

Oktatási tesztpanel tervezése FANUC M1iA ipari válogató robot számára

Murvai Dávid
Mechatronikai Tanszék
Debreceni Egyetem, Műszaki Kar
Debrecen, Magyarország
murvaiddavid92@gmail.com

Erdei Timotei István
Mechatronikai Tanszék
Debreceni Egyetem, Műszaki Kar
Debrecen, Magyarország
timoteierdei@eng.unideb.hu

Husi Géza
Mechatronikai Tanszék
Debreceni Egyetem, Műszaki Kar
Debrecen, Magyarország
husigeza@eng.unideb.hu

Absztrakt— A projekt során oktatási célokra használható tesztpanel került megtervezésre és megépítésre, ami egy a Debreceni Egyetem, Épületmechanika Kutató Központjában található, FANUC M1iA 0.5A ipari delta-tripod válogató robot oktató paneljeként szolgál. A tesztpanel Atmel mikrokontroller vezérte, amihez nyomtatott áramkörök kerültek elkészítésre. Ez a panel hozzájárul, hogy a hallgatók megismerkedhessenek az ipari robotok & mikrokontrollerek vezérlésével.

Kulcsszavak—FANUC, Arduino, ipari robot, mikrokontroller, programozás

I. BEVEZETŐ

Manapság a vezérlés oktatása nagymértékben támaszkodik a szimulációs csomagokra és virtuális laboratóriumokra, mivel ezek nélkülözhetetlenek a folyamatok megismerése során. Sajnos sokszor kizárólag erre a virtuális megvalósításokra vannak korlátozva a folyamatok, és a tanulók nem kapnak semmiféle fizikai visszajelzést. Ennek oka lehet a magas költség, ami egy oktatási célra szánt vezérlő rendszer beszerzésével és telepítésével jár. A kutatás/fejlesztésnek a Debreceni Egyetem adott otthont [13][14].



1. ábra: Hardware in the loop [5]

Erre jelenthet megoldást az Arduino fejlesztő [1] platform, amely segítségével könnyen bemutathatóak a vezérlési feladatok/rendszerek, igen sokrétűen, alacsony költségek mellett.

II. ELŐZMÉNYEK

A projekt alaptézise volt létrehozni egy olyan tesztpanel, amelyet Atmel mikrokontroller vezérel, és ami egy rendszerbe illeszhető együttműködni képes a Mechatronikai Tanszék, robot laborjában lévő FANUC M1iA 0.5A robottal.

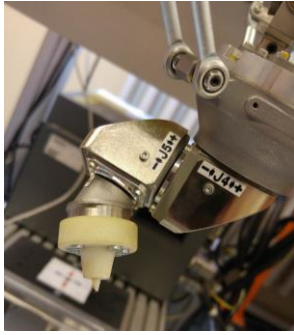


2. ábra: FANUC M1iA 0.5A robot [3]

A FANUC robot egy delta-tripod kialakítású válogató egység és a kisebb ipari robotok osztályába tartozik, kis munkaterülettel rendelkezik, főleg válogató, szortírozó feladatokat lát el. Tömege 23 kilogramm, ezért – és kis méretének köszönhetően – könnyen telepíthető a legtöbb helyre, 6 tengellyel rendelkezik. 280 mm-es maximális kinyúlásával, 4000 mm/s-os sebességével és +/- 0,02 mm-es ismétlési pontosságával a kis helyigényű, gyors, precíz munkavégzési körökben alkalmazzák, például az mikroelektronikai-, optikai- és gyógyszeriparban [2].

A tanszéken lévő robot anyagmozgatási feladatokat nem lát el, mivel még nem került felszerelésre a vákuumos megfogó egység, amivel be lehetne mutatni iparban betöltött funkcióit. Jelenleg egy 3D nyomtatott rugós tüskével van felszerelve. A fejegységet tekintve SLA [6], folyadék alapú adalék anyag 3D nyomtatóval készült.

A fejegységgel vonalvezetési/pályabejárési, feladatokat mutathatunk be oktatási céllal [15].



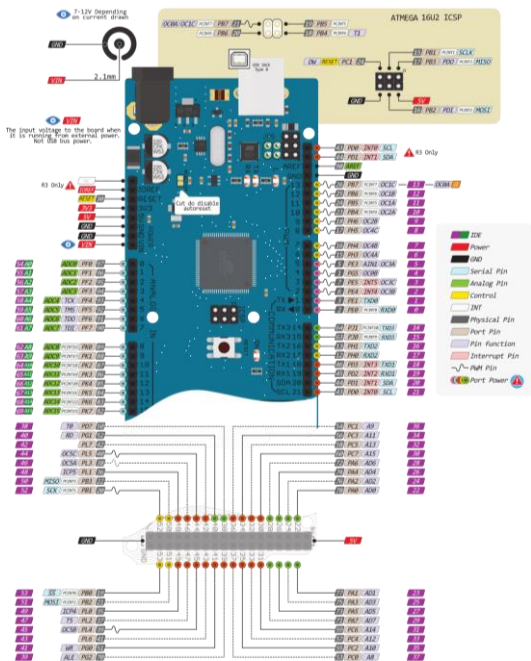
3. ábra: FANUC M11a 0.5A fejegegység

Erre jelent megoldást az megtervezett tesztpanel, amelyen nyomógombok segítségével lehet bemutatni a robot mozgását, mely során fizikai kontakt is történik.

A tesztpanel oktatási célokat szolgál, általa nyerhetnek bepillantást a hallgatók a FANUC robot működésébe, mozgásának megismerésébe, és nem utolsósorban az Arduino mikrokontroller általi vezérlésbe.

Az Arduino alapú vezérlésnek rengeteg előnye van. Maga a hardware olcsó és könnyen beszerezhető. Az oktatásban is egyre nagyobb teret hódít, mivel rendkívül rugalmasan alkalmazható a különféle rendszerekbe. Mi sem bizonyítja ezt jobban. Az Arduino egy flexibilis újraprogramozható mikrokontroller, melynek ki- és bemeneteit mi állíthatjuk be a programozás során, ezzel leegyszerűsítve a programozási folyamatot.

A programkódot saját fejlesztői környezetében, az úgynevezett Arduino IDE-ben tudjuk megírni. Ami Java keresztplatformos nyelvet alkalmaz. Az Arduino további előnye, platform független, aminek köszönhetően Windows, Linux, és Machintos operációs rendszereken is futtatható [7].

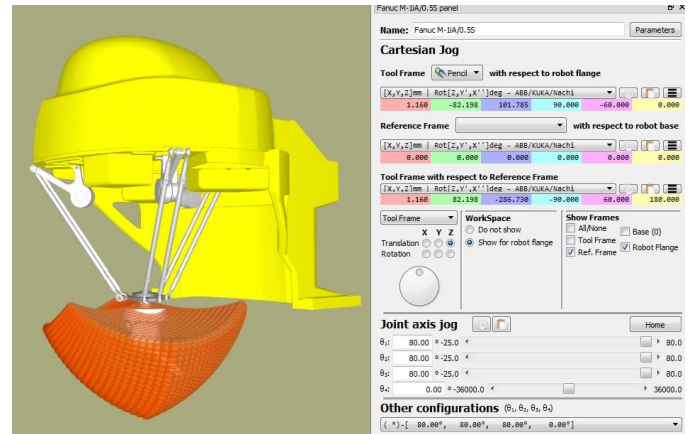


4. ábra Arduino Mega 2560 [8]

Projekt során egy Arduino Mega 2560 típusú board-ot használtam, amely 54 digitális I/O pinnel és 16 analóg bemenettel, valamint a többi Arduinohoz képest nagy tárhellyel rendelkezik [8]. Ez alkalmassá teszi olyan nagyobb szabású projektek vezérlésére is, ami például a 3D nyomtatással vagy robotikával kapcsolatos.

III. OKTATÁSI TESZTPANEL MEGTERVEZÉSE & KIVITELEZÉSE

A tesztpanel megtervezésével kiindulási pontként a FANUC ipari robot munkaterébe szolgált, valamint mozgásának ismerete. Mivel a robot kis helyen dolgozik, nekem is egy kompakt panelt kellett létrehoznom, amit be tud járni a robotkar és szemléltetni is lehet mozgásának pályáját.



5. ábra FANUC M11a 0.5A robot [4]

Az alap elgondolás egy nyomtatott áramkör volt, amelyen gombok helyezkednek el úgy, hogy a robot munkaterébe illeszkedjen és meg tudja nyomni őket. A gomboknak strapabíróknak kell lenniük tekintettel arra, hogy a FANUC ipari robot nagy erő kifejtésére alkalmas, és sokszor olyan hallgató fogja kezelni, aki még csak akkor tanulja a robot mozgását.

A gombokhoz állapotjelző LEDek tartoznak, mutató az éppen következő gombot, illetve a program állapotát. Ezeket a LEDeket zöld és piros színűre kerültek választásra.

A tesztpanelen a gombok helyzete a következőképpen néz ki: öt gomb van a panel felső részén úgy, hogy négy gomb a négy sarokban, egy pedig középen helyezkedik el. Ez a középső programváltó gombként funkcionálna, amíg a többi 4 gombot a robot nyomná meg a programnak megfelelően.

A gombok és a LEDek egy nyomtatott áramköri lapon helyezkednek el, így biztosítva az összeköttetésüket. A nyomtatott áramkör – röviden PCB – azért is bizonyult jó választásnak, mert nincs szükség drótozásra az elemek összekötésére, ezzel is csökkentve a kontakthiba lehetőségét, emellett nem foglal sok helyet és letisztult, esztétikus formát biztosít az áramkörnek.

A nyomtatott áramkört és az Arduino Mega 2560-at egy tartódobozba került elhelyezésre, amely stabilan tartja az alkatrészeket és védelmet biztosít a külső behatásokkal szemben. Továbbá a robottal való egy rendszerbe illesztésnél

rögzíteni lehet a tesztpanelen általa. A fent felsoroltak miatt egy masszív szerkezet kialakítására volt szükség. A tartódoboznak egy hegesztett acél keret szolgált.

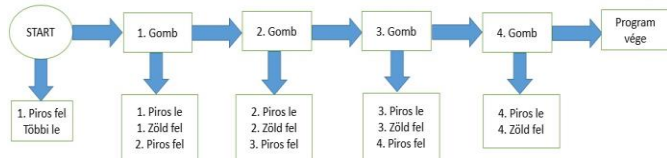
IV. ATMEGA2560 MIKROKONTROLLER PROGRAMOZÁSA & 3D TERVEZÉS

A konkretizálódás után, az Arduino programkódjának megírása következett. Programnyelvként a Java - C/C++ módosítását használja. Az ATMEGA2560 programkódja két fő részből, a „setup” és a „loop”-ból áll. Előbbiben állítjuk be a be és kimeneteket, utóbbiban futtatja le a program újra és újra ismétlődő végtelen ciklusát.

Alapvetően három üzemmód írására került sor, egy programkódon belül, így újabb feltöltés nélkül működtethető a FANUC robottal.

Az első program felkapcsolja nyomógombokhoz tartozó LEDeket, mint egy visszajelzés jelzésképpen, hogy a program futása elindult. Valamint ezáltal ellenőrizhető, hogy minden LED működik-e, illetve be van-e kötve.

Ezt követően a programváltó gomb megnyomásával ugrik a második programhoz az Arduino. Ebben a programban a gombok sorban való megnyomása történik. Először felkapcsolódik az egyes gombhoz tartozó piros LED. Miután megnyomtuk az egyes gombot, az egyes piros LED lekapcsolódik, felkapcsolódik az egyes zöld LED – jelezve a sikeres megnyomást – és felkapcsolódik a kettes gombhoz tartozó piros LED. Ezután a kettes gombot kell megnyomni és hasonló dolog történik, csak a megfelelő LEDekkel. Az összes gombot megnyomva a program végén az összes zöld LED világítani fog.



6. ábra: A 2. program folyamatábrája

Ez a program a pálya menti mozgást írja le, amikor a robotnak egy konkrét útvonalat kell bejárnia.

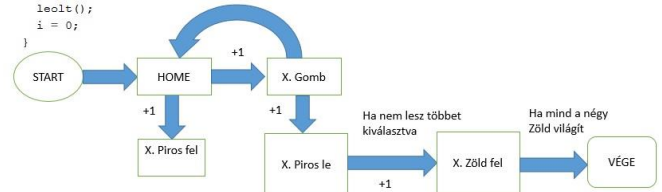
Ezután a programváltó gomb lenyomásával lép a program a harmadik programrészhez. Ebben a részben is felvillan egy piros LED és meg kell nyomni a hozzá tartozó gombot, ám itt már nem sorban történik a LEDek felvillanása. A program úgy lett írva, hogy miután a robot megnyomta a megfelelő gombot, visszatérjen HOME pozícióba, vagyis a középső programváltó gomb fölé és ezután kapcsolódik fel a következő piros LED. A LEDek sorrendje véletlenszerűre van megírva, és többször is szerepelhet ugyanazon programrészben belül. Ha egy gombhoz már nem fog visszatérni a program (és ezáltal a robot sem), akkor az ott lévő zöld LED fog felkapcsolódni és a program végéig felkapcsolva marad.

```

if (reading == HIGH && previous == LOW && millis() - time > debounce) {
    state++;
    val = true;
    time = millis();
}
previous = reading;

switch(state){
    case 1:
        if(val){
            hanyadik = 0;
            leolt();
            digitalWrite(x1, HIGH);
        }
        piros_ozd();
        val = false;
        break;
    case 2:
        if(val){
            leolt();
            i = 0;
        }
}

```

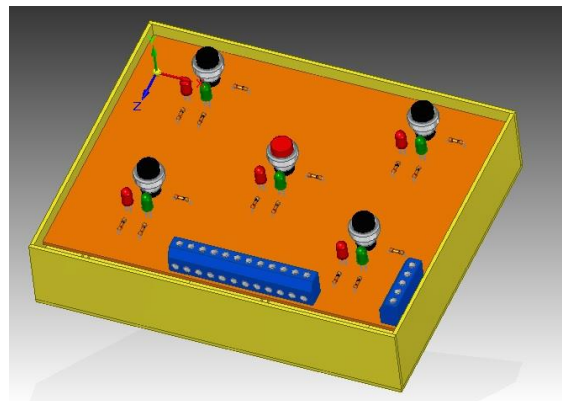


7. ábra: A 3. program kód részlete és a hozzá tartozó folyamatábra

Végül ennél a programnál is mind a négy gombhoz tartozó négy zöld LED világítani fog, ekkor vége van az utolsó programnak is, a robot megnyomja még egyszer a programváltó gombot és visszatér HOME pozícióba. A programváltó gomb nyomásával az Arduino visszalép az első programba, kezdődhet az egész futtatás előről.

A tesztpanel megépítése előtt 3D tervező programban került megtervezésre. A 3D rajz segítségével, pontosan megadhatjuk a méreteket, és ellenőrizhetjük, hogy a beültetendő alkatrészek az készíteni kívánt PCB –hez furataihoz illeszkednek e. 3D CAD program terén a Solid Edge 3D [10] tervező program volt alkalmazva a térbeli modell megalkotása során, mivel a Student Licence keretein belül ingyenesen volt a használata.

A 3D modellben összeállt a tartódoboz a PCB lap és a többi alkatrész helyzete.



8. ábra: A tesztpanel 3D modellje

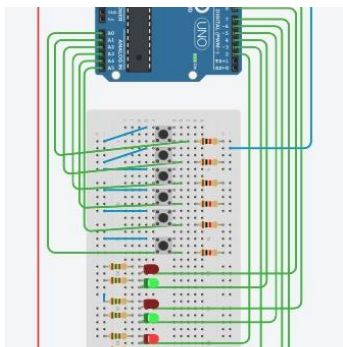
A 3D tervezés és rajzolás előnye még, hogy a modelltől a későbbiekben 3D nyomtató segítségével akár kinyomtathatjuk a szükséges alkatrészt.

V. TINKERCAD ATMEL SZIMULÁCIÓ & EAGLE

A tervezési fázishoz hozzátartozott az áramkör szimulációja. Ezt a TinkerCad [11] nevű böngésző alapú 3D tervező, modellező és áramkör építő eszköz segítségével lett megvalósítva.

Külön ATmel mikrokontrolleres projektet hozhatunk létre benne.

Egy próbapanelen összerakhatók a szükséges alkatrészek, majd vezetékekkel összeköthetők az Arduino pinjeivel. Ha elkészült az áramkörünk, nincs más dolgunk, mint feltölteni az áramkörhöz írt programkódot, és elindítani a szimulációt. A szimuláció közben eltelt időt egy számlálón láthatjuk, amelyen milliszekundumos pontossággal telik az idő.



9. ábra: Atmel mikrokontroller szimuláció

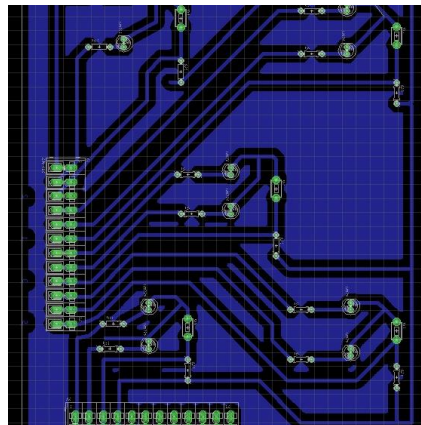
Amint megvolt a szimuláció, ellenőrzésre került a programkód, a PCB tervezés következett az EAGLE [12] nevű programban.

Az áramkör kapcsolási rajzát az EAGLE-ben új projekten belül kell létrehozni, amibe később a program a PCB nyomtatási tervét is menti. A kapcsolási rajznál egyesével ki kell választanunk az áramköri elemeket, és behúznunk a munkaablakba. Az alkatrészek az EAGLE könyvtáraiban található meg. Vannak olyan alkatrészek persze, amiket az EAGLE alapkönyvtára nem tartalmaz, de ezeket könnyen beszerezhetjük a hivatalos honlapon található könyvtárak közül.

Ha minden megfelelő a kapcsolási rajzunkban, következhet a PCB terv készítése, amit a meglévő kapcsolási rajzból generálja a program. Ilyenkor elvesznek az alkatrészek egymáshoz viszonyított helyzetei, csak a vezetékek maradnak meg az összekötött részek között. Ezeket az úgynevezett légkötéseket kell majd ábrázolnunk a végleges vezetősávok kialakításához, de előbb el kell helyeznünk a megfelelő pozícióba az alkatrészeinket. ha ez megvan automatikusan is generálhatunk vezetősáv rajzolatot, de ez általában nem a kívánt eredményt hozza. Biztosabb megoldás ha magunk rajzoljuk meg a vezetősávokat, mert később a vasalásos technikával készített nyomtatott áramkörrel így hatékonyabban tudunk dolgozni, mind a maratás, mind a fúrás és forrasztás során.

Ha összekötöttünk minden alkatrészt és készen van az áramkör, még rakhathunk telefeldet az üresen maradt részekre esztétikai szempontból, de ez opcionális.

Ha mindent rendben találunk, kezdődhet a PCB terv kinyomtatása. Arra ügyeljünk, hogy előtte minden réteget vegyünk le a a nyomtatási tervről, amelyet nem szeretnénk rávasalni a PCB lapra (alkatrészek körvonalai, feliratok stb.)

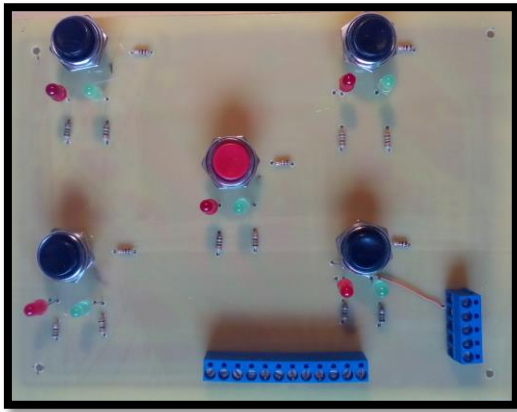


10. ábra: NyÁK terv

A PCB elkészítése vasalásos technikával került elkészítésre.

Szükség volt először is a kinyomtatott PCB tervre. Ezt lézernyomatóval fényes műnyomó papírra ajánlatos nyomtatni, hogy vasalás során át tudjon ragadni a tinta a rézfelületre. Szükség volt hozzá még egy egyoldalas nyers PCB lapra, ami egy üvegszállal erősített epoxigyanta lap, az egyik oldalán 35 µm vastag rézréteggel bevonva. Erre a rézes oldalára kellett rávasalni a kinyomtatott PCB tervet vasaló segítségével. Ha mindenhol rásimítottuk a PCB lapra a forró műnyomót, le kellett áztatni a papírt, hogy csak a tinta maradjon meg a réz felületen. Ha ez megvolt, következett a maratás gyorsmarató folyadékkal. Ez lemarta a fölösleges rézréteget, de meghagyta a vezetősíneket a tinta alatt. A maratás végeztével le kellett mosni a tintát hígítóval, és a jobb tapadás és az oxidáció elkerülése végett kémiai ónozóval vékony ónréteget vittem fel a megmaradt rézre.

Furatszerelt PCB lap elkészítve, mert így könnyebb beforrasztani vagy cserélni az alkatrészeket. A furatokba beforrasztás követően elkészült a PCB panel.

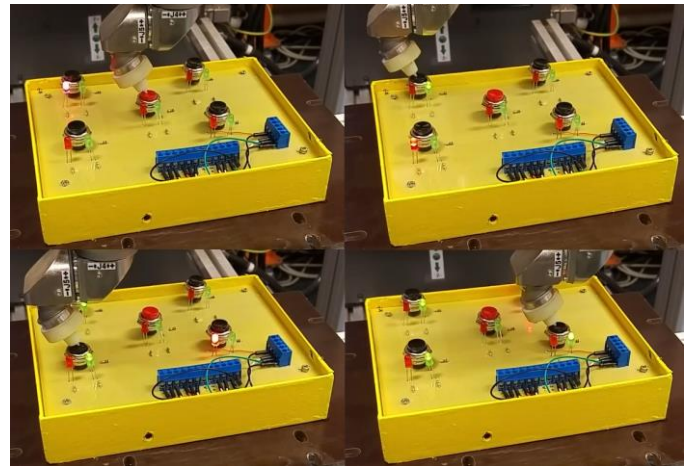


11. ábra: PCB Tesztpanel

Ezután a tartódoboz hiányzott a kész tesztpanelhez. Ezt 2 mm-es lemezből kivágott alkatrészekből lett összehegesztve, bevont elektródás ívhegesztéssel. Miután összeállt a szerkezet, felületkezelést kapott két réteg zománcfesték formájában.

VI. A TESZTPANEL & FANUC M11A 0.5A ALKALMAZÁSA

Az elkészült tesztpanel előzetes tesztjei után sor került a Debreceni Egyetem, Mechatronika Tanszék, robotlaborjában FANUC M11A 0.5A robottal való éles tesztekre is. Az első teendő a tesztpanel pozicionálása volt. Ehhez az eredeti, egy munkadarab rögzítésére szolgáló tartószerkezetet át kellett alakítani, mert nem volt kompatibilis a pannellel, túl alacsonynak bizonyult. A magasság növelése érdekében a már meglévő menetes száakra szabványos, 40x40 mm-es alumínium profilokat erősítettem, így sikerült elérnem a panel és a FANUC robot együttműködéséhez szükséges magasságot. A sík felület biztosítása érdekében az alumínium gépépítő profilokra egy műanyag lap lett felcsavarozva, és a tesztpanel helyzetének stabilizálása érdekében a tesztpanelt szintén csavar rögzítéssel a műanyag laphoz fogattam fel. A robot mozgásával bemérésre került, hogy milyen munkaterületet kell fednie a tesztpanelnek, a magasságát pedig a menetes száron lévő csavarok állításával lehetett pozicionálni. A panel pozicionálása után következett a robot programozása. A mozgás betanítása így a FANUC iPendant kézi kontrollerevel történt. A kontrollerevel kézi vezérléssel jutott el a robot egyik pontból a másikba, és a működés szempontjából fontos pontok rögzítésre kerültek. Ezekből a rögzített pontokból állt össze a program. A teszt során a már korábban bemutatott három program futott le. Az első program feladata az Arduino működésének jelzése volt, továbbá jelezte, hogy a robot elindulhat.



12. ábra: A tesztpanel és a FANUC robot együttműködése.

A robot HOME pozícióból indulva megnyomta a programváltó gombot, majd sorban a négy további gombot. Ezután ismét a programváltó gombot nyomta meg, majd HOME pozícióból indulva és oda visszatérve mindig azt a gombot nyomta meg, amelyik mellett felvillant a piros LED mindaddig, amíg mindegyik gomb mellett nem világított a zöld LED. Mikor minden zöld LED világított, az jelezte a program végét. A harmadik program után a robot ismét megnyomta a programváltó gombot, ezzel az Arduino alapállapotba állt (1. program, LED-ek égnek). A robot visszatért a HOME pozícióba, és ezzel a robot programja is befejeződött. Biztonsági okokból a teszt során a robot teljes sebességének 30%-án futott, így jól látható volt a robot mozgása, és a program futása az Arduino vezérelte tesztpanelen.

VII. ÖSZEGZÉS

A kitűzött feladatot teljesítésre került. Az elkészített PCB teszt panel megtervezésre és megépítésre került. A tesztpanel, a FANUC M11A 0.5A ipari robottal alkalmasnak bizonyult az együttműködésre. Oktatási célra alkalmas, mivel flexibilis Atmel mikrokontroller-re épült, így könnyen bővíthető/fejleszhető a jövőben.

VIII. KÖSZÖNETNYÍLVÁNÍTÁS

A publikáció elkészítését az EFOP-3.6.1-16-2016-00022 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg. Szeretnék köszönetet mondani Erdei Timotei Istvánnak, hogy elvállalta a konzulensem szerepét, a szükséges információkat biztosította számomra, és ha kérdésem támadt, mindig bizalommal fordulhattam hozzá, ő rendelkezésemre állt. Továbbá szeretnék köszönetet mondani Korsoveczki Gyulának, aki a tanszéken található FANUC M11A 0.5A robot működésének megismerésében támogatott, arról információkat biztosított nekem és segített a robot programjának megírásában is.

IX. HIVATKOZÁSOK

- [1] Jaroslav Sobota, Roman Pišl, Pavel Balda, Miloš Schlegel, (2016, Június 2). "Raspberry Pi and Arduino boards in control education", Sheffield, UK 2013 [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1474667015340684>
- [2] „FANUC M1iA 0.5A” FANUC Hungary Kft. (2016, Június 2). [Online]. Available: <https://www.fanuc.eu/hu/hu/robotok/robotsz%C5%B1r%C5%91-lap/m1-sorozat/m-1ia-05a>
- [3] „A FANUC M1iA 0.5A robot” [Online] (2016, Június 10).. Avalable: https://www.fanuc.co.jp/en/product/robot/f_r_genkotu.html
- [4] „Arduino Mega 2560” [Online]. (2016, Június 5).Available: https://store-cdn.arduino.cc/usa/catalog/product/cache/1/image/500x375/f8876a31b63532bbba4e781c30024a0a/a/0/a000067_iso_1.jpg
- [5] Hardware in the loop [Online]. (2016, Június 12). Available: <http://www.j2aircraft.com/application-benefits/simulated-environment-and-training/>
- [6] SLA 3D Printer [Online]. Available: <https://all3dp.com/1/best-resin-dlp-sla-3d-printer-kit-stereolithography/>
- [7] Arduino IDE [Online]. (2016, Június 8).Available: <https://www.arduino.cc/en/main/software>
- [8] Arduino MEGA2560 [Online]. (2016, Június 6).Available: <http://qqtrading.com.my/image/catalog/Products/Arduino/mega/arduino-mega-pinout-diagram.jpg>
- [9] RoboDk [Online]. (2016, Június 7).Available: <https://robodk.com/>
- [10] SolidEdge [Online]. Available: <http://graphit.hu/solidedge/>
- [11] TinkerCAD [Online]. Available: <https://www.tinkercad.com/>
- [12] EAGEL [Online]. (2016, Június 7).Available: <https://www.autodesk.com/products/eagle/overview/>
- [13] T. I. Erdei, Zs. Molnár, N. C. Obinna, G. Husi, „AGV cyber physical navigation system,” FMTÜ - XXII. International Scientific Conference of Young Engineers - Kolozsvár, 23-24.03.2017.
- [14] N. C. Obinna, T. I. Erdei, Zs. Molnár, G. Husi, „LabVIEW Motion Planning and Tracking of an Industrial Robotic Manipulator (KUKA KR5 arc): Design, Modelling, and Simulating the Robot’s Controller Unit,” FMTÜ - XXII. International Scientific Conference of Young Engineers - Kolozsvár, 23-24.03.2017.
- [15] Zs. Molnár, T. I. Erdei, A. Husam, G. Husi, „Self-developed CNC prototype as IoT device,” FMTÜ - XXII. International Scientific Conference of Young Engineers - Kolozsvár, 23-24.03.2017.