

LEGO robotok

X. rész

III.1.18. A Várj blokk

(Folytatás)

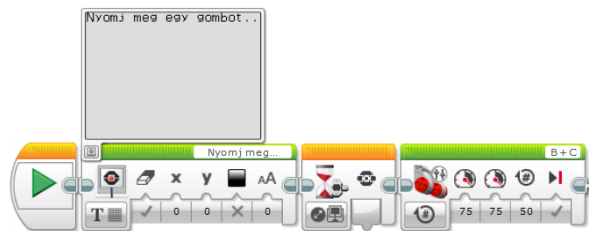
Változás módban a blokk a következőkre várhat:

- téglagombokra;
- színérzékelőre;
- infravörös érzékelőre;
- motor forgásra;
- időzítőre;
- érintésérzékelőre;
- üzenetre.

Változás módban a Várj blokk folyamatosan olvassa az adatokat az esetleges érzékelőkről vagy más komponensekről, és addig vár, amíg értékváltásra kerül sor, vagy egy általunk megadott értéket vesz fel a bemenet.

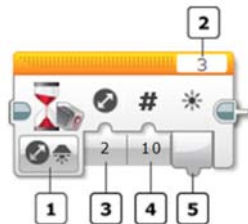
A következő érzékelők esetén a várakozás mód azt jelenti, hogy a program addig vár, ameddig az érzékelő a blokkba való belépés előtti értékhez képest bármilyen más különböző értéket érzékel, tehát megváltozik valami: téglagombok, színérzékelő szín módban, infravörös érzékelő távirányító módban, érintésérzékelő, üzenet szöveges vagy logikai módban. A Várj blokk visszatéríti a megváltozott – mért – értéket.

Például, a 68. ábrán látható programrészben a Várj blokk addig nem indítja el a robot motorjait, ameddig nem nyomtunk le egy akármilyen téglagombot.



68. ábra: Várakozás téglagomb lenyomására

Minden más esetben (színérzékelő visszavert és szórt fényerősség módban; infravörös érzékelő közelségi módban vagy irányjeladó haladási és közelségi módban; motor forgásérzékelője fok, fordulatszám vagy erősség módban; időzítő; üzenet numerikus adat módban) be tudjuk állítani a különbözőség irányát (nagyobb, kisebb, bármilyen) és küszöbértékét is. A Várj blokk visszatéríti a megváltozott – mért – értéket.



69. ábra: A Várj blokk változás módban

Az 1-es módszelektor segítségével ki tudjuk választani, hogy mire várjon a blokk – változás módban.

A 2-es gomb segítségével a portot állíthatjuk be (portszelektor).

A 3-as gomb segítségével állíthatjuk be a különbözőség irányát:

- 0 = nagyobb (Increase)
- 1 = kisebb (Decrease)
- 2 = bármilyen (Any).

A 4-es gombon tudjuk beállítani a küszöbértéket.

Az 5-ös gombon téríti vissza a blokk a mért (érzékel) értéket.

A 70. ábrán látható programrészen a Várj blokk addig nem indítja el a robot motorjait, ameddig a szobában a környezeti (szórt) fény erőssége 10 egységgel nagyobb nem lesz, mint amekkora volt a blokkba való belépéskor. Tehát ha belépünk a szobába és felkapcsoljuk a villanyt, akkor a robotunk beindul.



70. ábra: Várakozás a villany felkapcsolására

Üzenetek esetén a Várj blokknak létezik egy *frissítés* (Update) módja is.

A blokk ekkor addig vár, ameddig be nem érkezik a megfelelő típusú (szöveg, numerikus, vagy logikai) üzenet az üzenet címével (fejlécével) együtt. A blokk kimenetén megjelenik a beérkezett üzenet.



71. ábra: Üzenetek frissítés módja

III.1.19. A Ciklus blokk

A programozási nyelvek külön utasításosztályát képezik a ciklusszervező, iteratív számításvezérlő utasítások. Az osztály két lényeges alosztályra bomlik: a *rögzített lépésszámú* és a *változó lépésszámú ciklusokra*. A rögzített lépésszámú ciklusok az eleve megadott lépésszámig ismétlik a végrehajtandó utasításokat, a változó lépésszámú ciklusok pedig addig ismétlenek, ameddig egy megadott feltétel igaz (ha a ciklus *előltesztelés*), vagy hamis (ha a ciklus *hátultesztelés*). A feltétel logikai értékének módosulása maga után vonja a ciklus befejezését. Amennyiben például egy hátultesztelés ciklus esetében a feltétel mindig hamis, *végtelen ciklusról* beszélünk, hisz az ismétlés soha nem fog leállni.

A ciklusokban egy *ciklusváltozó* mondhatja meg az ismétlések számát, vagyis azt, hogy éppen hányadik ismétlésnél tartunk.

A végrehajtandó utasításokat a ciklus *magjának* nevezzük. A ciklus magját el kell határolni a többi utasítástól. A ciklus befejezése után a mag utasításai többet nem hajtódnak végre, hanem a vezérlés a ciklust követő utasításokkal folytatódik.

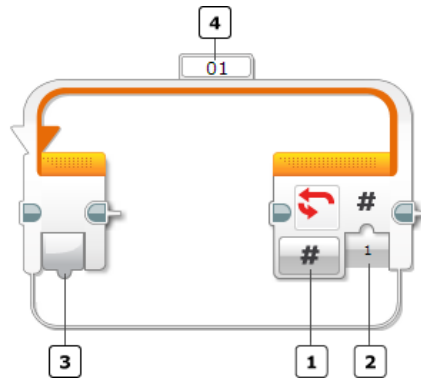
Növekménynek vagy *lépésnek* nevezzük a ciklusváltozót módosító értéket.

Az 1-es módszelektor segítségével tudjuk kiválasztani a ciklus típusát, megállási feltételét.

A 2-es gomb segítségével a ciklus bemenetét (bemeneteit) tudjuk megadni.

A 3-as gomb a ciklusváltozó értékét adja vissza.

A 4-es gomb segítségével szimbolikus nevet adhatunk a ciklusunknak, így hivatkozási alapot teremthetünk a ciklusra, amelyet később más blokkokban (például ciklusbefejező blokk) felhasználhatunk.

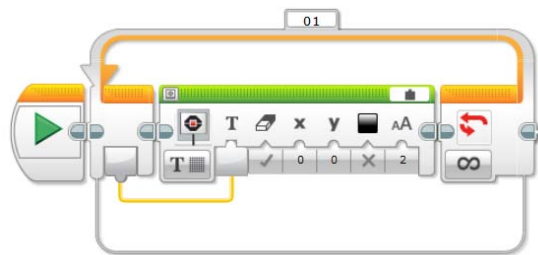


72. ábra: *Ciklus*

Végtelen ciklus

A végtelen ciklus olyan ciklus, amelynek futása külső esemény bekövetkezése nélkül sohasem zárulna le. Egy ilyen külső esemény például a téglá Vissza (Back) gombjának a megnyomása, amellyel kilépünk a programból.

A 73. ábrán egy végtelen ciklust hoztunk létre úgy, hogy a módszelektort *végtelenre* (Unlimited) állítottuk. A ciklus a végtelenségig ismétlődik, és kiírja a robot képernyőjére a ciklusváltozó egyre növekedő értékeit. A ciklust csak a program bezárásával lehet leállítani, ha nem, addig működik, ameddig a robotból ki nem fogy az elem.

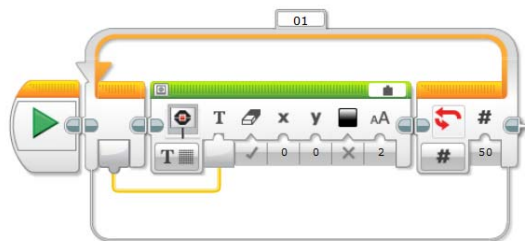


73. ábra: *Végtelen ciklus*

Rögzített lépésszámú ciklus

A rögzített lépésszámú ciklus szervezéséhez a módszelektorban válasszuk ki a *számol* (Count) beállítást. Így megjelenik egy gomb, amely segítségével megadhatjuk, hogy a ciklus hányszor iteráljon.

A 73. ábrán látható végtelen ciklust könnyű átírni rögzített lépésszámú ciklussá. A 74. ábrán látható ciklus 50-szer fogja kiírni a ciklusváltozó értékét, vagyis elszámol 0-tól 49-ig. Megjegyzendő, hogy a ciklus automatikus ciklusváltozója mindig 0-tól indul.



74. ábra: Rögzített lépésszámú ciklus

Időciklusok

Lehetőség van időciklusok szervezésére is. Ha a módszelektorral az *idő* (Time) beállítást választjuk, akkor a ciklus a másodpercben megadott időegységig fog futni. Az eltelt időt mindig a ciklusmag végrehajtása után teszteli, és ha az idő kisebb, mint a beállított érték, akkor még egyszer végrehajtja a ciklusmagot.

Változó lépésszámú ciklus

Logikai feltételhez kötött változó lépésszámú ciklust úgy tudunk szervezni, hogy a módszelektor *logikai* (Logic) beállítását választjuk. Így, hátultesztelés ciklus lévén, mindannyiszor végrehajtja a ciklusmagot, ameddig a megadott logikai feltétel hamis. Amint a logikai feltétel igazgá válik, a ciklus leáll. Vigyázat, mert, ha a logikai feltételt úgy adjuk meg, hogy az mindig hamis, végtelen ciklusunk lesz!

Megfigyelhető – mivel hátultesztelés ciklusunk van –, hogy a ciklusmag egyszer mindenképp végrehajtódik, mert csak a végén teszteli a ciklusunk a logikai feltételt. Előltesztelés ciklus szervezésére nincs lehetőség, csak ha *elágazó* (Switch) utasítást használunk. Ennek használatával később ismerkedünk meg, de az elv az, hogy teszteljük a logikai feltételt, és ha az már eleve igaz, nem lépünk be a ciklusba.

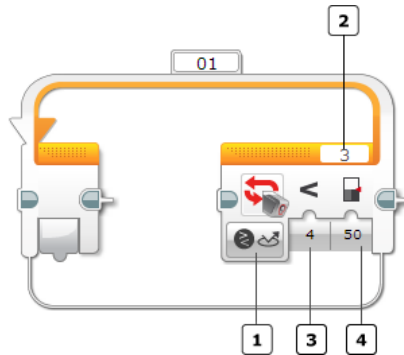
Logikai feltételhez kötött ciklust kell használnunk akkor is, amikor egynél több érzékelő adataiból következtetve szeretnénk ismételni utasításokat, hisz a ciklusszervezésben csak egy érzékelő által szolgáltatott visszatérési érték felhasználása megengedett.

Érzékelők által vezérelt ciklusok

A ciklus blokk több olyan módot is tartalmaz, amely segítségével be lehet olvasni egy megadott szenzor értékét, és ezt össze lehet vetni (hasonlítani) egy megadott értékkel. A ciklus addig fog tartani, ameddig értékegyezés nem lesz.

Ebben az esetben a 75. ábrán látható a ciklus blokk általános alakja.

Az 1-es módszelektör segítségével tudjuk kiválasztani a ciklus típusát.
 A 2-es gomb segítségével tudjuk kiválasztani az érzékelő portját (port szelektor).
 A 3-as gombbal adhatjuk meg az összehasonlító műveletet (a 16. táblázat szerint).
 A 4-es gomb segítségével pedig a küszöbértéket állíthatjuk be.



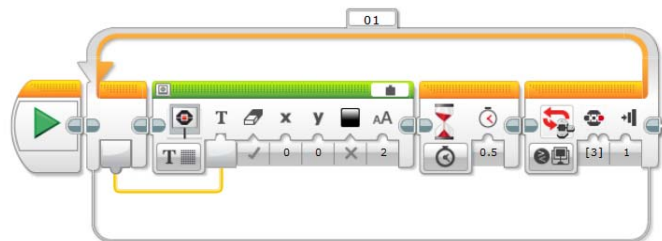
75. ábra: Érzékelők által vezérelt ciklusok

A ciklus blokk a következő érzékelőket ismeri:

- téglagombok;
- színérzékelő;
- infravörös érzékelő;
- motor forgás;
- időzítő;
- érintésérzékelő;
- üzenet.

Értelemszerűen léteznek olyan érzékelők is, amelyeknél nincs összehasonlító művelet, például a színérzékelő szín módja esetén a ciklus akkor állhat le, amikor a színérzékelő egy adott színt érzékelt. Ez az érzékelt szín nem lehet egy küszöbérték, vagyis nincs például a pirosnál nagyobb vagy kisebb szín.

A 76. ábrán látható ciklus addig írja ki a ciklusváltó értékeit, ameddig a középső téglagombot benyomott állapotban nem találja.



76. ábra: Gombnyomásig ismét

Kovács Lehel István