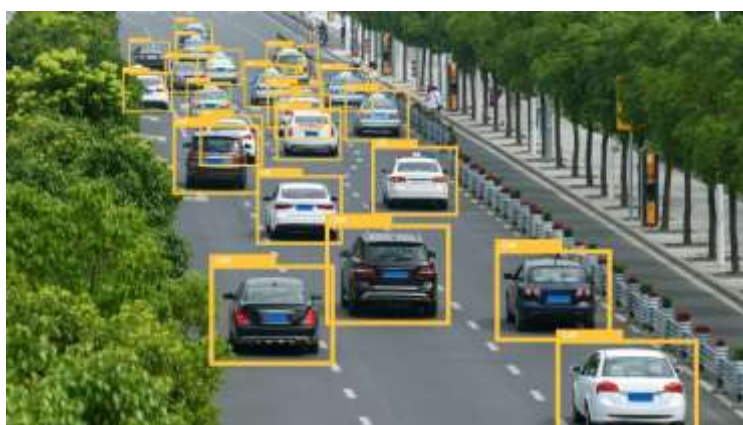
A vevő oldalt is átalakítottam



Érdekesé teszi ez a fut függvény, de nélküle is működik.



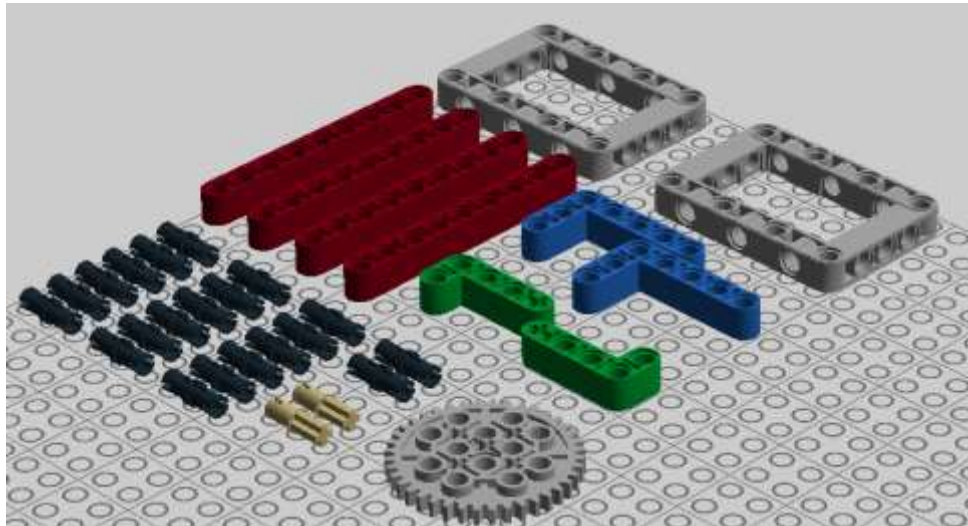
Természetesen lehetne még fokozni az élményeket (infra távirányító, rfid kártya, V2 esetén akár fűtlyre is) láthatjuk mennyire egyszerű csatlakoztatni a grove termékeket, amelyekből több száz rendelkezésre áll. Érdekes lehet automatizálni pl. távolságérzékelővel, infra kapuval vagy a mesterséges intelligenciát (AI) segítségül hívni és akár a rendszám azonosításával működtetni.



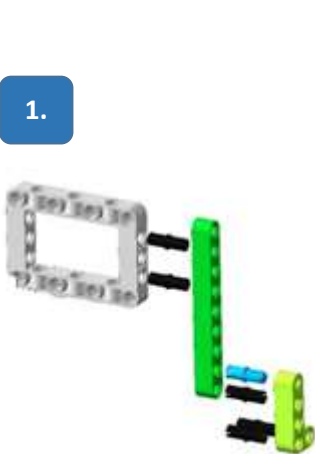
5. Állítható sebességű ventilátor

Már foglalkoztunk a motorok vezérlésével, építsünk egy egyszerű ventilátort, amelynek tudjuk változtatni a forgási sebességét.

Építőelemek összeszerelési lépései



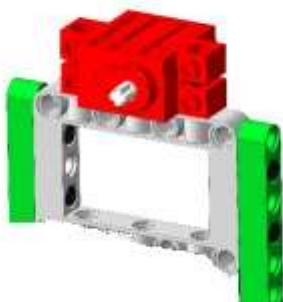
1.



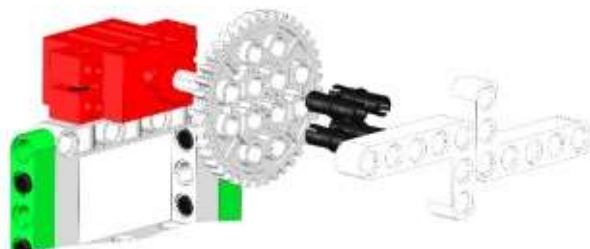
2.

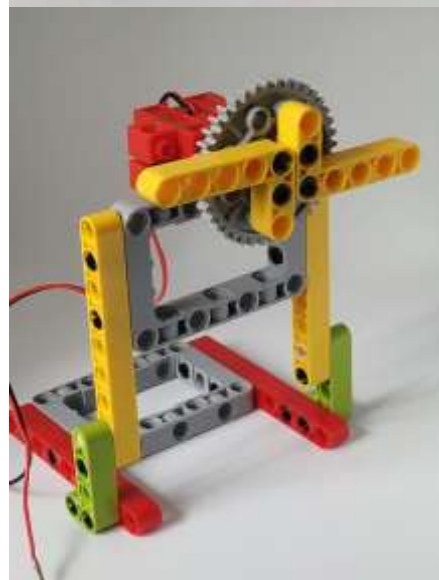
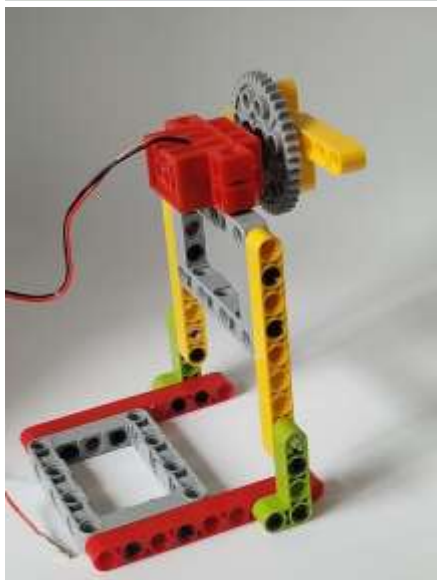
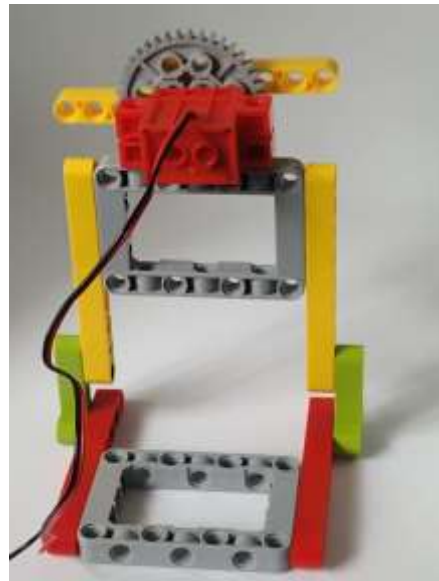
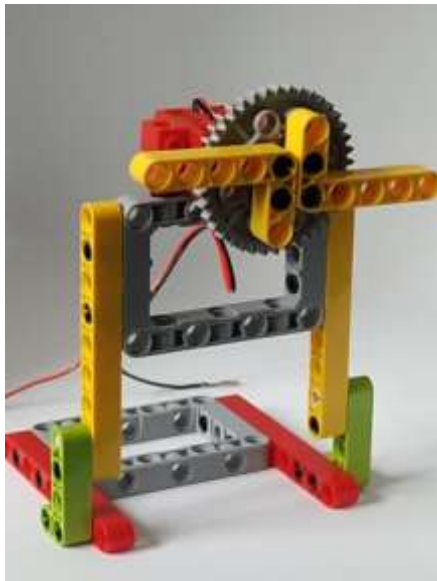


3.



4.



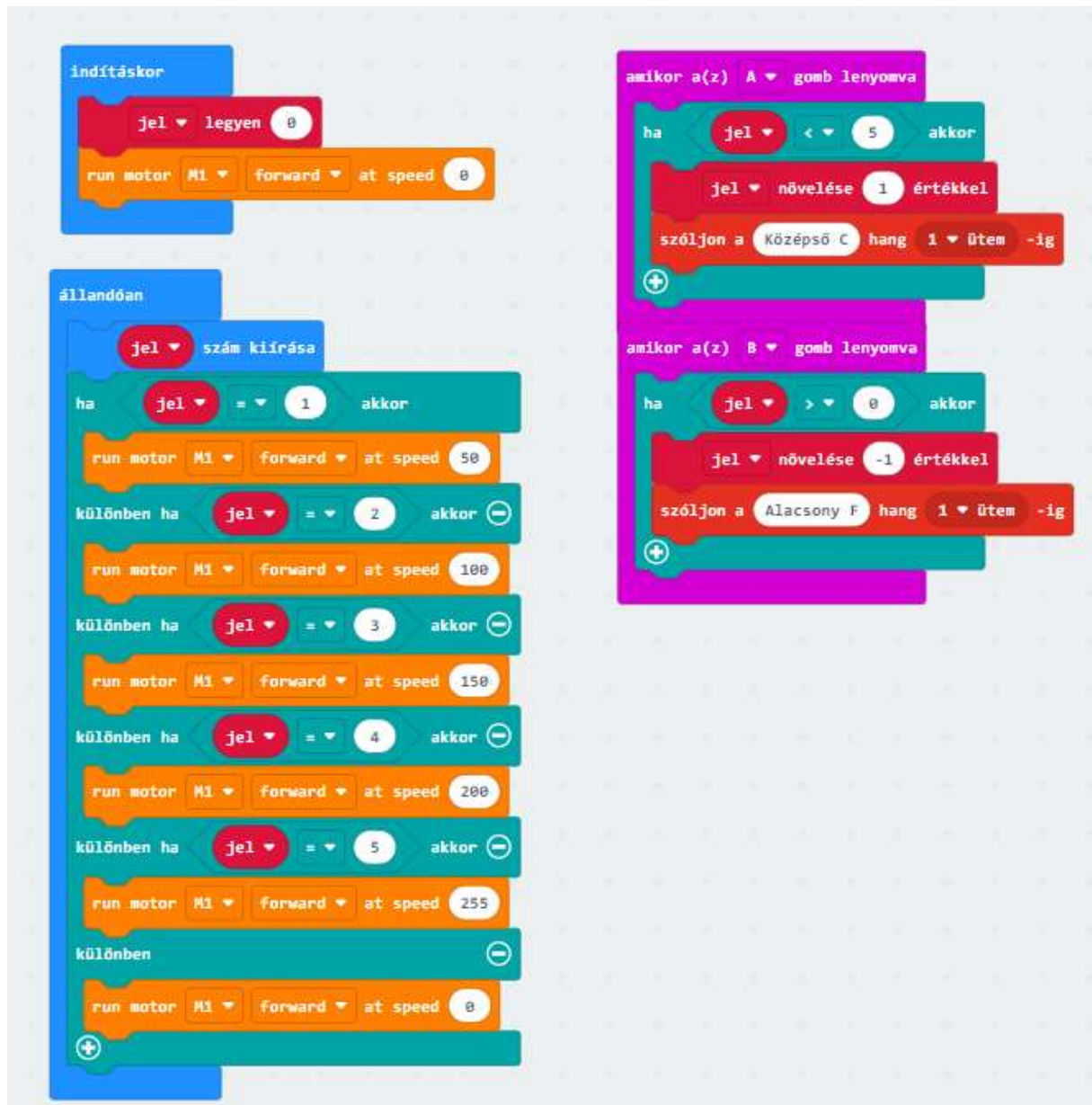


Kísérleti jelenségek:

A program sikeres letöltése után a micro:bit képernyőjén megjeleníti a ventilátor sebességét. Ha megnyomjuk az A gombot, a hangjelző jelez és a ventilátor gyorsítja a sebességét a B gomb megnyomásával az előzőek fordítottja játszódik le.

Létrehoztunk egy **jel** nevű változót, nyilván több megoldása is létezik a feladatnak mi most ezt gondoltuk. Még érdekesebbé tehető a projektet, ha használjuk a neopixel led-eket vagy kijelzővel szenzorokkal egészítjük ki.



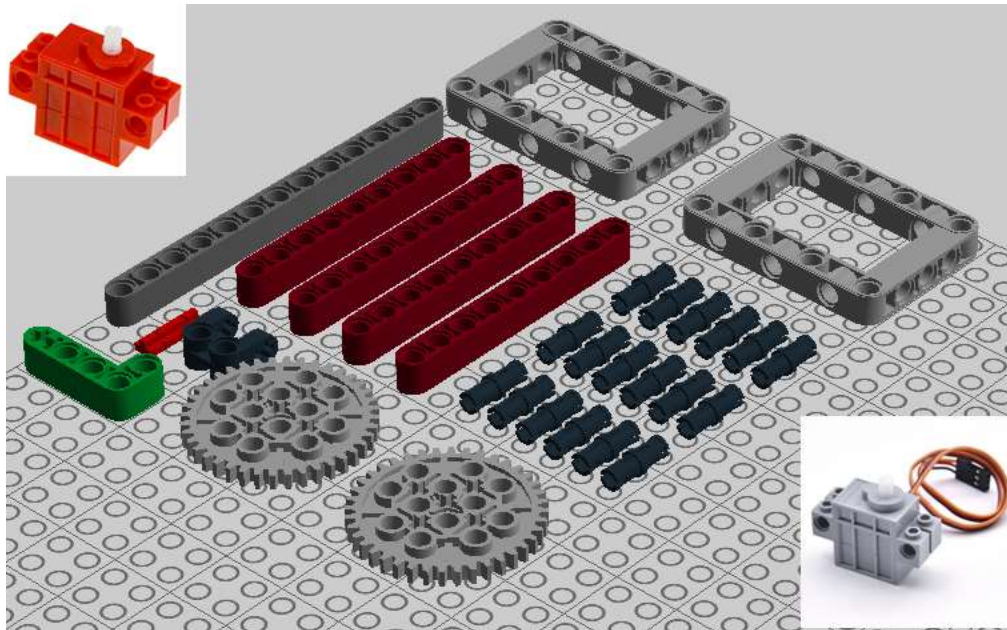


6. Állítható forgásirányú ventilátor .

Tanulási célok:

Ebben a leckében elsősorban azt mutatjuk, hogyan vezérelhető a szervo és egy „sima” DC motor.

A szükséges alkatrészecskék:



Építés:

1.



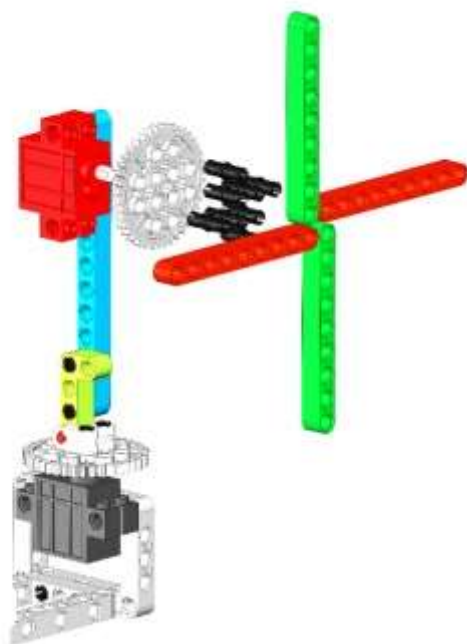
2.



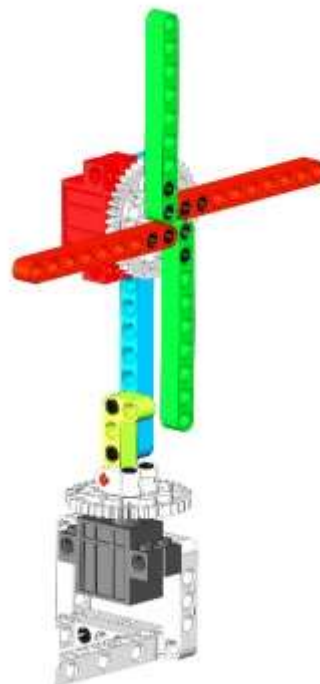
Tekintve, hogy a szervo az alsó részen építhető akár ez a lépés ki is maradhat, tudjuk csatlakoztatni az alaplapra. A színhűséget sajnos nem tudjuk garantálni, teljesen eseti, hogy milyen színben sikerül beszerezni az elemeket.



9.



10.



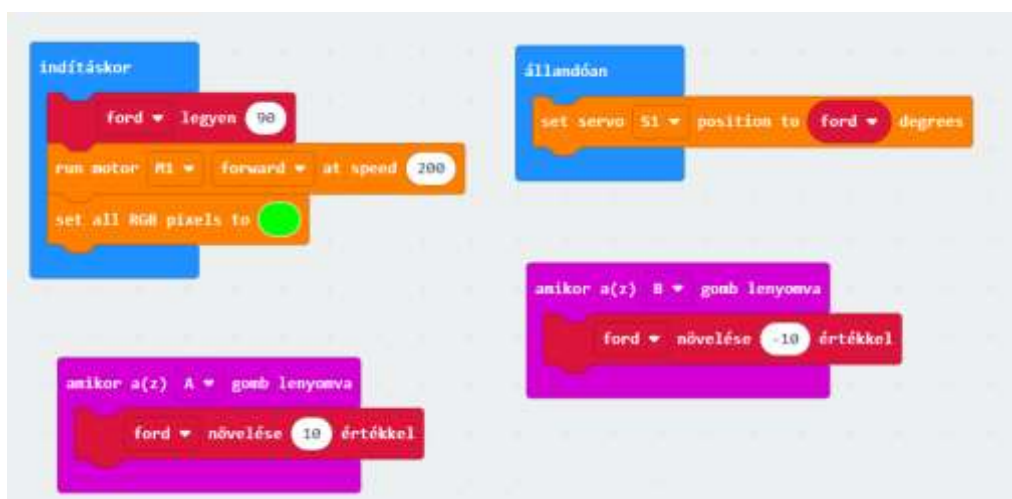
A csatlakoztatás és kódolás:

A DC motor a Motor1, a szervo motor az S1 vezérlőre

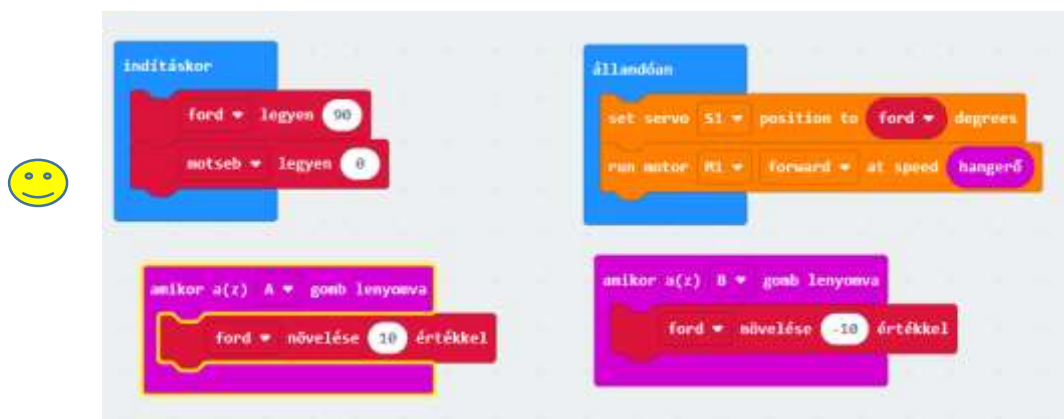
Rendelkezésre áll még a reka:biten 2 db intelligens LED, ill. a micro:bit belső szenzorai, ez már bőséges lehetőséget biztosít, hogy élvezetes és látványos projekteket készítsünk.

Nagyon egyszerű program:

A **ford** - fordulás szöge létrehozunk egy változót, a szervó ezt a pozíciót veszi fel, változtatni az A ill. B gomb lenyomásával lehet (célszerű lesz majd rákérdezni egy max. ill. min. értékre, hogy ne lehessen értelmetlen adat pl. -20, 290), a motor forgási irányára, ill. a LED-ek színét is lehetne változtatni.



A motor forgási sebességének hangvezérlése (V2 esetén!)



A hangvezérlés segítségével (süvítő szél!) akár modellezhetjük a szélérőmű működését is.

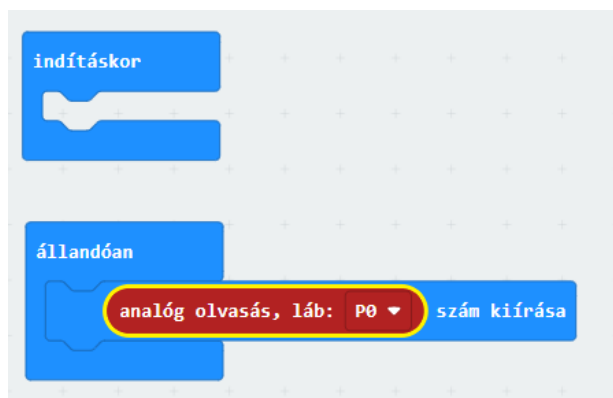
7. Forgásszög mérése

Forgatható potencióméter (ellenállás) nem túl bonyolult elektronikai eszköz, viszont nagyon széles a felhasználási területe



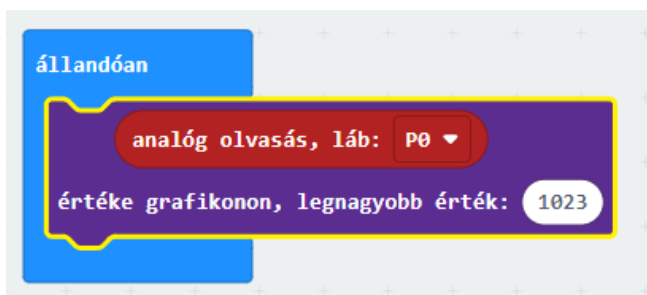
Csatlakoztassuk pl. az első grove csatlakozóba és forgassuk lassan el valamelyik irányba és figyeljük a led-et, látható, hogy változik a fényerőssége.

Változtatható ellenállás, tehát ha elforgatom változik az ellenállás így a feszültség is amit analóg bemeneten tudok mérni. Nézzünk egy példa programot:



Látható, hogy 0 és 1023 között változik az érték. Sajnos az 5*5-ös kijelző nem a legalkalmasabb ennek a megjelenítésére, de azért leolvashatók az értékek.

Talán szemléletesebb és érdekesebb a következő utasítás:

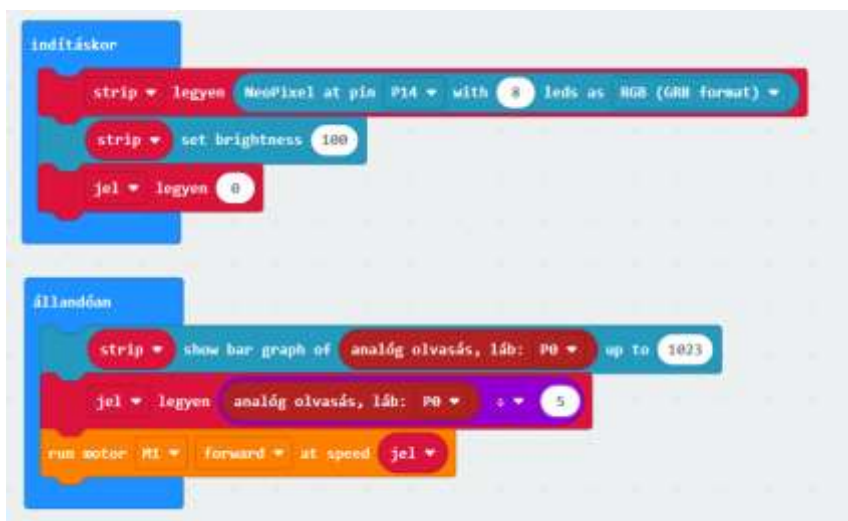


Fokozzuk az élményt, csatlakoztassuk a neopixel oszlopot a P13-14-hez.



A forgási szög függvényében gyulladnak fel a fények.

Kössünk be egy motort az M1-be és látható lesz, hogy szabályozzuk a motor sebességét is.



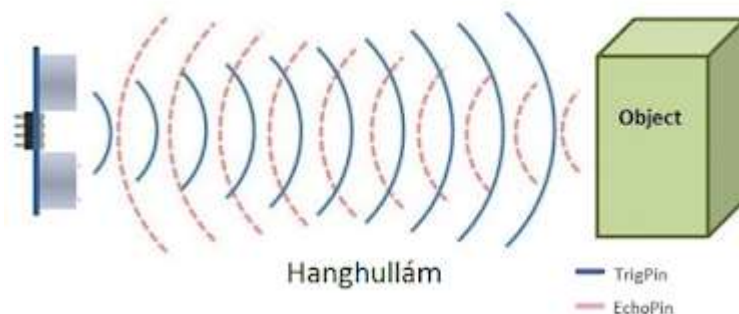
Remélem, hogy nagyon sok projektbe sikerül majd beépíteni.



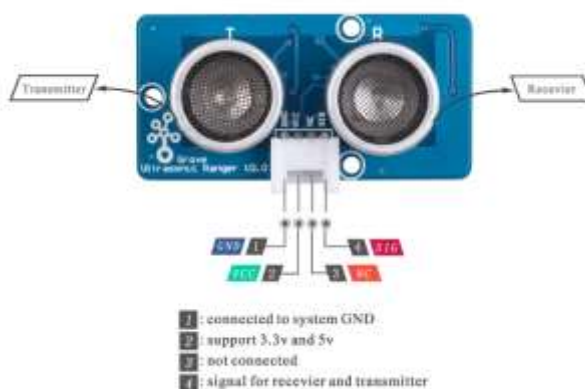
8. Távolságmérés

Ultraszónusos távolságérzékelő (HC-SR04)

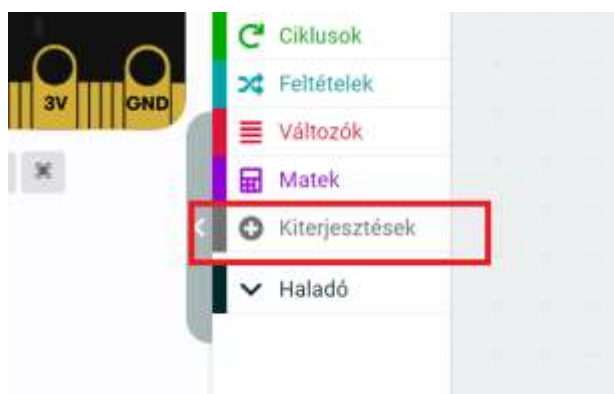
Hanghullámokról



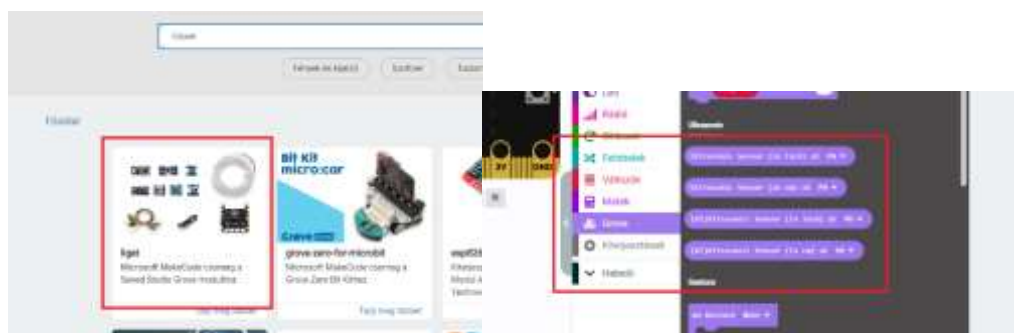
Tételezzük fel, hogy egy objektum 10 cm-re van a szenzortól. A hang sebessége 340m/s (méter/szekundum) vagy 0,034 cm/ μ s (centiméter/mikroszekundum), tehát ekkor a hanghullámnak körülbelül 294 μ s kell haladnia. De az a hullám, amit kapsz majd az Echo pintoól az kétszeres érték mivel a hanghullám elér az objektumhoz majd visszapattanik róla. Tehát annak érdekében, hogy a távolságot cm-ben kapjuk meg, meg kell szoroznunk a visszaérkező útidő értékét (EchoPin) 0.034-gyel (hang sebessége) és osztanunk kell 2-vel.



A hagyományos HC-SR04 ultrahangos modulokhoz képest a Grove - Ultrasonic Distance Sensor egy egychipes mikroszámítógépet integrál, és az adó és a vevő jel egy érintkezőn osztozik időosztásos multiplexeléssel, így csak egy I/O érintkező van elfoglalva. Egy másik különbség az, hogy a HC-SR04 csak 5 V feszültséget támogat, míg a Grove - Ultrasonic Distance Sensor 5 V és 3,3 V feszültséget. Nekünk pontosan erre van szükségünk, hiszen a reka:bit (micro:bit) is 3,3V feszültséget szolgáltat.

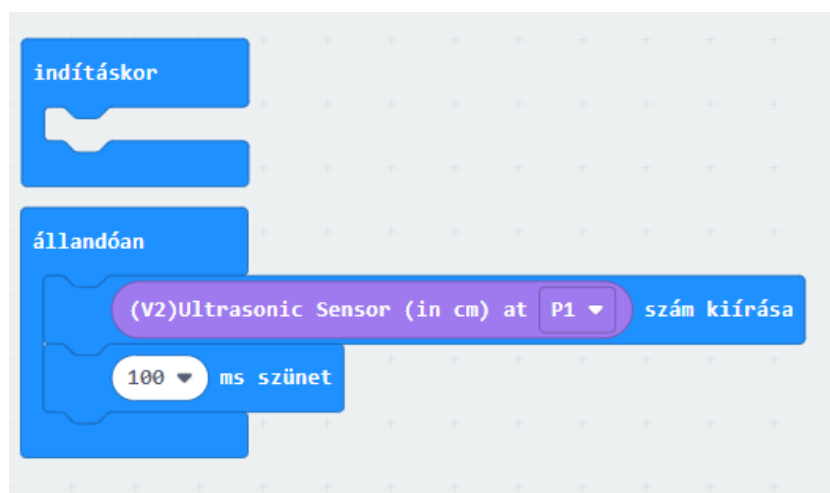


A Grove - Ultrasonic Distance Sensort szeretnénk programozni, az első lépés, hogy megkeressük a megfelelő kiterjesztés.



Itt választhatunk, hogy v1 vagy v2-el akarjuk használni a távolságmérőt. Nincs, túl sok utasítás dönthetünk, hogy cm-ben vagy inch-ben akarjuk az eredményt.

A grove csatlakozón a sárga vezeték csatlakozását kell figyelni, a második aljzatba dugtam tehát a P1-et használom.



1-2 cm-es pontossággal mér, ami nagyobb távolságnál teljesen jó, 10 cm alatt megfelelő.

Érdekes egy vonalzó mentén tapasztalatot szerezni, de figyeljünk, hogy nagyobb kiterjedésű tárgyat használjunk pl. telefon.

Ismét egy szuper eszköz (szenzor) amivel a projektjeinket feldobhatjuk pl. sorompó!

9. Hangszer készítése

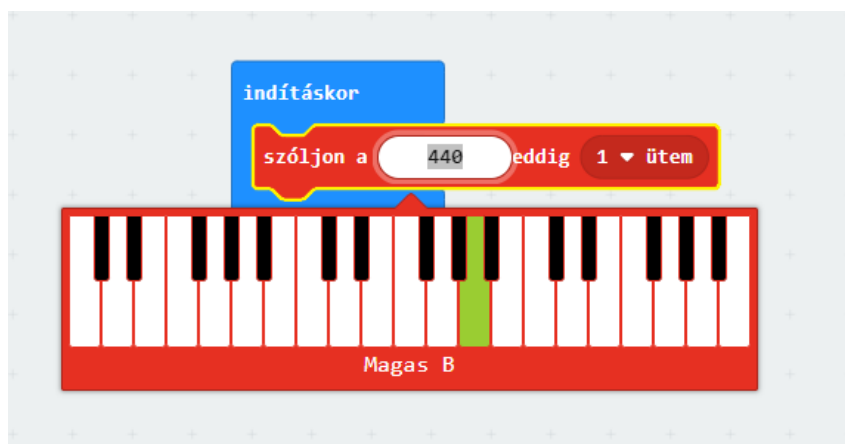
Dalok átírása kottából

Ha újra szeretnénk létrehozni kedvenc dalainkat a micro: bit-en, először a kották alapvető ismeretére van szükségünk.






Mielőtt belevágnánk érdemes néhány dolgot tisztázni:



Ahol középső C = hang és 1 ütem = időtartam.



Minden az időzítésről szól.

Notes	Name	Value	Code
	Semibreve Whole note	4 beat	4 ▾ beat
	Minim Half note	2 beat	2 ▾ beat
	Crotchet Quarter note	1 beat	1 ▾ beat
	Quaver Eighth note	1/2 beat	1/2 ▾ beat
	Semiquaver Sixteenth note	1/4 beat	1/4 ▾ beat

Notes – teachwithict.com

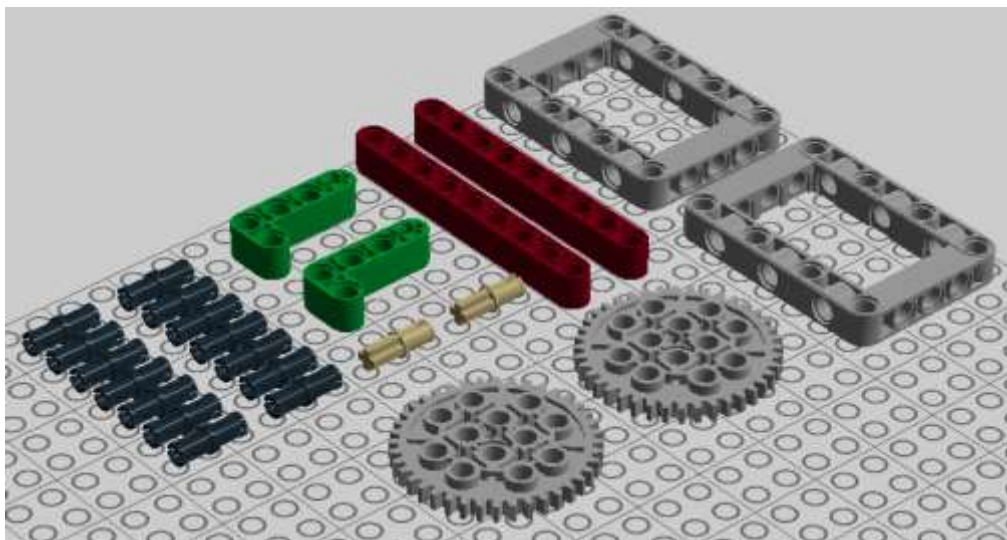
Notes	Name	Value	Code
	Dotted Semibreve Whole note	6 beat	4 ▾ beat ▾ ▾ 1.5
	Minim Half note	3 beat	2 ▾ beat ▾ 1.5
	Crotchet Quarter note	1 1/2 beat	1 ▾ beat ▾ 1.5
	Quaver Eighth note	3/4 beat	1/2 ▾ beat ▾ 1.5
	Semiquaver Sixteenth note	3/8 beat	1/4 ▾ beat ▾ 1.5

Dotted notes – teachwithict.com

10. Zene_Metronóm

Programozunk zenét és szolgáltatassuk a ritmust.

Építőelemek összeszerelési lépései:



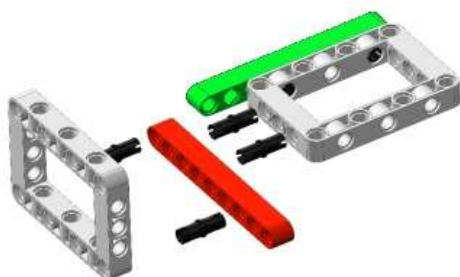
1.



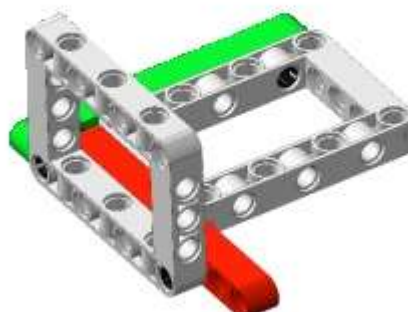
2.



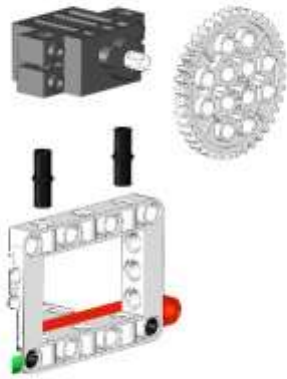
3.



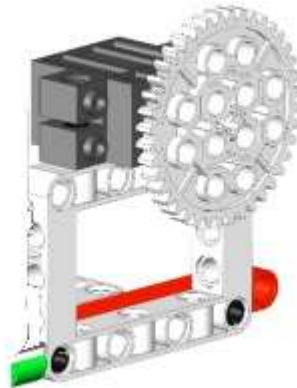
4.



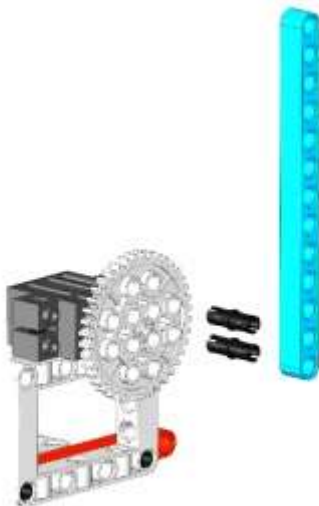
5.



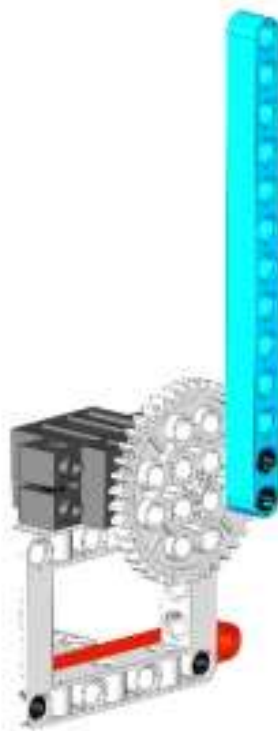
6.



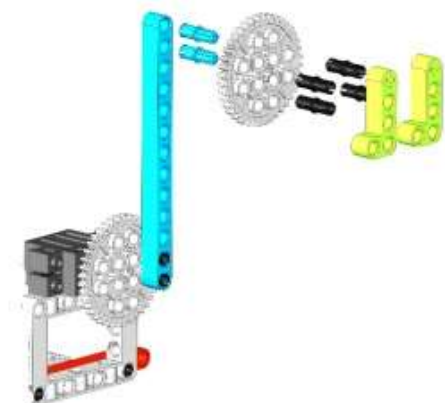
7.



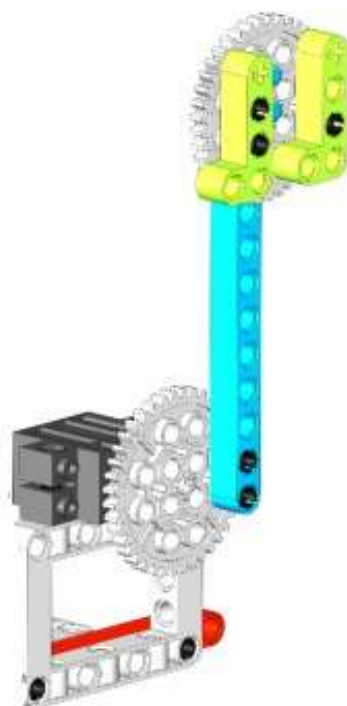
8.



9.



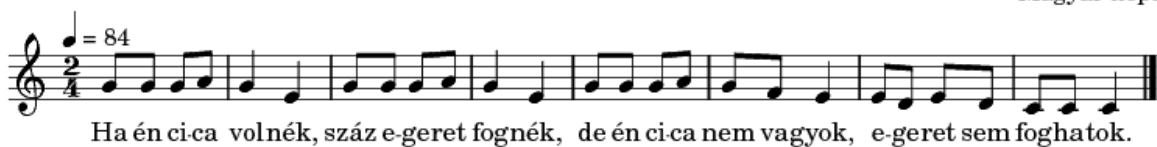
10.



A lényeg megépíthető, de javaslom rögzíteni, mert a lendület nagy erővel mozgatja.
Érdekes lehet kiegészíteni fényjátékkal akár egy LED szalagon.

Ha én cica volnék

Magyar népdal


<http://dalok.theisz.hu/?page=song&id=HaEnCicaVolnek>

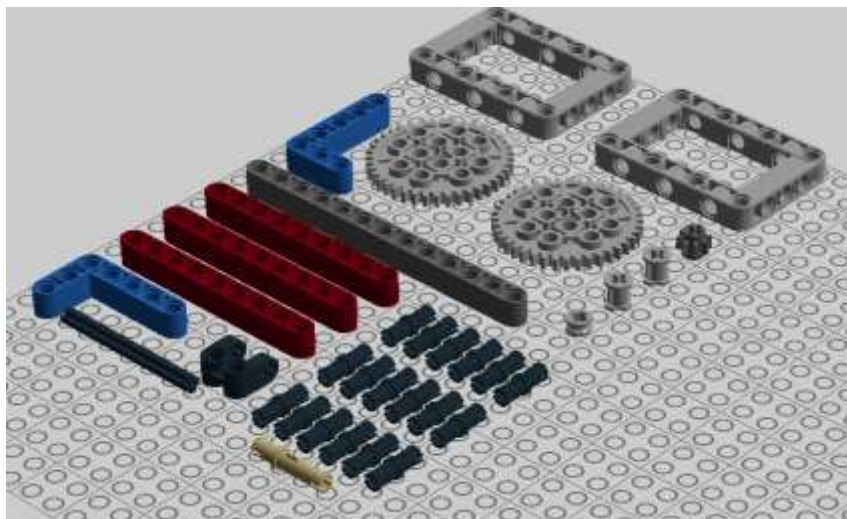

Próbáltam a ciklusokkal rövidebbre írni, hát ennyire sikerült.



11. Hajítógép

A hajítógép avagy katapult mechanikus szerkezet, állványra helyezett hideglőfegyverfajta. A lőpor feltalálása előtt az ókor és a középkor meghatározó ostromgépe illetve tábori tüzérsége volt.

Építőelemek összeszerelési lépései:



1.



2.



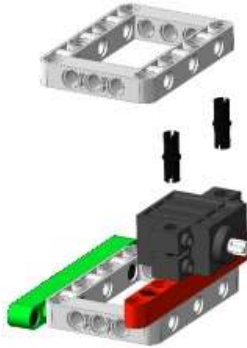
3.



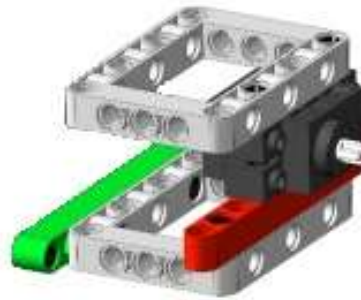
4.



5.



6.



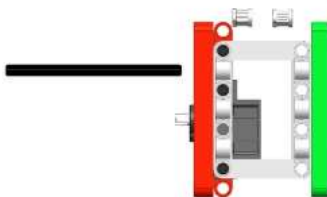
7.



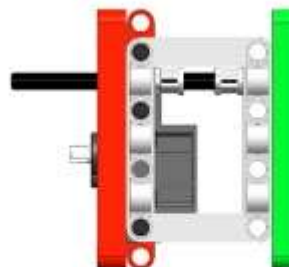
8.



9.



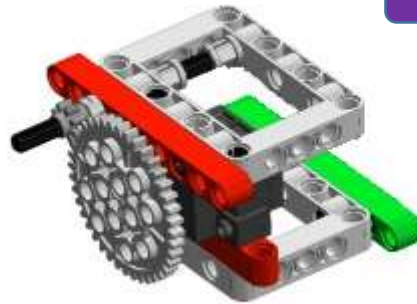
10.



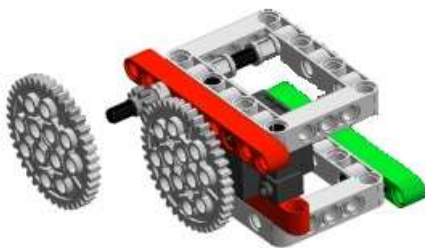
11.



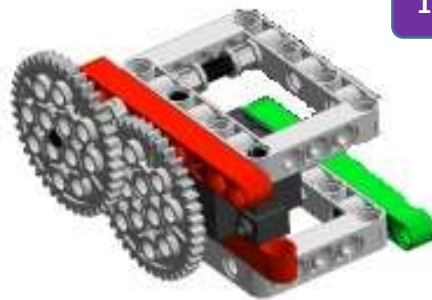
12.



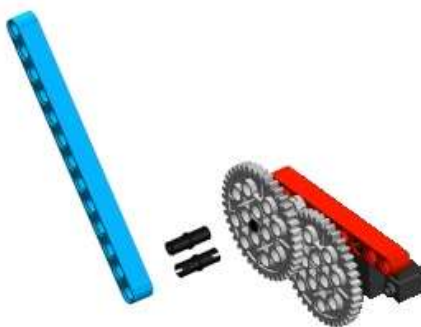
13.



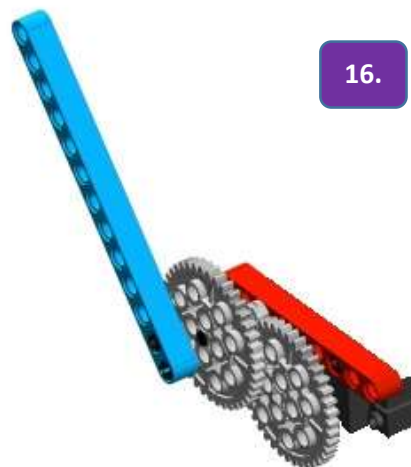
14.



15.



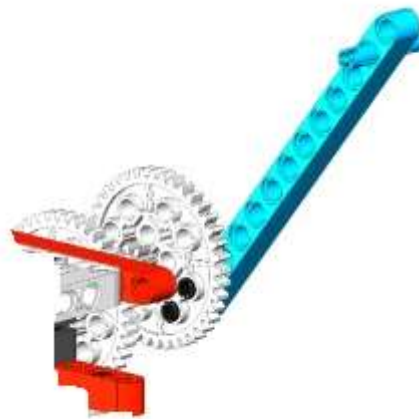
16.



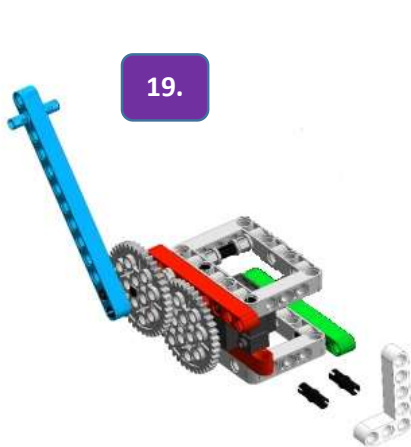
17.



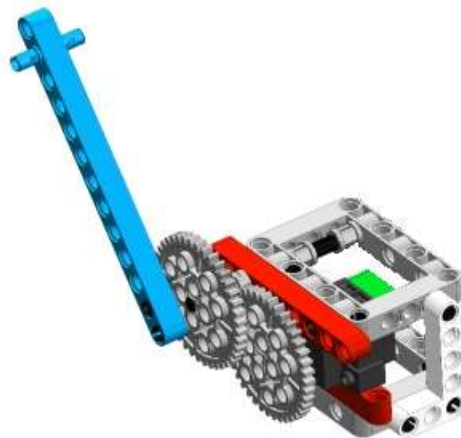
18.



19.



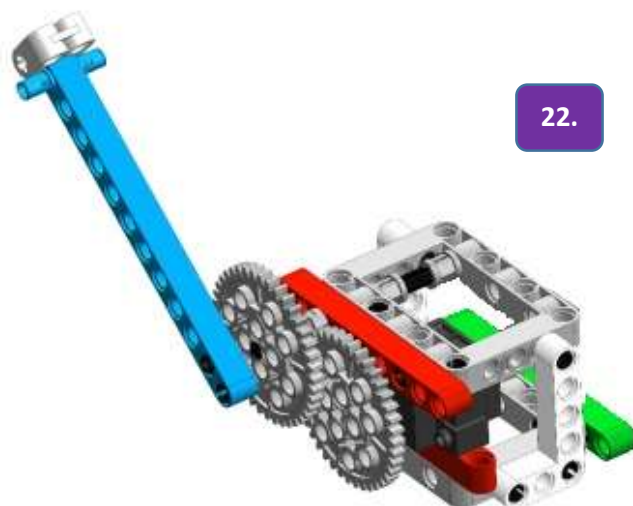
20.



21.

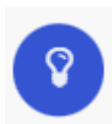
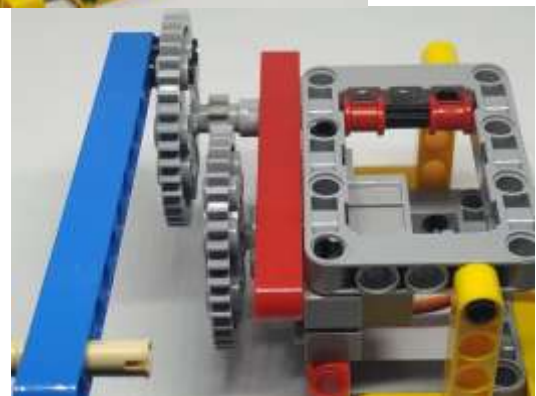
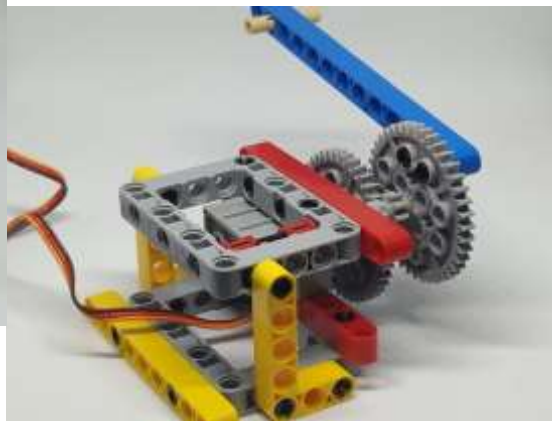
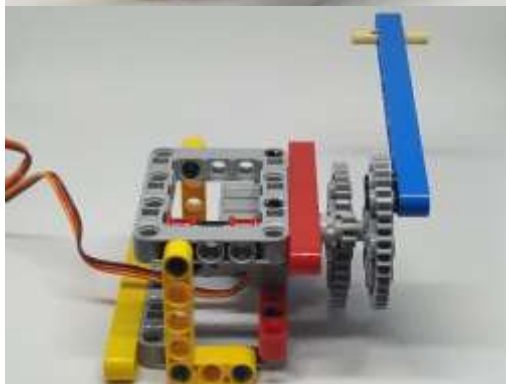
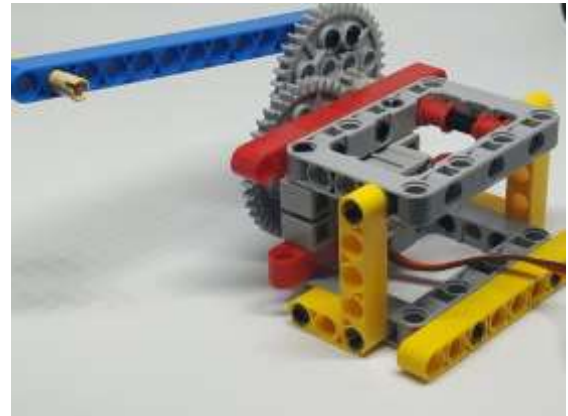
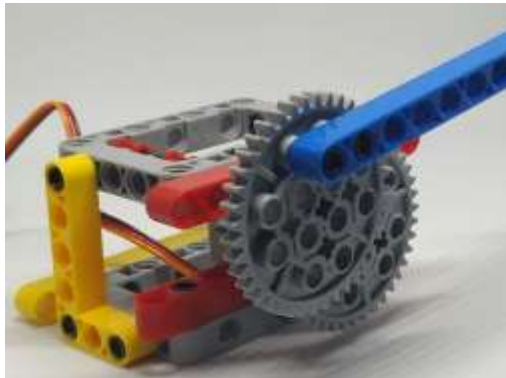


22.



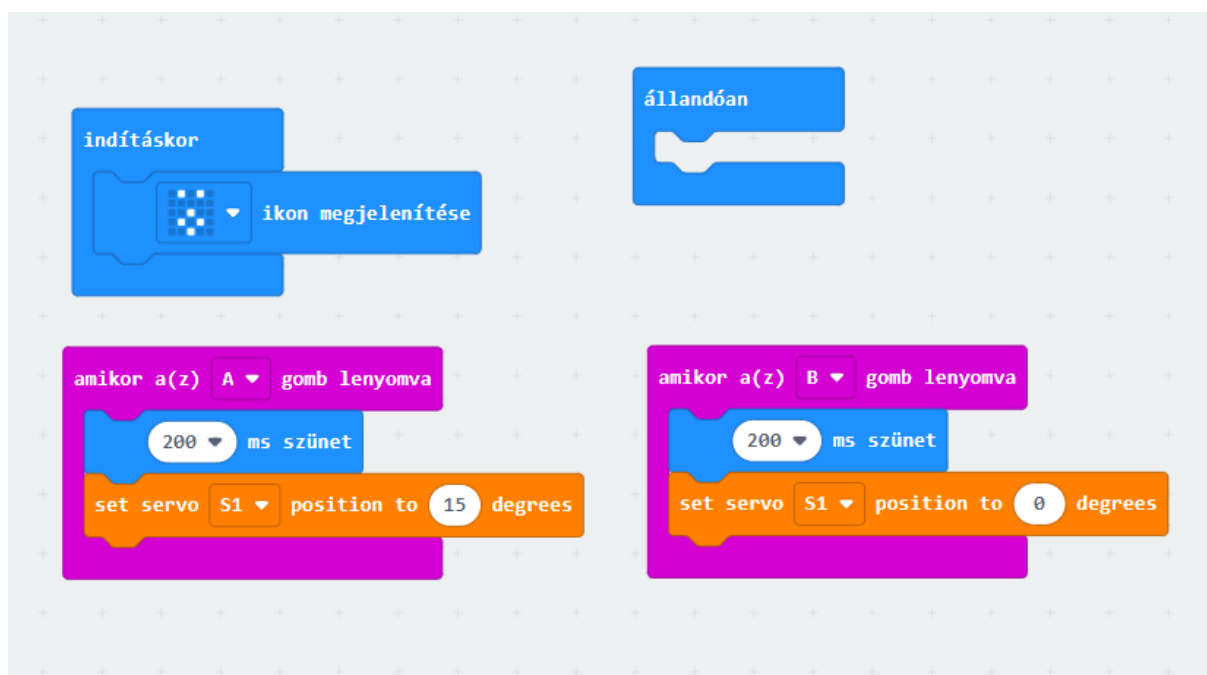
Kísérleti jelenségek

Az A gombot megnyomva dobhatjuk ki az objektumot, a B gombbal visszatérünk.



Az összeszerelés lényegesen nehezebb, mint a szerkezet programozása.

Lehetne folytatni a projekteket, de ez csak a figyelem felkeltése, néhány lehetőség reményeim szerint a felhasználók sokkal jobbakat találnak majd ki.



Nagyon remélem, hogy sikerült felkeltenem az érdeklődést a készlet iránt és sok hasznos percet töltenek majd vele. Ha tudok segíteni, nagyon szívesen teszem:



Vígvári György

vigvari.gyorgy@vigvari.hu

www.vigvari.hu

