**Robotprogramozás**

Jegyzőkönyv

Hallgatók:

Balogh Dániel (QPHXZJ)

Horváth Zsolt (O3CCCQ)

Menyhárt Balázs (BDVUD4)

Pinczés Péter (GYOOWR)

Mérésvezető:

Drexler Dániel András

Mérés időpontja:

2016.03.08.

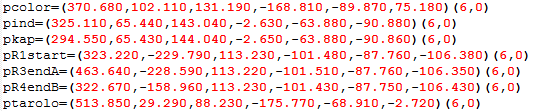
**Bevezetés**

Napjainkban a gyártástechnológiában rendkívül fontossá vált a robotok alkalmazása. Bizonyos feladatok esetén a robotok használata több előnnyel is bír az emberi munkaerővel szemben. Egyrészt a robotkar nem fárad el, nem tart pihenőt, könnyebben elvégzi a nehéz fizikai munkát. Ennek köszönhetően sokkal gyorsabb munkavégzésre képes, mint az ember. Ezenkívül pontosabb, precízebb is a működése, sokkal kisebb a hibafaktor. Hátránya azonban, hogy nem tudnak bármilyen munkát elvégezni, fontos hogy a feladatuk automatizálható, monoton legyen. A leggyakrabban anyagmozgatásra használnak robotkarokat, a mérés célja is ennek megvalósítása volt. A robotnak háromféle munkadarabot (fa, alumínium, réz) kell a tárolóból a megfelelő rekeszbe áttennie. Mindegyik fajtából négy darab van, amelyeket összesen hat darab két elem tárolására elegendő rekeszbe kell szállítani. A munkadarabok anyagát különböző érzékelők segítségével lehet megállapítani.

**A pontok betanítása**

Az első feladatunk, hogy a robotnak betanítsunk bizonyos, a munkája szempontjából fontos pontokat. Ez azt jelenti, hogy a kart a betanító pult segítségével a kívánt pozíciókba kell mozgatni, majd az aktuális helyzetben a koordinátákat ú.n. pozícióváltozóként kimenteni és eltárolni. Azért van rá szükség, mert ez sokkal egyszerűbb megoldás, mint mindig újra és újra megadni a koordinátákat.

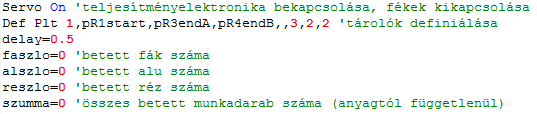
A feladat elvégzéséhez összesen hét pontot kellett a robotnak betanítani, a három érzékelő pozícióját, a hat célrekesz közül három sarkon lévőnek a pozícióját (ezek alapján kiszámítható a többi célrekesz pozíciója is), valamint a tároló pozícióját, ahonnan a robotkar a munkadarabokat kiveszi. Fontos, hogy a pozícióváltozók neve p betűvel kezdődjön.



1. ábra: a pozícióváltozók értékei

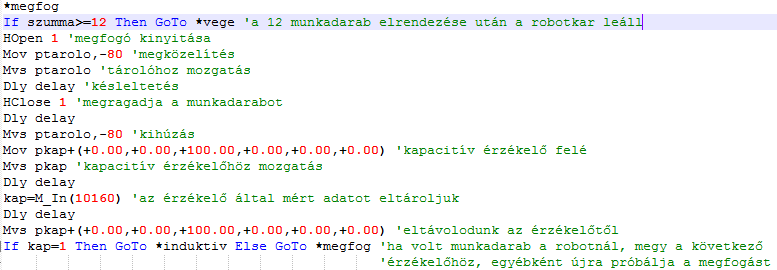
**A robot programozása**

Miután megvannak a szükséges pozícióváltozók, a következő feladat a robot megfelelő mozgását, viselkedését biztosító program megírása. A programot MELFA BASIC V nyelven kell megírni. A program elején be kell kapcsolni a teljesítményelektronikát, valamint kiengedni a fékeket. Ezt a servo on utasítás végzi. Ezenkívül definiáltunk néhány szükséges skalárváltozót, valamint a célrekeszek struktúráját.



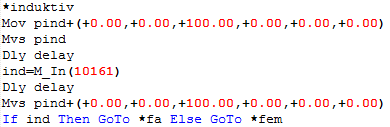
2. ábra: a program eleje

A robotnak először ki kell vennie egy munkadarabot a tárolóból, majd a kapacitív érzékelőhöz kell tartania, ami megvizsgálja, hogy valóban van-e munkadarab a megfogóban. Ha van, akkor tovább kell vinnie az induktív érzékelőhöz, ha nincs, akkor újra elvégeznie ezt a műveletsorozatot. A kódban látható, hogy a tárolóhoz és az érzékelőhöz való mozgatás előtt a robot mindig egy bizonyos távolságra megközelíti ezeket az elemeket, majd miután elvégezte ott a feladatát, ugyanígy el is távolodik tőlük. Ez biztonsági szempontból fontos, az ütközések elkerülése végett. Szintén fontos, hogy az egyes műveletek elé és mögé, mint a megfogás és az érzékelés, kell egy késleltetést rakni, hogy a robot ne végezze túl gyorsan a következő mozgást, és legyen ideje befejezni az adott műveletet.



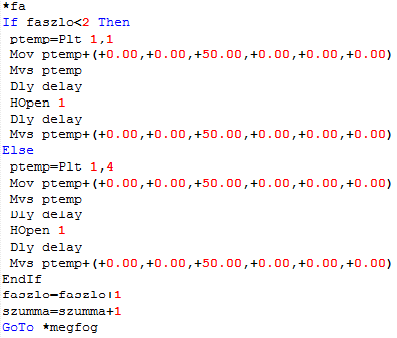
3. ábra: megfogás és kapacitív érzékelő

A következő lépés az induktív érzékelés. Az induktív érzékelő azt állapítja meg, hogy a robotnál lévő munkadarab fémből vagy fából készült. Ha fémből, akkor a munkadarabot a színérzékelőhöz kell mozgatni, ugyanis kétféle fém munkadarab is van, amelyeket a színük alapján könnyen meg lehet különböztetni. Ha az anyag fa, akkor lehet a neki megfelelő rekeszbe tenni.



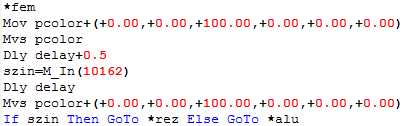
4. ábra: induktív érzékelő

Ha a munkadarab anyag fa, akkor szállítsa a robot a megfelelő rekeszbe. Fa tárolására két rekesz szolgál. Mindkettő rekesz két elemet tud tárolni, és összesen négy munkadarab készült fából. Tehát ha az első rekesz megtelt, utána a karnak a második rekeszbe kell tennie a tárgyakat. Erre a faszlo nevű számlálót használjuk. Továbbá növelni kell a szumma nevű számlálót is, ami az összes helyretett munkadarabot számolja. Ha kész a művelet, visszaugrunk a \*megfog címkéhez.



5. ábra: fa munkadarab

Ha a munkadarab fém, meg kell állapítani, hogy alumíniumból vagy rézből készült-e. Ez a színérzékelő feladata.



6. ábra: színérzékelő

Ezután ugyanazt kell tenni, mint a fa munkadarabok esetében, az anyagnak megfelelő rekeszbe tenni a munkadarabot, ugyanúgy növelve az anyagnak megfelelő számlálót, valamint az összes munkadarab számlálóját. A kódok bemásolása felesleges, ld.: 5. ábra más rekeszekkel.

Amikor a szumma nevű számláló értéke eléri a tizenkettőt (3. ábra), a program a \*vege címkéhez ugrik. Itt már csak egy servo off utasítás található, amely kikapcsolja a teljesítményelektronikát és bekapcsolja a fékeket, valamint az end utasítás, ami a program végét jelzi.

**A működés ellenőrzése, konklúzió**

A program megírása után ellenőrizni kell annak helyes működését. A biztonság kedvéért célszerű először lépésenként, kis sebességgel haladni. Ha a robotkar a megfelelő működést produkálja, akkor el lehet indítani a programot folyamatosan, nagy sebességgel.

A mérés során tapasztalhattuk, hogy a robot használata valóban gyorssá és pontossá teszi az automatikusan végezhető munkákat. Azonban van pár dolog, amire nagy figyelmet kell fordítani. Az egyik negatív eredmény, amit tapasztaltunk, hogy az érzékelők (főleg a színérzékelő) nem működtek kellőképpen nagy biztonsággal. Ezért fontos az érzékelők megfelelő kiválasztása, valamint ügyelni kell rá, hogy a munkadarabot elégséges ideig tartsuk az érzékelő közelében, továbbá színérzékelő esetén a kedvező fényviszonyok beállítására.

A másik fontos tapasztalat, hogy a kódot nagy körültekintéssel kell megírni, mert egyetlen apró hiba is nagy károkat okozhat. A mérésen egy apró hiba egy kis koccanáshoz vezetett (a robotkar hozzáért a tárolóhoz, miközben a célrekesz felé vitt egy munkadarabot), ami épp annyira mozdította el a tárolót, hogy a következő megfogási fázisnál a robotkar megfogója a tárolóba ütközött és letört. Ez esetünkben nem okozott nagy kárt az olcsó eszközök használata miatt, de egy valódi gyártósoron egy ilyen hiba óriási károkat okozhat, ezért a kód megírását és tesztelését mindig nagy figyelemmel és a teljes működésre kell elvégezni.