FIATAL MŰSZAKIAK TUDOMÁNYOS ÜLÉSSZAKA XIX.

Kolozsvár, 2014. március 20–21.

3D MODELL ROBOTKAR MOZGATÁSÁNAK BETANÍTÁSA IPARI ESZKÖZÖKKEL

3D MODEL OF ROBOT ARM MOVING AND TEACHING BY INDUSTRIAL DEVICES

ZILAHÍ Krisztián László⁽¹⁾, TÓTH János⁽²⁾

(1) Debreceni Egyetem, Műszaki Kar, Villamosmérnöki és Mechatronikai Tanszék, 4028, Magyarország, Debrecen, Ótemető utca, 2-4, imkrisztian@hotmail.com

(2) Debreceni Egyetem, Műszaki Kar, Villamosmérnöki és Mechatronikai Tanszék, 4028, Magyarország, Debrecen, Ótemető utca, 2-4; Telefon / Fax: +36 (52) 415-155/77742, tothjanos@eng.unideb.hu

Abstract

Processions controller devices and Human-Machine Interfaces (HMI) are quite important in automated systems. They provide the necessary informations about the system's functioning for the user. The aim of this paper was to create communication between humans and machines with industrial devices, which is completed with a teachable system. In this case the teachble system is a 3D model of robot arm, controlled by a Programmable Logic Controller (PLC). Teaching is a greatly important part of industrial systems. It makes easier the programming of sequential or time based controlling. The machine-operator have to control and move the system manually only once, and then the system can make the process and moves automatically.

Keywords: robot arm, PLC, HMI, programming, teaching

Összefoglalás

Az automatizált rendszerekben fontos szerepet töltenek be a folyamatokat irányító, szabályozó berendezések, valamint az ipari megjelenítő egységek (HMI), melyek a felhasználó számára biztosítják a berendezés működtetéséhez szükséges információkat. A dolgozat célja az iparban használt eszközökkel megvalósított ember-gép kommunikáció létrehozása volt, melyet egy betanítható rendszerrel egészítettük ki. A betanítható rendszer - ebben a cikkben - egy programozható logikai vezérlővel (PLC) irányított 3D robotkar modell. A betaníthatóság fontos szerepet tölt be az ipari rendszereknél. A sorrendi és idő alapú vezérlések esetén megkönnyítheti a folyamatok, mozgások leprogramozását. Elegendő a gépkezelőnek egyszer kézzel végig irányítani, mozgatni a rendszert, majd az automatikusan ismétli azt.

Kulcsszavak: robotkar, PLC, HMI, programozás, betanítás

1. A felhasznált 3D robotkar modell kinematikája

A cikk során a Fishertechnik nevű cég által gyártott építőelemekből felépített robotkar modellt használtuk. A vezérléséhez szükség van a robotkar kinematikájának ismeretéhez. A robotkar kinematikai láncá szerinti egy RTT típusú, amely egy rotációs és az ahhoz csatlakozó

csuklóval és két translációs ízülettel rendelkezik. A kinematikai lánc nyílt, melynek végén egy megfogó található.

1. ábra. A felhasznált RTT - típusú robot kinematikai láncja és munkatere

Az RTT típusú robotok munkatere egy hengergyűrű (1. ábra), tehát henger koordináta rendszerben dolgozik. A robotkar kinematikájából adódóan érdemes az O_0 ponthoz rögzíteni a bázispontot, mely ponthoz csatoljuk a világi koordináta rendszert, valamint a robot koordináta rendszert. A robot koordináták meghatározzák a robot manipulátor effektorának (megfogó) helyzetét. Ennek pozíciója a Descartes-féle koordináta rendszer koordinátái (x, y, z) segítségével, írható le, orientációja a módosított Euler-szögekkel adható meg (φ, θ, ψ) . Az ízületek koordináta rendszereit a Denavit-Hartenberg konvekció alapján vettük fel. A felvett koordináta rendszerekre felírjuk az egyes ízületekhez tartozó Denavit-Hartenberg transzformációs mátrixokat (1) [1].

$${}^0D_1 = \begin{pmatrix} \cos(\theta_0) & -\sin(\theta_0) & 0 & 0 \\ \sin(\theta_0) & \cos(\theta_0) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} {}^1D_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{pmatrix} {}^2D_3 = \begin{pmatrix} 1 & 0 & 0 & d_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (1)$$

Ezután az O_0 rögzített ponthoz kötött Denavit-Hartenberg féle homogén transzformációs mátrix (2) az egymást követő koordináta rendszerek szorzatával határozható meg.

$${}^0T_3 = {}^0D_1 \cdot {}^1D_2 \cdot {}^2D_3 = \begin{pmatrix} \cos(\theta_0) & -\sin(\theta_0) & 0 & d_3 \cdot \cos(\theta_0) \\ \sin(\theta_0) & \cos(\theta_0) & 0 & d_3 \cdot \sin(\theta_0) \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2)$$

2. A mozgatáshoz és betanításhoz használt ipari eszközök

2. ábra. A mozgatáshoz és betanításhoz használt ipari eszközök

A robotkar mozgatásához olyan programozható eszközöket használtunk fel, amelyek segítségével a kezelő vezérelni tudja a megépített robotkارت. A felhasználó számára a rendszerrel való kapcsolatot Phoenix Contact TP 12T típusú HMI biztosítja. A HMI és a robotkar között egy Phoenix Contact ILC-150ETH típusú PLC tartja a kapcsolatot, a PLC digitális kimenetei a motort kapcsoló relék tekercseihez vannak bekötve, a roboton lévő mikrokapcsolók jelei a PLC digitális bemeneteibe érkezik. A HMI és PLC kapcsolata egy INTERBUS - OPC (Object Linking and Embedding for Process Control) szerveren keresztül történik, TCP/IP protokollon keresztül, mely egy ipari kommunikációs szabvány [2].

3. A megjelenítő egységre létrehozott menüstruktúra

A HMI felületéről a kezelő információkat kap a rendszerről, valamint utasításokat is adhat. A HMI által továbbított információkat és parancsokat, valamint a robotkaron lévő szenzoroktól származó adatokat is a PLC-re megírt program dolgozza fel.

3. ábra. A létrehozott menüstruktúra és a hozzá tartozó kezelőfelületek

A kezelő felületet (3. ábra) menüpontokra tagoltuk. Figyelembe véve a különböző pontok jelentőségét a felhasználó szempontjából. Az első menüpont az „Alaphelyzet felvétel”, melyben a robotkar beáll az alaphelyzetébe, hogy az útmérő rendszerét bekalibrálja. A „Kézi mozgatás” pont alatt lehet a robotot manuálisan irányítani, megismerkedni a mozgatásért felelős felülettel. A „Kézi programozás” funkcióban a felületen manuálisan megírt program vihető be a PLC-nek. A „Betanító üzemmód” esetén a már ismert „Kézi mozgatás” felületére épülő struktúrával mozgatható a robot, kiegészítve egy betanító funkcióval, mellyel tetszőleges mozdulatsorra (tengelyenként 1000 pozícióra) tanítható meg. „A megfogó helyzete” funkció a „Kézi programozás” és a „Betanító üzemmód” menüpontokból érhető el. Itt a robotkar effektorának mozgása követhető grafikonokon, melyet egy piros kereszt jelöl.

4. A betanítás módszere

Kielemezve a tengelyek mozgásának időbeni változásának lehetőségeit arra a következtetésre jutottunk, hogy elegendő két feltételt vizsgálni a mozdulatsor megjegyzéséhez. A számláló mechanizmus miatt, a tengely pozíció időbeni változásának értéke nem folytonos (lépcsős függvény).

Az első feltétel, amikor a tengely egy pozíción áll adott ideig (4. ábra I. rész). Ezt úgy vizsgáltuk, hogyha a tengely koordináta értéke nem változik $t = 0,5$ s-on belül, akkor a pozíciót (p_{m1}) és a pozíción töltött időt (t_{m1}) kell megjegyezni. A második feltétel szerint

lokális szélső értékeket kell keresni (4. ábra II. rész), azaz, ha megváltozik a tengely mozgásának iránya, akkor a tengely koordináta – idő grafikonon lokális minimumok vagy maximumok keletkeznek. Itt elegendő csak a szélsőérték koordinátáját (p_{m2}) megjegyezni, $t_{m2} = 0$ s-ot rendelve hozzá. Ez a pozíció úgy határozható meg, hogy figyelni kell az utolsó két (p_{m2-1} , p_{m2}) és az aktuális pozíciót (p_{m2+1}). Ha teljesül a $p_{m2-1} = p_{m2+1}$, akkor a p_{m2} koordinátát kell megjegyeztetni.

4. ábra. A pozíció rögzítésének lehetséges feltételei

A feltételeket figyelembe véve elkészítettük az azokat figyelő programrészletet. A feltételek teljesülését egy számlálóval (rot_CTU1) számolom. A tanulási folyamat (5. ábra) alatt (tanul_rot), az események bekövetkeztével növeljük a számláló értékét (rot_CTU1.CV). A koordináták és idők mentésére tengelyenként egy-egy tömböt definiáltunk (pl.: memory_pos_rot, memory_time_rot). A tömbökbe való mentés során a számláló értékének megfelelően a tömbök sorindexe (i) változik, így a megjegyzendő értékek a tömbök aktuális sorindexű sorába menthetők. A tárolt adatokat a visszajátszásért felelős programrész használja fel.

```

17 rot_CTU1(CU:=(tanul_rot AND (rot_rtrig.Q OR rot_rtrig2.Q)),
18 RESET:= NOT(tanul_rot), PV:= 1000);
19 i:=rot_CTU1.CV;
20 IF rot_rtrig.Q THEN
21   memory_pos_rot_1[i] := rot_M2;
22   memory_time_rot_1[i] := t#0s;
23 END_IF;
24 IF rot_rtrig2.Q THEN
25   memory_pos_rot_1[i] := rot_M1;
26   memory_time_rot_1[i] := (rot_ido + t#0.5s);
27 END_IF;

```

5. ábra. A koordináták és a hozzátartozó idők mentési folyamata

5. Összefoglalás

Elkészítettünk egy megjelenítő alkalmazást, melyen keresztül létre jött a felhasználó és a rendszer közötti kommunikáció. Az alkalmazás segítségével a kezelő igénybe tudja venni az általunk megírt különböző funkciókat, mint például a robotkar kézi vezérlése, betanítása, programbevitel. Az általunk megvalósított rendszer vezérlési része alkalmas a hasonló feladatok ellátására, így kiválóan felhasználható ismétlődő mozgássorok megtanítására és azok visszajátszására [3].

Irodalom

- [1] Spong Mark W., Hutchinson Seth, and Vidyasagar M. Robot Modeling and Control. New York: John Wiley and Sons, 2005. 496 pp. ISBN: 9780471649908
- [2] Ajtonyi István: Ipari kommunikációs rendszerek I. 1. kötet. kiadó: AUT-INFO Kiadó, Miskolc, 2008. 431 p. ISBN szám: 9789630658133
- [3] 3D modell robotkar mozgásának betanítása ipari eszközökkel – videó:
<http://www.youtube.com/watch?v=we8ZYFfWjUs>