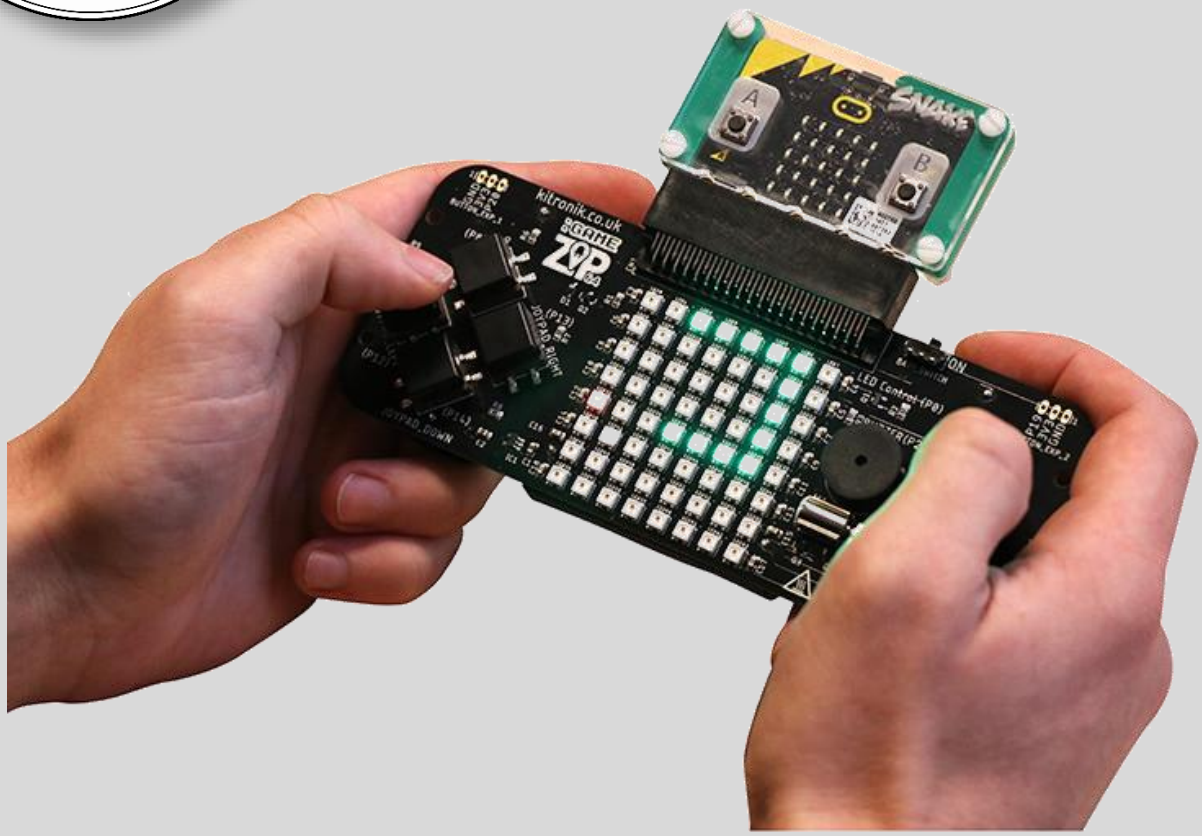
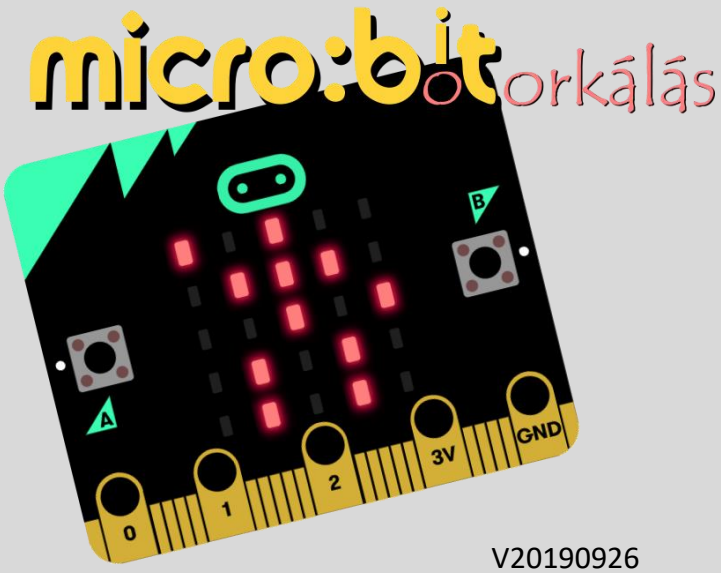
Micro:bit (:GAME ZIP 64) botorkálás

Segédanyag tanárok számára



A Kitronik GAME ZIP 64 használata

Dr. Abonyi-Tóth Andor



Eötvös Loránd Tudományegyetem
Informatikai Kar

V20190926

Tartalomjegyzék:

A készletről 2

Tudnivalók, óvintézkedések 2

Az eszköz felépítése 3

Programozás a makecode felületen 3

Szükséges blokkok hozzáadása 3

A :GAME ZIP64 kategóriában elérhető blokkok 4

Egyszerű demó projektek a korábban ismertetett blokkok megismeréséhez 6

1. Színek váltakozása 6

2. Szivárvány színeinek megjelenítése 6

3. LED-ek színeinek beállítása koordinátánként 6

4. Véletlenszerűen meghatározott LED-el, véletlenszerű színnel 7

5. Színek rotálása 7

6. Rezgőmotor és zümmer használata 7

Komplex projekt elkészítése 9

Űrhajós játék elkészítése a GAME ZIP 64-re. 9

Végszó 11

A készletről

A GAME ZIP 64 készletet a Hi-Tech alapítvány, a micro:bit lapkákat és kiegészítőket a Kitronik magyarországi képviselete a PI-SHOP (https://www.pi-shop.hu/) bocsátotta a micro:bit botorkálás program rendelkezésére.

Ez a speciális készlet a következő eszközöket tartalmazza:

- 10 db :GAME ZIP 64 kontroller
- 10 db micro:bit védőtokban, elemtartóval
- 32 db tölthető AA elem a GAME ZIP kontrollerekhez
- 3 db elemtöltő (12 db elem szimultán töltésére)
- 10 db USB kábel

A kontroller segítségével még látványosabb, élvezetesebb játékprogramokat készíthetnek a diákok. A kontrolleren többek között 8×8-as RGB LED mátrix, 4 gomb az irányításhoz (fel, le, jobbra, balra), 2 tűz (fire) gomb, 1 db rezgőmotor és a hangok megszólaltatásához egy buzzer (zümmer) található.

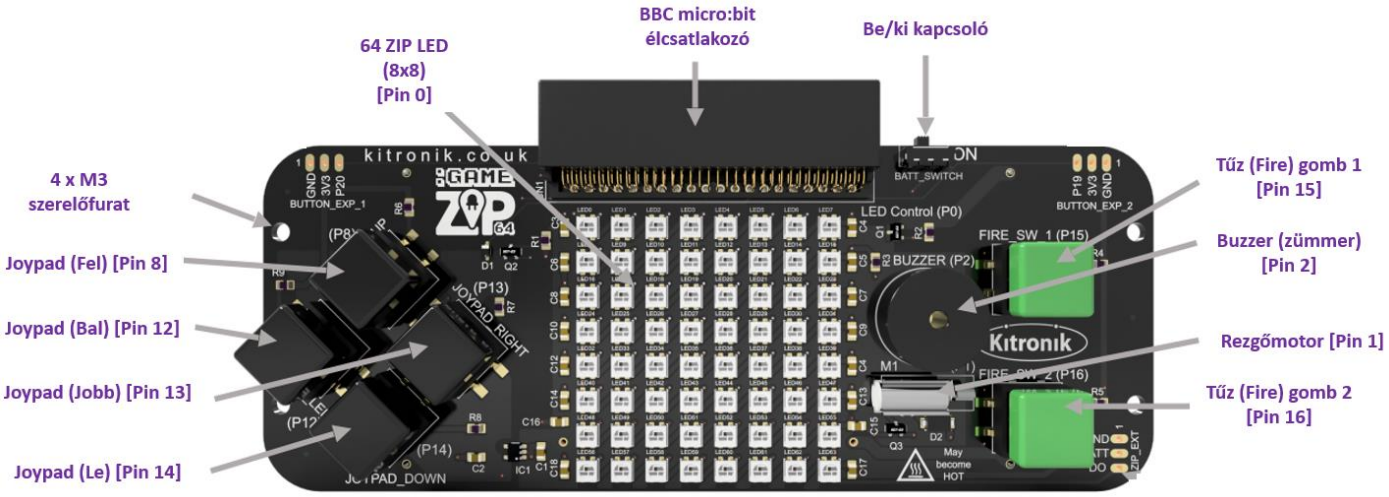
Tudnivalók, óvintézkedések

- A kontroller működéséhez az AA elemeket el kell helyezni az elemtartóban a polaritásuknak megfelelően. **USB kábellel nem működik** az eszköz!
- A kontrollerek **épségére** vigyázzunk! A habszivacs tartóból óvatosan vegyük ki, mivel a **kiálló alkatrészek** (különösen az elemtartók) sérülékenyek!
- Az **elemek cseréjénél** szintén kellő **gondossággal** járjunk el! Vigyázzunk, hogy a csere során ne sérüljenek meg az elemtartók és az alkatrészek!
- A csomag **postázása** előtt a dobozt **béleljük ki** habszivaccsal, papírral, hogy ne rázkódjanak az alkatrészek szállítás közben!
- Fontos! A Neopixel LED mátrix eléggé **vakítóan tud világítani.** A szemek és az elemek kímélése érdekében mindig alacsony világosságértéket (brightness) használjunk! (pl. 5, a maximum 255 helyett) a programozás során.
- A LED-ek huzamosabb használat és nagy fényerő mellett **felforrósodhatnak**, így azokat lehetőleg ne érintsük meg!
- Amikor nem használjuk az eszközt, az **On/Off kapcsolóval kapcsoljuk ki**, különben az elemek gyorsan lemerülnek.

Készítette: Dr. Abonyi-Tóth Andor, ELTE Informatikai Kar

Az eszköz felépítése

Az alábbi ábrán láthatjuk az eszköz főbb részeit. A micro:bit eszközt a felül található élcsatlakozóba kell bedugnunk.

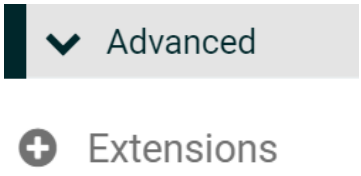


Az elemeket a hátoldalon található elemtartókban kell elhelyeznünk polaritásnak megfelelően.

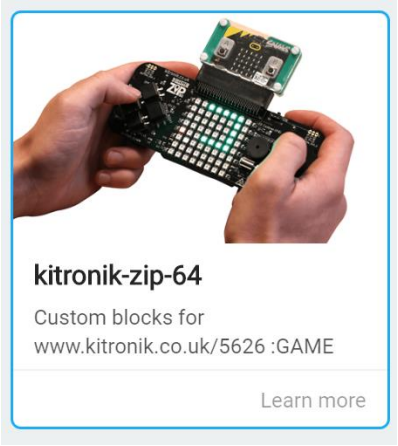
Programozás a makecode felületen

A Makecode felületet a https://makecode.microbit.org/ webcímen találjuk.

Szükséges blokkok hozzáadása



A Makecode felülethez adjuk hozzá a Kitronik ZIP 64 nevű speciális blokkot. Ehhez kattintsunk az **Advanced** kategóriára, majd a **+Extensions** linkre.

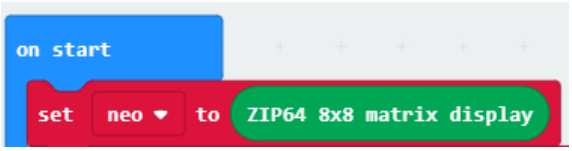
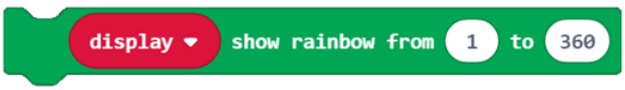
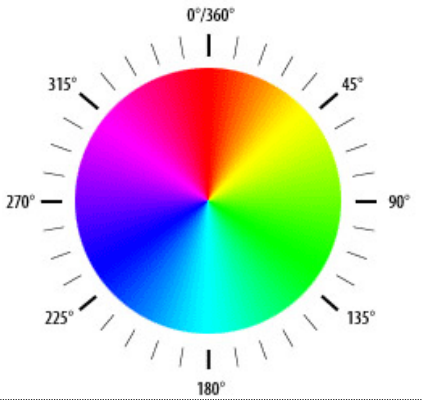
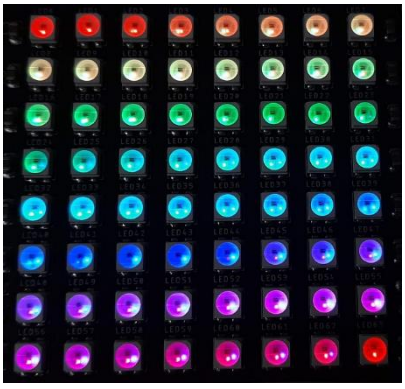


Keressünk rá a „zip 64” kifejezésre. Ekkor megjelenik a **kitronik-zip-64** modul, amelyre kattintva telepíthetjük azt. Ha a keresés nem adna találatot a kitronik ZIP 64-re, akkor a https://github.com/KitronikLtd/pxt-kitronik-zip-64.git webcímet be kell írni a mezőbe.

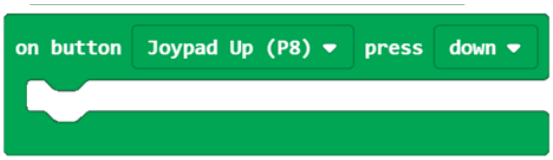
Ha mindent jól csináltunk, megjelenik a blokkok között a következő blokk:

Készítette: Dr. Abonyi-Tóth Andor, ELTE Informatikai Kar

A :GAME ZIP64 kategóriában elérhető blokkok

Display (Megjelenítő) szakasz	
	A LED mátrix inicializálásához szükséges. A blokot egy változó értékének kell adnunk. A kijelző későbbi beállításainál majd erre a változóra hivatkozunk. pl. 
	Az adott koordinátájú LED színének megváltoztatására szolgál. Az oszlopok és sorok nullától indexelődnek. A bal felső LED koordinátája: (0;0). A jobb alsóé: (7;7).
	A teljes LED mátrix színének beállítására szolgál. A választható színek: red, orange, yellow, green, blue, indigo, violet, purple, white, black.
	A kijelző tartalmának megjelenítésére szolgál. Amennyiben pl. a LED mátrix elemeit koordinátánként színeztük át, akkor szükség van a módosítások érvényre juttatására, amit ezzel a blokkal tehetünk meg.
	A kijelzőt letörli, alaphelyzetbe állítja.
	A kijelző fényerősségének beállítására szolgál (0-255). Vigyázzunk, mert a magas értékeknél nagyon vakító lehet a kijelző!
	A blokk segítségével a kijelző LED-jein a szivárvány színei jelennek meg. A megadott értékekkel azt tudjuk befolyásolni, hogy mi legyen a kezdő szín, illetve a végső szín.
	A bal oldali színkörön láthatjuk, hogy melyik szögértéknek melyik szín felel meg. Ha a szivárványt az alapbeállítás szerint rajzoljuk (1 és 360 között), akkor a jobb oldalon látható eredményt kapjuk. 

Készítette: Dr. Abonyi-Tóth Andor, ELTE Informatikai Kar

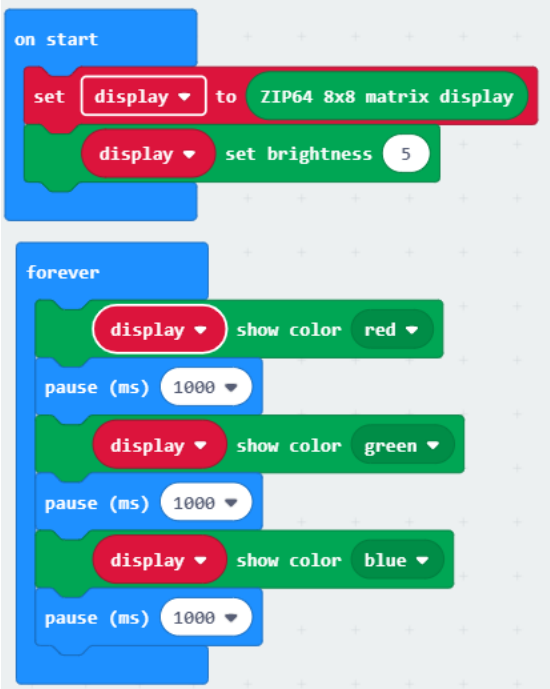
	A kijelző színeinek megváltoztatására, rotálására szolgál.
Ahhoz, hogy megértsük mi fog történni, ismét a színkört kell segítségül hívnunk. A színkörben a 0 fokhoz a piros szín tartozik. Ha 90 fokot fordulunk az óramutató járásával megegyező irányába, akkor viszont már a zöld színt kapjuk, ha újabb 90 fokot, akkor a türkízt és így tovább. Vagyis a rotálás során a megadott fokkal fognak a színek eltolódni.	
	Színkonstans, amely a színek beállítására szolgál. A választható színek: red, orange, yellow, green, blue, indigo, violet, purple, white, black.
	A blokk segítségével tetszőleges színt beállíthatunk, az RGB koordináta-rendszer szerint.
Input (Bemenet) szakasz	
	Logikai feltételeknél használhatjuk. A kifejezés igaz értéket ad vissza, ha a Joypad vagy Fire (tűz) gombok le lettek-e nyomva.
	Ezzel adhatjuk meg, hogy mi történjen (milyen blokkok hajtódjanak végre) a Joypad és Fire (tűz) gombok megnyomásakor.
Feedback (visszajelzés) szakasz	
	Ezzel a blokkal a rezgőmotort kapcsolhatjuk be a megadott ideig. Ilyenkor a kontroller el kezd rezegni.
	Amennyiben a hangokat a kontroller zümmerjén akarjuk megszólaltatni, ezt a blokkot kell használnunk a projektünk elején.

Egyszerű demó projektek a korábban ismertetett blokkok megismeréséhez

1. Színek váltakozása

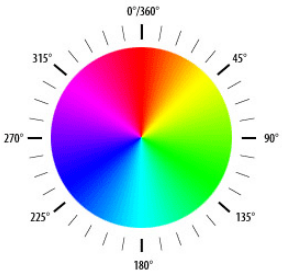
Az alábbi projektben beállítjuk a fényerőt 5-ös értékre és a piros, kék, zöld színeket váltogatjuk a teljes kijelzőn 1 másodperces késleltetéssel.

A projekt elérhetősége:
https://makecode.microbit.org/_C1aid76JWipy

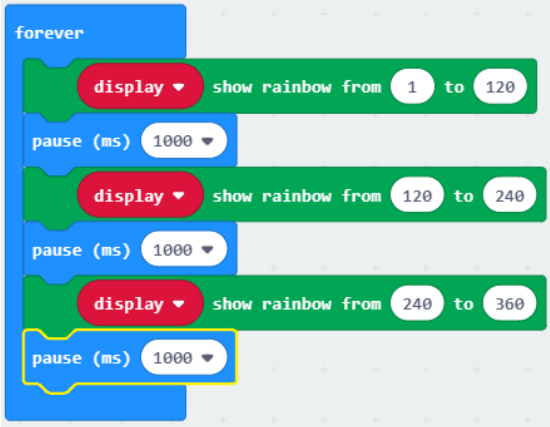


2. Szivárvány színeinek megjelenítése

Ebben a projektben a szivárvány színeit jelenítjük meg, mégpedig három fázisban. Először 1 és 120 között (piros-zöld), aztán 120 és 240 (zöld-kék) között, majd 240 és 360 (kék-piros) között.



A projekt elérhetősége:
https://makecode.microbit.org/_g0VTY7YdTK3D



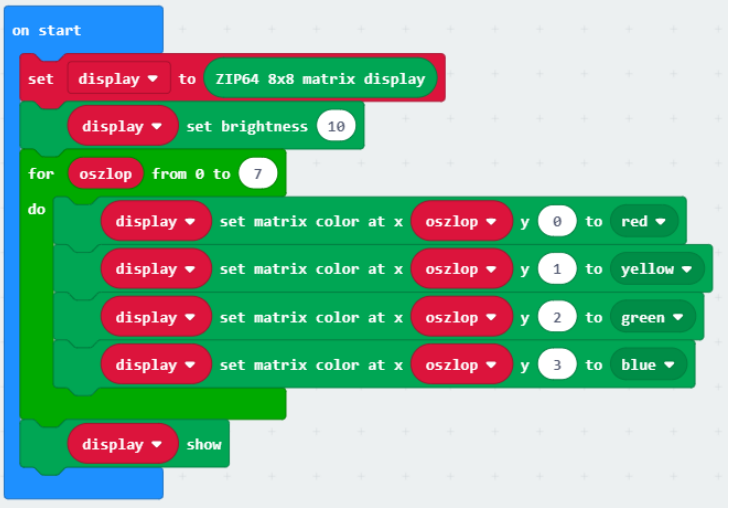
Az On start blokkunk tartalma ugyanaz, mint az előbbi esetben, ezért azt nem tüntettük fel.

3. LED-ek színeinek beállítása koordinátánként

Ebben a projektben soronként más-más színt állítunk be a LED mátrixon. Ezt úgy oldjuk meg, hogy a megfelelő koordinátájú pontokat átszínezzük. A LED mátrix 1. sora piros, a 2. sárga, a 3. zöld, a 4. kék lesz. A koordináták 0-tól indexelődnek, így az 1. sor/oszlop a 0-s indexű, a 8. sor/oszlop pedig a 7-es indexű.

Amennyiben koordinátánként színezzük át a LED-eket, fontos, hogy a kirajzolás után a Show blokkot is helyezzük el, különben nem jelenik meg az eredmény.

A projekt elérhetősége:
https://makecode.microbit.org/_DpTUwa5efYoH

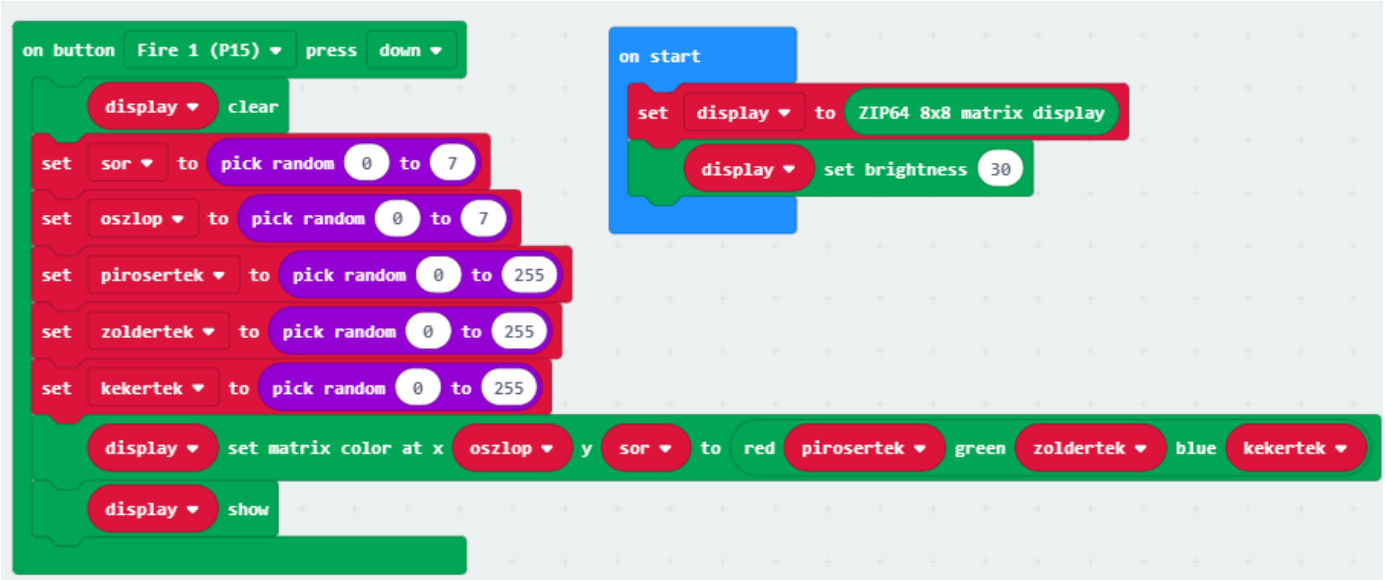


Készítette: Dr. Abonyi-Tóth Andor, ELTE Informatikai Kar

4. Véletlenszerűen meghatározott LED-el, véletlenszerű színnel

A következő projektben véletlenszerűen választott LED-et átszínezünk egy véletlenszerűen meghatározott színre. A színt RGB koordinátarendszer szerint adjuk meg. A vörös, zöld és kék komponens is 0 és 255 között fog egy véletlenszerűen meghatározott értéket felvenni.

A megjelenítés a Tűz 1 (Fire 1) gomb megnyomásakor történik meg.

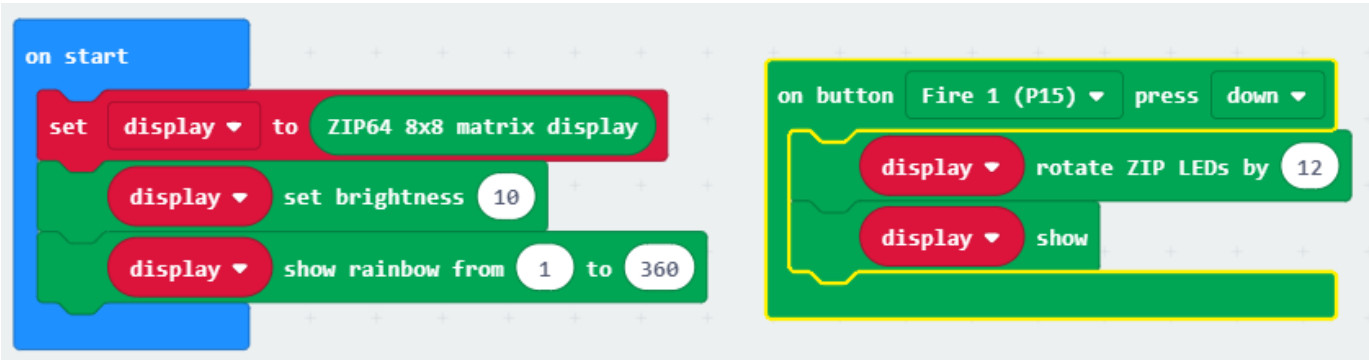


A projekt elérhetősége: https://makecode.microbit.org/_1FEUpH4d2ex9

5. Színek rotálása

A LED mátrixon megjelenő színeket akár rotálhatjuk is egy megadott értékkel.

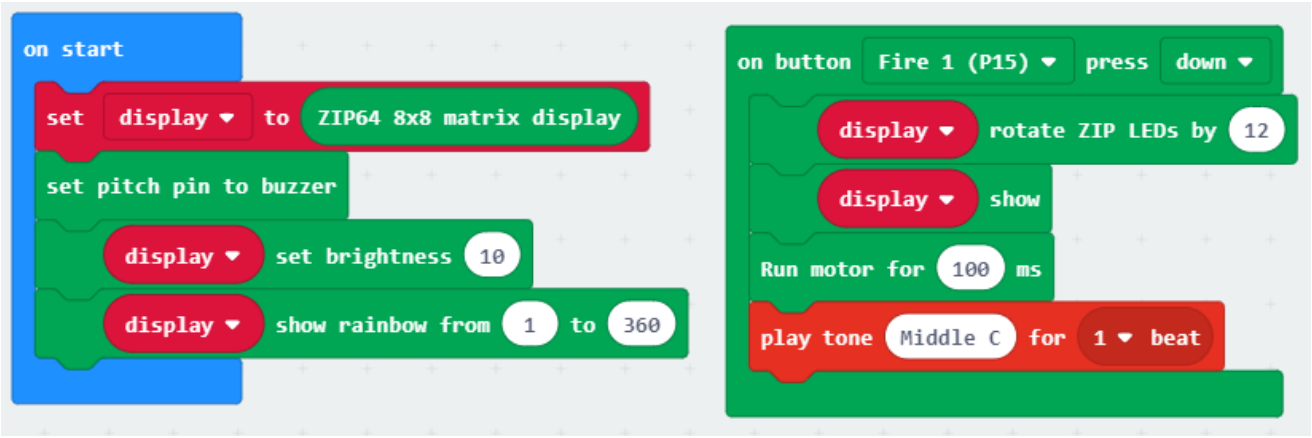
Ebben a projektben megjelenítjük a szivárvány színeit, majd a Tűz1 (Fire 1) gomb hatására 12 fokkal rotáljuk, így a színek minden gombnyomásnál meg fognak változni.



A projekt elérhetősége: https://makecode.microbit.org/_W8kaYP5dphDK

6. Rezgőmotor és zümmer használata

Az előző projektet továbbfejlesztjük úgy, hogy a Tűz 1 gomb megnyomásakor lejátszunk egy hangot, és a rezgőmotort is elindítjuk.
Ahhoz, hogy a hang a zümmeren játszódjon le, a **set pitch pin to buzzer** blokkot el kell helyezni a projektünkben.



A projekt elérhetősége: https://makecode.microbit.org/_P7fcFkLbmb3D

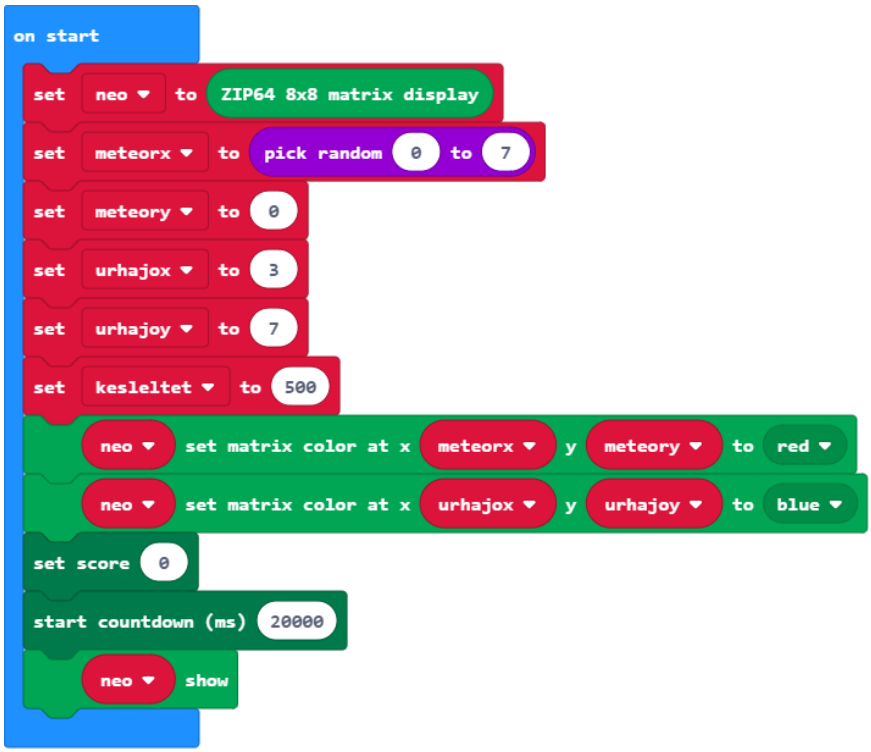
Komplex projekt elkészítése

Űrhajós játék elkészítése a GAME ZIP 64-re.

A szakköri anyagunk (http://microbit.inf.elte.hu/szakkori-anyag/) 65. oldalán ismertetett űrhajós játékot megvalósítottuk az eszközre hangolva.

Készítsünk egy játékot. A játék lényege: A kijelző alsó sorában jelenjen meg egy pont. Ez lesz az űrhajó. A kontroller bal és jobb gombjával lehessen balra és jobbra mozgatni. Fentről jöjjön meteor véletlenszerű helyről, amelyet ki kell kerülni. Amikor a meteor elérte az alsó sort, kerüljön a felső sorba véletlenszerű helyre. Ütközéskor érjen véget a játék. Számoljuk, hogy hány meteort kerültünk ki. 20 másodpercig tartson a játék.

Sajnos a :GAME ZIP kijelzőn nem működik a sprite-os megoldás, így többet kell programoznunk. Amiben ez a játék más, mint a szakköri anyagban, hogy a késleltetés idejét folyamatosan csökkentjük, így gyorsul a játék.



Kezdetben iniciáljuk a kijelzőt, amelyhez a neo változót használjuk.

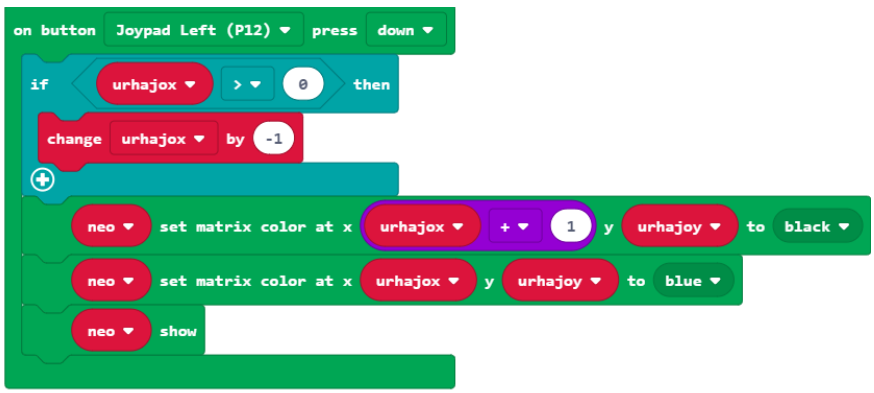
Beállítjuk a meteor és az űrhajó koordinátáit.

A késleltetés alap értéke legyen 500, ezt később csökkentjük, hogy nehezedjen a játékmenet.

Kirajzoljuk a meteort pirossal, az űrhajót kékkel.

A pontszámot nullázzuk, és elindítjuk a visszaszámlálót 20 másodpercről.

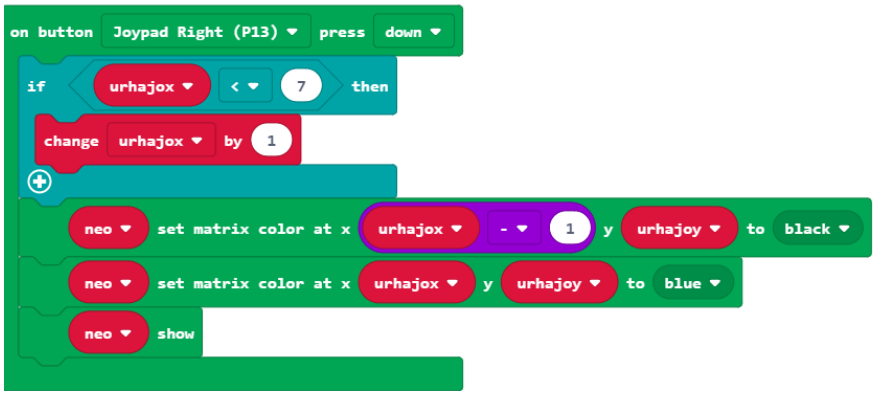
Megjelenítjük a változtatásokat a kijelzőn.



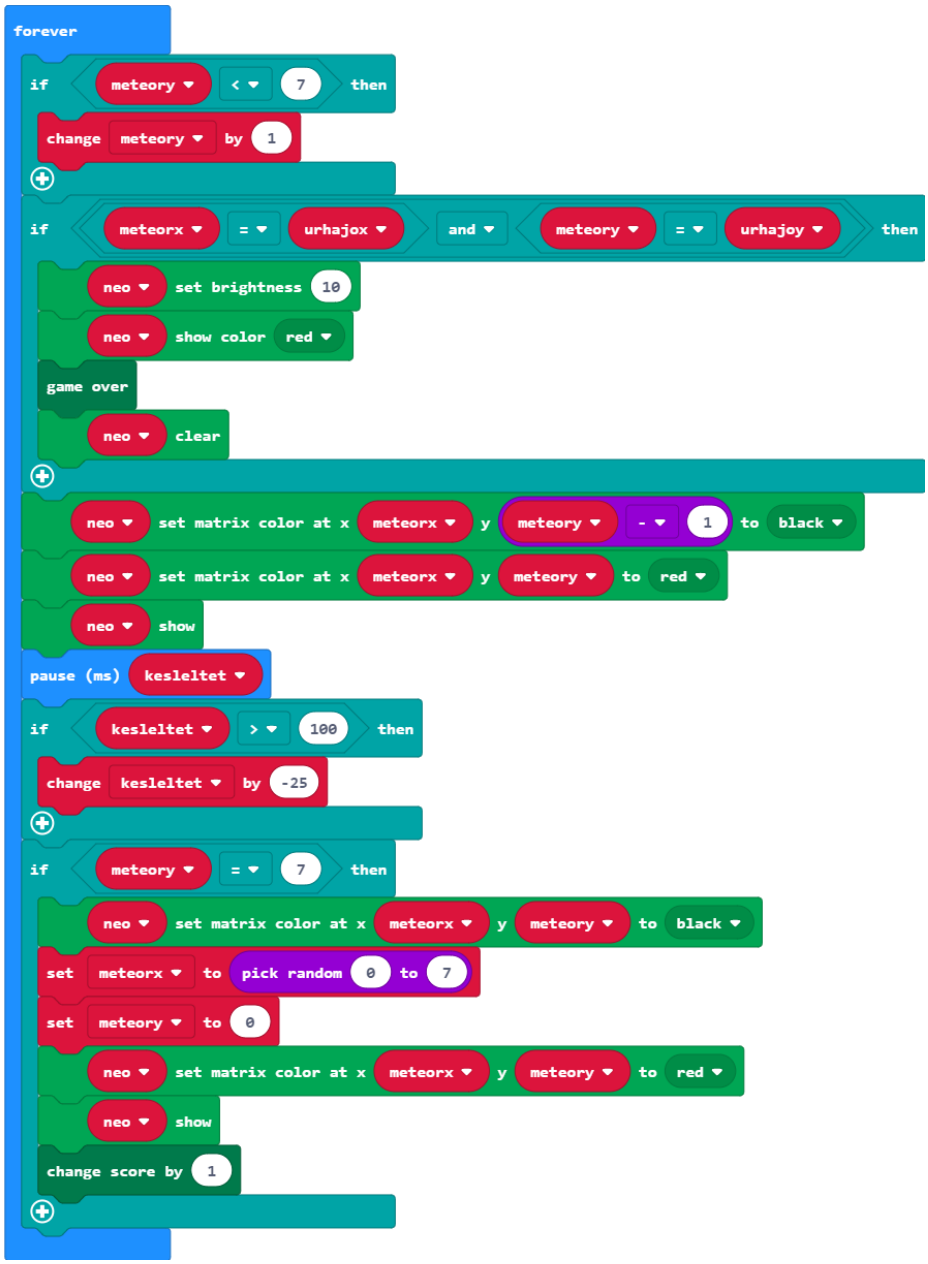
Beállítjuk, hogy Joypad balra gombjának megnyomásakor mi történjen. Csak akkor léphet balra az űrhajó, ha nem az 1. oszlopban áll (0-s indexű oszlop).

A régi helyéről letöröljük (feketével), az új helyén megjelenítjük, kék színnel.

Készítette: Dr. Abonyi-Tóth Andor, ELTE Informatikai Kar



Hasonlóan készítjük el a Joypad jobb gombjához tartozó blokkokat, csak itt azt vizsgáljuk, hogy csak akkor léphet jobbra az űrhajó, ha nem az utolsó oszlopban áll.



A forever blokkba írjuk, hogy mi történjen a játék során.

A meteort lefele mozgatjuk és vizsgáljuk, hogy a meteor ugyanazon a koordinátán van-e, mint az űrhajó. Ha igen, akkor ütközés történt, így véget ér a játék.

Ha nem történt ütközés, akkor a meteor régi helyét töröljük feketével, és megjelenítjük az új helyen.

A megadott késleltetési értékkel lassítjuk a játékot.

Csökkentjük a késleltetés értékét, így a következő kör már gyorsabb lesz.

Ha a meteor az alsó sorba ért ütközés néllkül, akkor eltüntetjük, és egy véletlenszerű helyre rakjuk az első sorban.

Növeljük a játékos pontszámát.

Az elkészült projekt elérhetősége: https://makecode.microbit.org/_1xAJqzVk0Vzw

Készítette: Dr. Abonyi-Tóth Andor, ELTE Informatikai Kar

Lehetséges továbbfejlesztési lehetőségek:

- Legyen az űrhajónak pajzsa. Ez jelentse azt, hogy pl. 2 ütközést kibír az űrhajó, és csak az utána lévő ütközéskor ér véget a játék!
- Legyenek hangeffektetek a játék során!
- Az űrhajó ne egy, hanem 3 egység széles legyen!
- Ütközéskor a rezgőmotor legyen elindítva!

Végszó

Ezt a kis füzetet azért állítottuk össze, hogy az eszközzel való ismerkedést megkönnyítsük. A későbbiekben egy jóval részletesebb szakköri anyaggal is készülünk. Addig is jó kísérletezést, játékfejlesztést kívánunk!

Készítette: Dr. Abonyi-Tóth Andor, ELTE Informatikai Kar