

Implementing and testing “Aubot” robot using self-study and collaborative learning strategies

R. Toth ^a, R. Auer ^b

^a Roland Tóth, Robert Bosch Kft., Budapest 1103, Hungary, halesz89@gmail.com

^b Richárd Auer, T-Systems Hungary Ltd, Budapest 1097, Hungary, auer.richard@t-systems.hu

Abstract

The presented robot development is based on a well-defined engineering task which mainly increases the efficiency of critical thinking, problem-solving, self-study strategy and teamwork and it provides the project-based education and problem-based learning. This article introduces the design, construction and programming of a line-tracking robot, which called the "Aubot", and it provides a comprehensive introduction to the basics of robotics, such as PWM and engine control. The article will guide you through the entire construction, introducing the different technologies and methods. During the task solution we met so problems which contributes our own motivation to learn autonomously and collaboratively.

Keywords: project-based learning (PBL); problem-solving skills; autonom robot; Arduino; PWM

Az “Aubot” robot megvalósítása és tesztelése önálló és kollaboratív tanulási stratégiákat alkalmazva

Tóth R. ^a, Auer R. ^b

^a Tóth Roland, Robert Bosch Kft., Budapest 1103, Magyarország, halesz89@gmail.com

^b Auer Richárd, T-Systems Magyarország Ltd, Budapest 1097, Magyarország, auer.richard@t-systems.hu

Absztrakt

A fejlesztés egy előre meghatározott mérnöki feladatra épít, amely leginkább a kritikus gondolkodás, a problémamegoldó képesség, az önirányított tanulási stratégia és a csoportmunka képesség hatékonyságát növeli, lehetővé téve a projekt alapú oktatást és probléma alapú tanulást. A cikk egy „Aubot” fantázia nevű, vonalkövető robot tervezését, megépítését és programozását mutatja be, illetve egy átfogó bevezetést nyújt a robotika alapjaiba, mint például a PWM szabályozása vagy a meghajtásért felelős motor vezérlésébe. A cikk végig vezet az építés egészén, bemutatja a különböző technológiákat és módszereket. A feladat megoldása során olyan problémákkal találkoztunk, amelyek önálló vagy kollaboratív tanulási késztetést ébresztettek bennük.

Kulcsszavak: projekt alapú oktatás; problémamegoldó képesség; autonóm robot; Arduino; PWM

1. Bevezető

A motivációnk alapját Katona, J., & Kovari, A. (2016) cikke adta, ahol nagy hangsúlyt kap a projekt alapú oktatás (Moursund, 1999), ahol a diákok feladata egy agy-számítógép interfész rendszer kifejlesztése és hatékonyságának tesztelése volt. Az ilyen jellegű projekt feladatok

alatt jól fejlődhet a problémamegoldó készségünk és életközeli összefüggéseket tanulhatunk (Chu és Hwang, 2010).

A bemutatott fejlesztés egy hallgatói projektmunka eredménye, melyben a hallgatók egy egyedi kialakítású, költséghatékony robotot dolgoztak ki és valósítottak meg. A feladat megvalósítása során mind az önálló ismeretelsajátítás, mind pedig a kollaboratív technikák fontos szerepet kaptak (Orvis, 2008). A téma elméleti háttérének összefoglalása, a felhasznált eszközök kiválasztása, a szoftver megtervezése, megvalósítása a hallgatók közös munkájaként valósult meg.

2. Téma elméleti háttére

Robotok. Az a szó, amelynek hallatán sok embernek a filmekben látott emberek életére törő robotok jutnak eszébe. A valóság valójában nagyon is távol áll ettől még, de elmondható, hogy a robotok fejlődése a XXI. században egyre jobban gyorsul. Az élet számos területén találkozhatunk velük, legyen szó akár a hadi iparról, vagy a gyártósorokon dolgozó forrasztó robotokról, illetve nem szabad elfeledkeznünk az égitesteket kutató robotokról sem. A robotokban alkalmazott beavatkozók lehetnek egyszerű egyenáramú, vagy korszerű villamos hajtások Kovari, A. (2009), de alkalmazhatnak pneumatikus, vagy nagy erő/nyomaték kifejtés esetén szervó hidraulikus beavatkozókat is. Kovari, A. (2015), Kovari, A. (2009) A téma komplexitása és a gondolat, hogy egy bizonyos szinten autonóm robotot hozzunk létre elég indíték volt a project elkezdéséhez. A robotika több területet ötvöz hardveres és szoftveres szinten, vagyis elektromechanikai szerkezetről is beszélhetünk.

Természetesen közben adódtak problémák melyeket ugyan úgy be fogunk mutatni, mint a probléma megoldásához vezető utat is. A projectben két vonalkövető robot készült el. Ahogy korábban említésre került, robotokat a XXI. században nagyon sok területen használják, amely tárgyalása az alábbi fejezetekben található annak érdekében, hogy az olvasó jobban megértse a cikk végén tárgyalt továbbfejlesztési, felhasználási területek miértjét. Moubarak, P. M., & Ben-Tzvi, P. (2011) A robotokat két nagy csoportra oszthatjuk:

- katonai alkalmazású robotok
- polgári alkalmazású robotok

2.1. *Katonai alkalmazású robotok*

A katonai alkalmazású szárazföldi robotokkal többnyire olyan - az ember számára veszélyes - helyeken találkozhatunk, mint a sugárzással vagy harci gázzal szennyezett területek. Továbbá másik alkalmazási területe a bombák hatástalanítása és ezzel emberéletek megmentése. Szárazföldi robotokat legnagyobb számban a légierőnél és a haditengerészetnél alkalmaznak az úgynevezett drone-ok révén. Az ilyen célú felhasználásukat sokan támadják, ugyanis ezeket a harci gépeket az emberek távolról irányítják, amely sok esetben etikai problémákba ütközik. A robotok egy másik még fejlődő ága a mállásrobotok, vagyis szállítórobotok, amik a katonák menetfelszerelését hivatottak szállítani. Ez különböző területeken jól manőverező robotokat kíván. A Boston Dynamics jóvoltából már most is akad néhány ígéretes próbálkozás, amelyek tesztelés céljából a katonasághoz is kikerültek. Pár említésre méltó robot tőlük: BigDog, Cheetah, LittleDog, RiSE, Atlas. Moubarak, P. M., & Ben-Tzvi, P. (2011), Moubarak, P., & Ben-Tzvi, P. (2012), Gutkind, L. (2006). Az 1. ábrán látható a BigDog névre keresztelt szállító robot.



1. ábra: Big Dog névre keresztelt robot

Forrás: https://en.wikipedia.org/wiki/Boston_Dynamics

2.2. *Polgári alkalmazású robotok*

Még több a kérdés és a vita a polgári alkalmazású robotokkal kapcsolatban, ugyanis léteznek már olyan drone-ok, amelyek mindenki számára elérhetőek és azok hétköznapi felhasználása egyes esetekben jogsértő lehet. Elég csak arra gondolni, hogy egy kamerával felszerelt drone

egy kisebb kertvárosban gyakorlatilag bárhová benézhet, amivel magánszférát sérthet. Ezért minden valószínűség szerint a közeljövőben az ilyen esetekből fakadó jogi procedúrák könnyítése okán új, kifejezetten drone-ok felhasználására vonatkozó törvények láthatnak napvilágot. Amerikában az egyik legnagyobb szállítványozó cég bejelentette, hogy a közeljövőben a kiszállításoknál szóba kerülhet a drone általi kézbesítés, de technikai problémákon kívül még jogi problémák is fennállnak. Európában a Svájci Posta már elkezdte tesztelni a robotpostát. Persze vannak olyan területek, amelyeknél nem állnak fenn hasonló jogi kérdések, például a szobában alkalmazható robot porszívók vagy az Amerikában használatos őrző védő robotok. Moubarak, P. M., & Ben-Tzvi, P. (2011) A polgári robotok további csoportját alkotják a mobil kialakításúak, amelyek számos kutatások alapját is képezhetik, kedvező áruk és relatíve könnyű beszerezhetőségük miatt. Katona, J., Ujbanyi, T., Sziladi, G., & Kovari, A. (2016), Katona, J., Kovari, A., Ujbanyi, T., & Sziladi, G. (2017)

3. Célkitűzések érzékelési módok

A project célkitűzése egy olyan robot megépítése, amelynek érzékelni és követni kell tudni egy, a padlóra elhelyezett fekete vonalat. Első lépésként azt kell megvalósítani, hogy a robot érzékelje a vonalat. A cél eléréséhez több módszert alkalmazhatunk, amelyek közül néhányat az 1. táblázat mutat be.

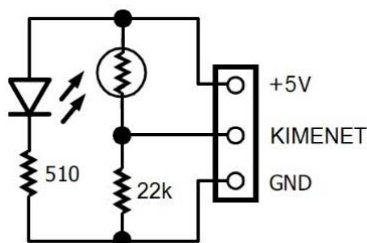
A project célkitűzése egy olyan robot megépítése, amelynek érzékelni és követni kell tudni egy, a padlóra elhelyezett fekete vonalat. Első lépésként azt kell megvalósítani, hogy a robot érzékelje a vonalat. A cél eléréséhez több módszert alkalmazhatunk, amelyek közül néhányat az 1. táblázat mutat be.

1. táblázat: Vonalérzékelések módjai

Vonal érzékelésnek a módjai		
<i>Megvilágítás</i>	<i>Érzékelő típusa</i>	<i>Nehézség</i>
Led	Fotoellenállás (LDR)	könnyű
Led	Fototranzisztor	könnyű
IR Led	Fototranzisztor	könnyű
természetes fény	Kamera CCD vagy CMOS	nehéz

Az első három megoldás esetén az érzékelőnek Led-es megvilágításra van szüksége. Kamera alkalmazása esetén elegendő a természetes fény. Az általunk választott, fotoellenállást alkalmazó megoldás elsősorban annak alacsony költsége miatt került kiválasztásra. A

fotoellenállás egy kadmium szulfid (CdS) ellenálláscsik, amelynek az ellenállása a ráeső fény intenzitásától függ. Sötétben az ellenállása nagy, világosban pedig alacsony. Ezt a változást lehet kihasználni, ha egy feszültség osztó megoldással használjuk. A feszültség osztó ellenállást a 2. ábra illusztrálja.

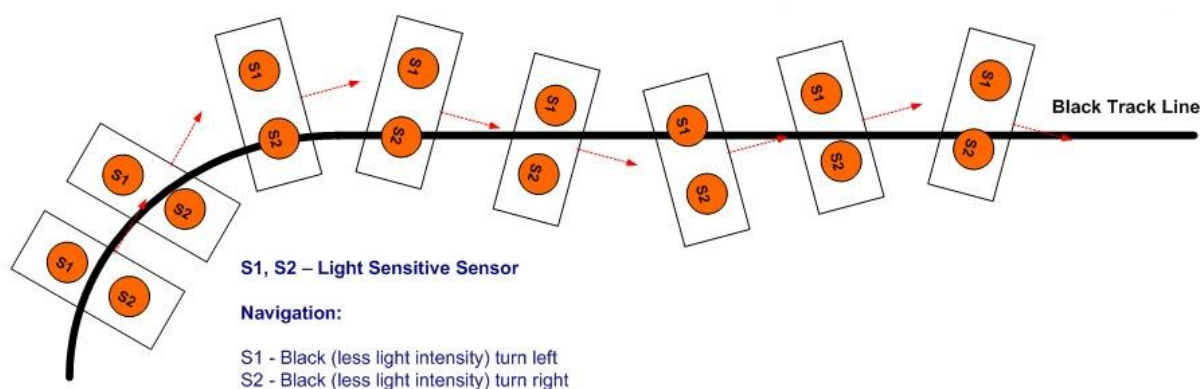


2. ábra: Feszültségosztó kapcsolás

Az alkalmazandó ellenállás értékeket a következő képlettel számolhatók:

$$resistor = \sqrt{R_{dark} * R_{bright}} \quad (1)$$

A fejlesztés során szükségessé vált a fotoellenállás értékének mérése sötét és világos fényviszonyok között. A fenti képlettel kiszámolható az alkalmazandó ellenállás értéke, amelyből kettőt egymással párhuzamosan elhelyezve válik lehetővé a vonal érzékelése a robot számára. A navigálás menete a 3. ábrán látható.



3. ábra: Navigálás a vonal mentén

Forrás: <http://beambuilder.blogspot.hu/2015/02/lm324-quad-op-amp-line-follower-robot.html>

Ahogy az ábrán is látható, a navigálás szenzorok segítségével történik. Amikor a szenzorok a padlóról visszaverődő fény alapján magas kimeneti jelet generálnak (vagyis a vonal a két szenzor közé esik), a robot előre megy. A navigáció során ezt a jelet vizsgáljuk. Amikor a

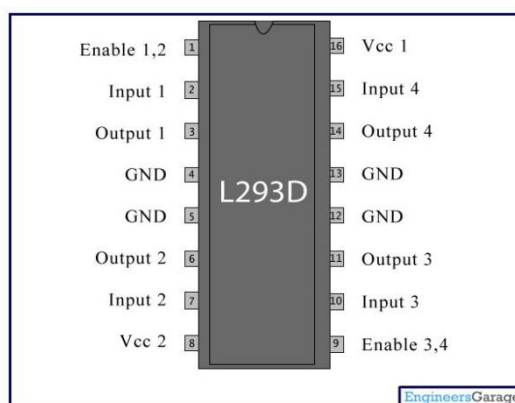
szenzor eltérne és kezdene ráfutni a vonalra, a robot a helyes vonalkövetés érdekében korrigálja magát. A 2. táblázat mutatja a lehetséges állapotokat.

2. táblázat: Vonalérzékelések módjai

Bal szenzor	Jobb szenzor	Motor
Padló felett	Padló felett	Bal és jobb motor előre
Vonal felett	Padló felett	Bal motor állj, jobb előre
Padló felett	Vonal felett	Bal motor előre, jobb motor állj
Vonal felett	Vonal felett	Kereszteződés Ball és jobb motor előre

4. Motorvezérlés

A motor vezérlése egy L293D dupla H-hidas motorvezérlővel került megvalósításra, amelynek előnye, hogy egyszerű és nem utolsó sorban az olcsó megoldások közé sorolható. Továbbá felhasználásra került egy CD 4069 inverter. Az L293-as vezérlő rajza a 4. ábrán látható.

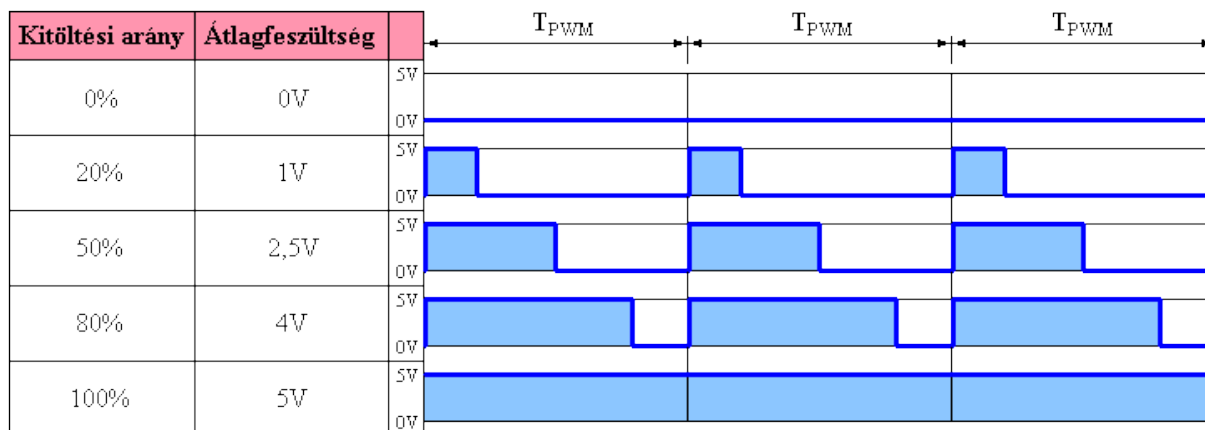


4. ábra: L293 motorvezérlő láb kiosztása

Forrás: <http://www.engineersgarage.com/>

A motorvezérlőben a VCC 2 az elemtartó GND-re került bekötésre, addig a VCC 1 a működéshez szükséges 5V-ot adja. Az Input 1-4 az inverter-től kapja a jelet. A motor hajtása (előre vagy hátra) a kapott jel függvényében változik. A motor sebességét PWM jelekkel szabályozzuk, ami az Enable ágakon került rákötésre.

Olyan állandó periódusjelű (és frekvenciájú) jelekről van szó, ahol az átlagfeszültség beállítása a jel tényezőjének változtatásával történik meg. Olyasmit képzeljünk el, mintha nagyon gyorsan ki/bekapcsolnánk a motort és a ki/be kapcsolási idő egymáshoz viszonyított aránya határozná meg az átlag feszültséget. Az 5. ábra ezt szemlélteti.



5. ábra: PWM jelek

Forrás: <http://www.t-es-t.hu/elopek/pic/felhkk/kk/sz1604.htm>

A fenti ábrán az X tengely jelöli az időt, az Y tengely a feszültséget, a T_{PWM} -et az egy teljes periódusnak felel meg és a periódushoz viszonyított aránya a kitöltési tényezője adja meg az átlagfeszültséget. A lenti képletben a 255-ös szorzó a 8 bit okán került bevezetésre, amit a mikro vezérlő kiküld.

$$KívántPWM = \frac{V_{eff}}{V} \times 255 \quad (2)$$

Például, ha az elérni kívánt feszültség 2.5 V, akkor:

$$KívántPWM = \frac{2.5}{5} \times 255 = 127.5 \quad (3)$$

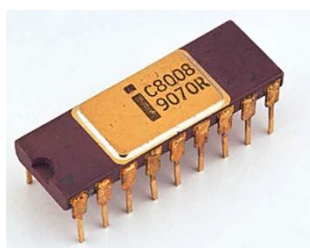
5. Arduino bemutatása hardveres oldalról

1969-ben a Busicom nevű vállalat (amelynek jogelődje 1945 és 1967 között a Nippon Machine Corporation volt az 1967-es névváltásig) azzal a céllal kereste fel és bízta meg Intel Corporation-t (aki ekkor már elismert és keresett gyártónak számított a memóriachipek gyártása terén), hogy segédkezzen négy féle áramkör legyártásában. A négyféle áramkörből az egyszerűség kedvéért végül egy komplex áramkör lett, amely különböző folyamatok sorrendjét tartalmazta. A leadott megrendelés teljesítése után az Intel megvette a jogokat a

Basicomtól, így az Intel tulajdonába került egy olyan technológia, amely Banzi, M., & Shiloh, M. (2014):

- egy chiprészletet tartalmaz, ahol folyamatok hajthatók végre,
- egy másik chip/chiprészlet, ahol ezen folyamatok sorrendisége letárolható,
- ennek a letároló résznek a tartalma megváltoztatható, felhasználó által elkészíthető.

A szabadlom birtokában az Intel 1971. november 15-én piacra dobta I4004-es processzorát, mely processzoronként 2300 db tranzisztort tartalmazott. A gyors fejlesztésnek köszönhetően 1972-ben létrejött a 8 bites 8008-as jelzésű, processzoronként 3300 db tranzisztort tartalmazó processzor, amely a 6. ábrán látható.



6. ábra: Intel 8008 mikroprocesszor

Forrás: <http://www.tomshardware.com/news/intel-8008-cpu-processor-anniversary,15176.html>

Az Arduino-nak számos változata létezik, amelyekről oldalakon keresztül lehetne írni. Mi most ezt nem tesszük meg. A project keretein belül az Arduino UNO-t, illetve annak kisebb testvérét, az Arduino NANO-t vesszük górcső alá. A választást az indokolja, hogy az említett mikrokontrollerekkel készültek el a robotok. A két mikrokontroller azonos tudásszintet képvisel, azonban néhány apróbb eltérés felfedezhető köztük.



7. ábra: Arduino UNO

Forrás: <http://www.arduino.cc/en/Main/ArduinoBoardUno>

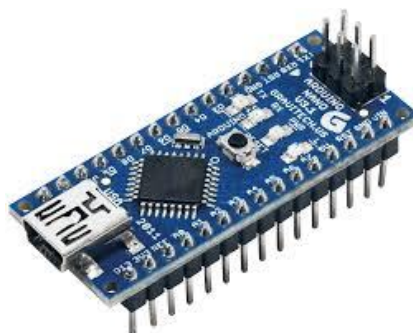
Az UNO esetében kétféle mikrokontrollerről beszélhetünk. Az egyik sima DIP IC-vel szerelt változat, addig a másik egy SMD kiadás. Mindkettő ugyanazt az ATmega 328-as mikrokontrollert kapta, amely rendelkezik 14 ki- és bemeneti lábbal. Ebből a 14 lábból 6-ot használhatunk PWM jelfeldolgozó lábként is (ez utóbbi fontos a fejlesztendő robot szempontjából), illetve 6 analóg láb is helyet kapott az alaplapon. Ezeken kívül kapott egy 16MHz-es kristály oszcillátort, egy USB csatlakozót, illetve egy táp csatlakoztatási lehetőséget.

Amiben lényegesen eltér az UNO a többi társától, az az, hogy nem használ FTDI USB driver-t, amelyet telepíteni kellene. Helyette kapott egy programozott USB-Serial átalakító chipet, amely az ATmega16U2 nevet viseli.

Néhány technikai adat:

- Mikrokontroller: ATmega328
- Működési feszültség: 5V
- Szükséges bemeneti feszültség: 7-12V
- Bemeneti feszültség limit: 6-20V
- Digitális I/O lábak száma: 14 (ebből 6 pwm-ként került felhasználásra)
- Analóg lábak száma: 6
- Flash memória: 32Kb (ebből 0.5Kb-et foglal a bootloader)
- ESRAM: 2Kb
- EEPROM: 1Kb
- Órajel sebesség: 16MHz

Mind a 14 digitális lába kihasználható, illetve bemeneti lábként, melyek programozáshoz a fejlesztői környezet függvényekkel szolgál. Ilyenek például a `pinMode()`; `digitalWrite()`; `digitalRead()`; függvények. Használatukkal - kis utánajárást követően - akár egyszerű villogtató program is írható. Természetesen mind a 14 láb 5V-ot használ, illetve képesek 40 mA befogadására és leadására. Valamennyi láb beépített ellenállás védelemmel rendelkezik 20-50 kOhm-ig.



8. ábra: Arduino NANO

Forrás: <http://makerclub.org/component/arduino-nano/>

Az Arduino NANO egy kisebb kiadása az UNO-nak. A NANO két chippel szerelt (ATmega168, ATmega328), amelyek eltérő verziószám alatt futnak. A NANO esetében a lapkán nincs tápcsatlakozó, kizárólag mini USB-vel működik.

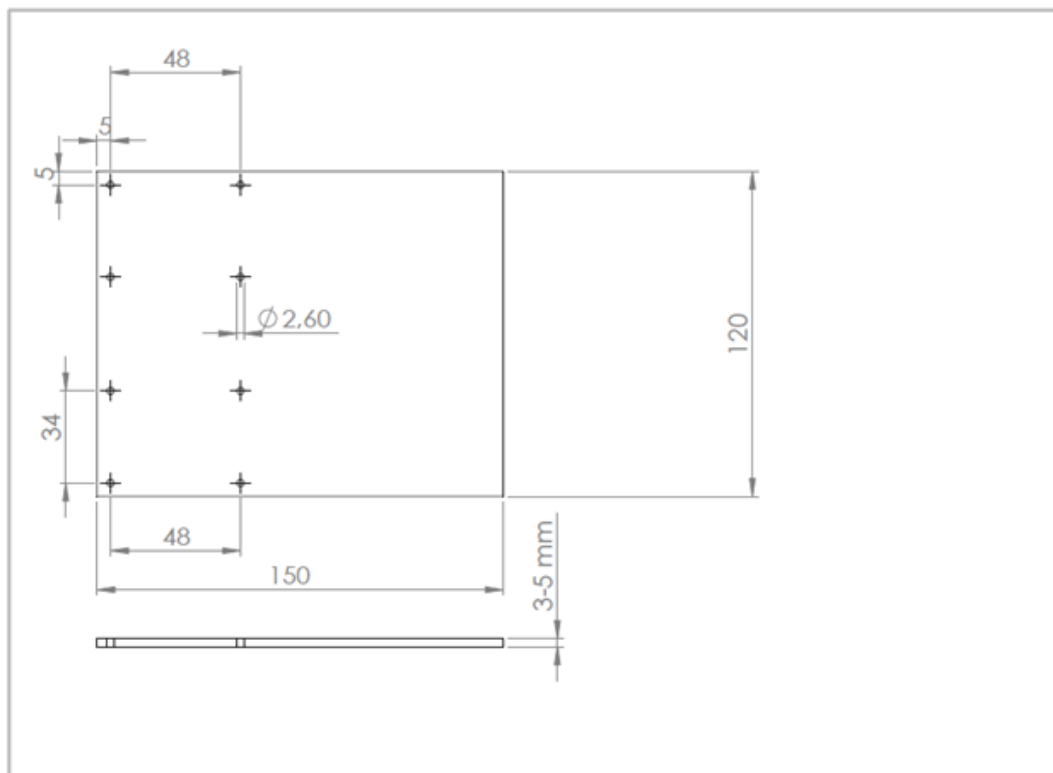
Néhány technikai adat:

- Mikrokontroller: ATmega328 vagy ATmega168
- Működési feszültség: 5V
- Szükséges bemeneti feszültség: 7-12V
- Bemeneti feszültség limit: 6-20V
- Digitális I/O lábak száma: 14 (ebből 6 PWM-ként került felhasználásra)
- Analóg lábak száma: 8
- Flash memória: ATmega168 esetén 16Kb, ATmega328 esetén 32Kb melyekből 2Kb a bootloader
- SRAM: 1Kb, ATmega168-as lapkánál, amíg ATmega328 esetén ez 2Kb
- EEPROM: 512byte az ATmega168-nál ez ATmega328-nál 1Kb
- Órajel sebesség: 16MHz

Ahogy a technikai adatokból is látszik, az ATmega168-as chippel szerelt lapka szerényebb tudású, azonban programozhatóság szempontjából elhanyagolható a különbség. Az UNO jumper foglalattal rendelkezik, ami annyit jelent, hogy nem kell a rákötni kívánt szenzorjainkat a próbapanelen összedugdosni, hanem elegendő egy ónozott végű vezeték és a hőmérő már is betehető a foglalatba. Ellentétben a NANO-val, ahol a tűskesoros csatlakozásnak érdemes egy próbanyákra foglalatot készíteni, amelyen szépen kivezethetőek a lábak.

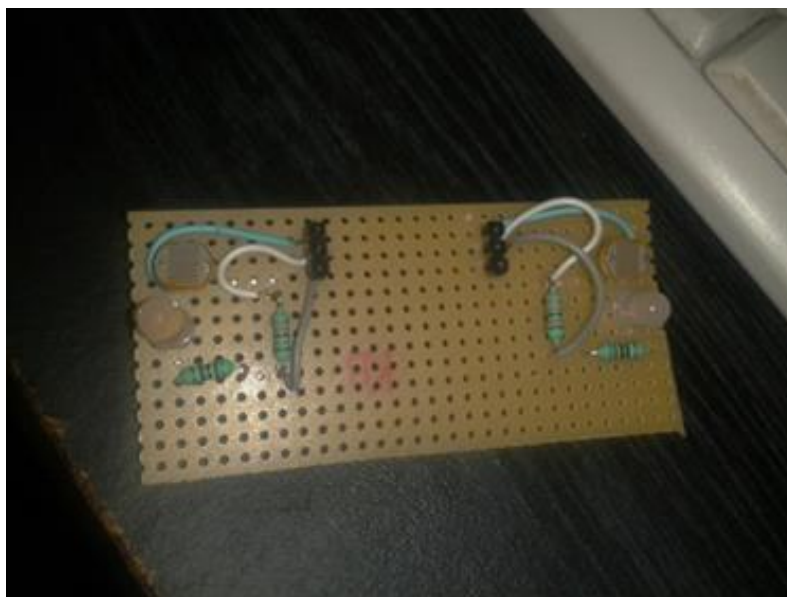
6. A robot megépítése

Váz: Ebben a részben készül el a robot alváza, itt kerülnek fel a motorok. Ez egy egyszerű plexi alapra került ráépítésre a módosíthatóság érdekében. A furatpozíciók megrajzolása TopSolid program segítségével történt, amelyet a 9. ábra szemléltet.

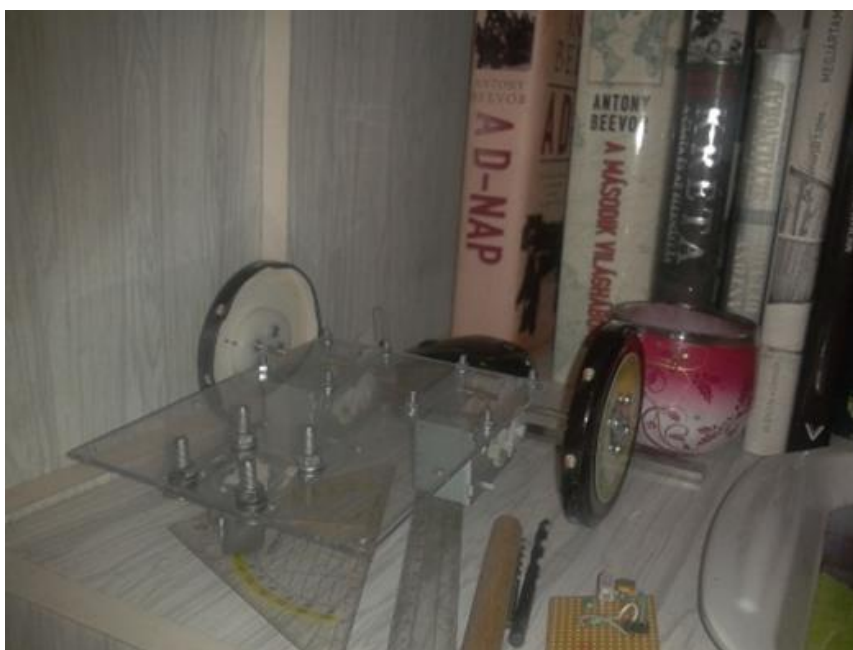


9. ábra: Robot plexi terv

Első lépésként a vázszerkezet készült el. A hátsó bolygókerék egy – a boltban is kapható – bútorgörgő segítségével mozog. A két robot komponenseinek elrendezése eltérő. A kész alvázat és a motorvezérlő panelt a 10. ábra szemlélteti. Az egyik motor páros külföldről került megrendelésre, gumírozott kerékkel ellátva. A másik robot esetében csak a motor került megrendelésre, így ahhoz a kereket is meg kellett csinálni. Ezt a robotot mutatja a 10. ábra. Ezt követően a vonalérzékelő szenzorokat készítettük el, amely a 11. ábrán látható.



10. ábra: A kész robot alváza



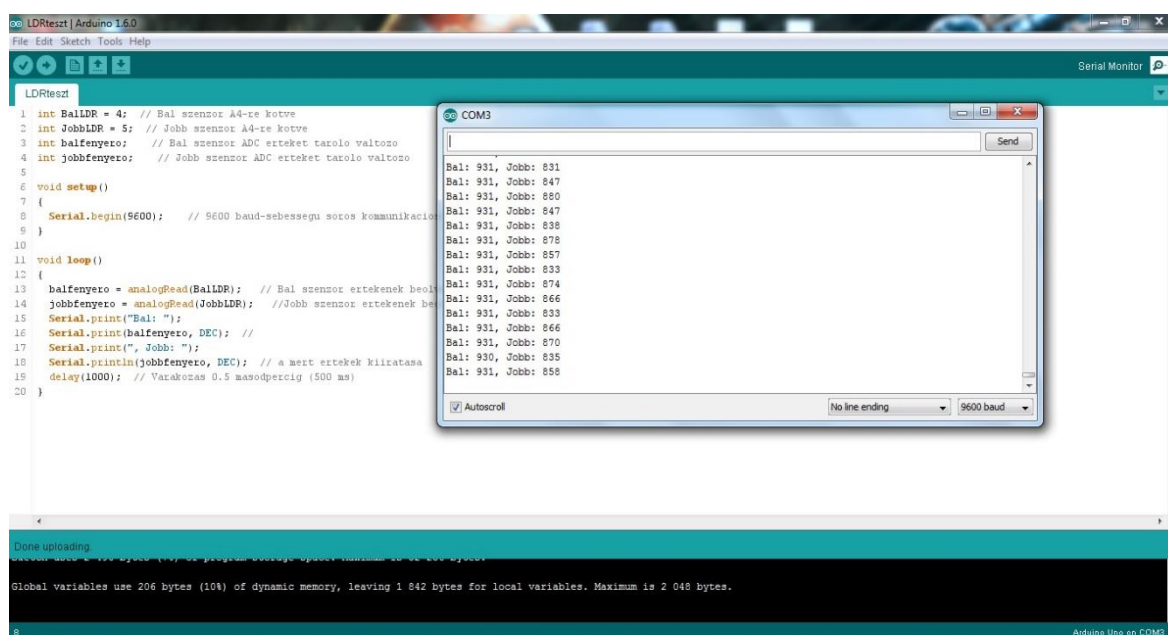
11. ábra: A vonalérzékelő szenzorokkal felszerelt egység

7. Programozás

Első lépésben a vonalérzékelő szenzorokat kell kalibrálni, hogy meghatározzuk azokat a konstansokat, amelyek a robotvezérlő program számára szükségesek. A robot pontos irányítása érdekében azt is meg kell vizsgálni, hogy a motorvezérlő parancsokra hogyan reagál a robot.

12.1. Vonalérzékelő szenzor tesztelése

Mielőtt a vonalérzékelő szenzorokat használni tudnánk, elemezni kell az általuk szolgáltatott jeleket, hogy el tudjuk dönteni mikor, van egy adott szenzor a vonal felett és mikor a padló felett. Egy tesztprogram segítségével, amely a legújabb Microsoft Windows operációs rendszeren a megfelelő driver telepítése után tökéletesen fut, megtörtént a szenzorok értékeinek vizsgálata, a vizsgált értékek pedig egy LCD kijelzőn kerültek megjelenítésre. A tesztprogram analóg lábon figyeli 0-1023-ig az ellenállás értékeket, majd ebből az adathalmazból kell kiválasztani azt az értéket, amelyhez a későbbiek során viszonyíthatunk. Ha például szenzor a vonal felett van, akkor az ADC értéke 770-780 körül van. Ha a padló felett, akkor pedig 910-920 körül. Ezért határértéknek egy 820-as érték lett beállítva. Ha a szenzor által érzékelt jel kisebb a beállított határértéknél, akkor a kiértékelő arra az eredményre jut, hogy a szenzor a vonal felett van. Ha a szenzor értéke nagyobb, mint a határérték, akkor az a padló feletti állapotot jelenti. A vonalérzékelő tesztelését a 12. ábra mutatja.

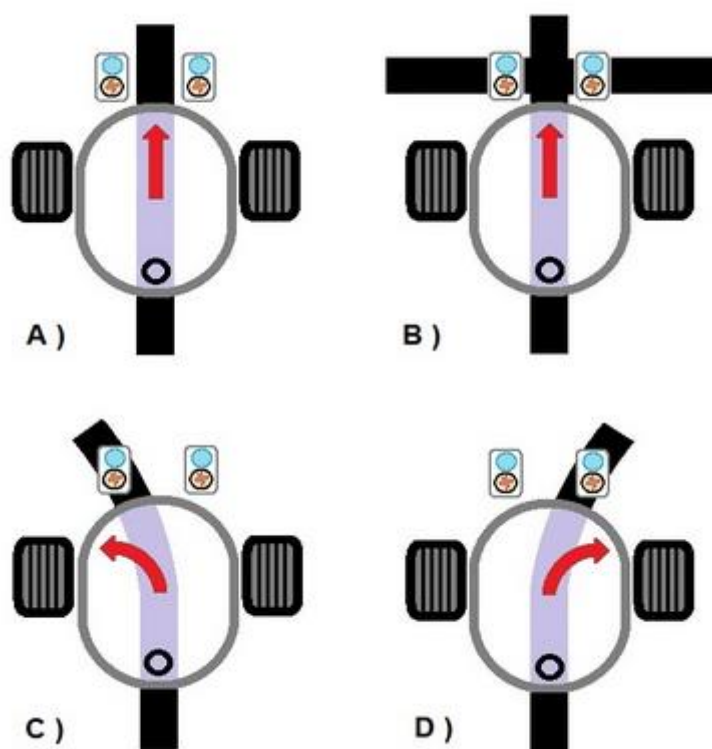


12. ábra: Szenzorok kalibrálása, tesztelése

12.2. Vonalkövető algoritmus

Miután értelmezhetővé váltak a vonalérzékelő szenzor adatai, és a robotunk képes az egyenes haladásra, a vonalkövető algoritmus és a robotvezérlő program elkészítésére kerül sor.

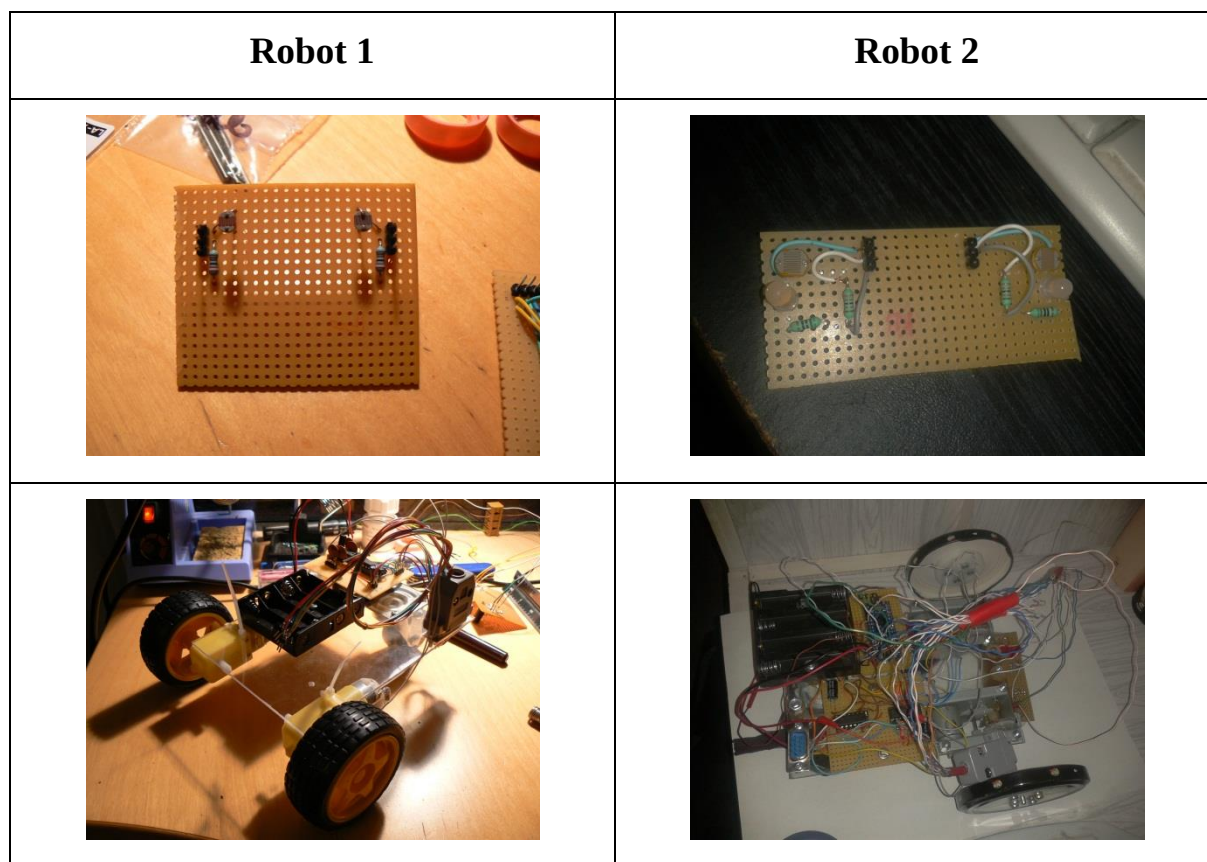
Bekapcsolás után az ADC-vel beolvasásra kerül először a jobb, majd a bal oldali LDR érzékelővel mért fényerősség. A mikro vezérlő a vonal pozíciójától függően úgy irányítja a robotot (úgy változtatja a robotot meghajtó DC motorok sebességét és forgásirányát), hogy a vonal mindig a két szenzor között maradjon. A fejlesztett robotok működését a 13. ábra szemlélteti, ha a bal oldali szenzor a vonal felett helyezkedik el (a bal oldali szenzor által mért fényerősség a beállított határértéknél kisebb), akkor a robot balra fordul (C). Ha a jobb oldali szenzor a vonal felett helyezkedik el (a jobb oldali szenzor által mért fényerősség a beállított határértéknél kisebb), akkor a robot jobbra fordul (D). Ha mindkét detektor jele a megadott határértéknél nagyobb (a vonal a két érzékelő között van), vagy ha mindkét detektor jele a megadott határértéknél kisebb (a robot kereszteződéshez ért), akkor a robot előre megy (A és B). A határérték változtatásával lehetőségünk van a robotot a környezeti fényviszonyokhoz hangolni. A következő ADC mérés előtt a mikro vezérlő várakozik 50 ms-ig. Ez az esetlegesen fellépő oszcillációk csökkentésére szolgál.



13. ábra: Navigálás a vonal mentén

Forrás: <http://hobbyrobot.hu/>

12.3. A megépített és szoftveresen felprogramozott robotok



14. ábra: A fejlesztett Robot 1 és Robot 2

13. Teszteredmények

A robotokat többször is teszteltük kisebb/nagyobb útvonalakon. A tesztelés során vettük észre, hogy a két motor nem azonos sebességgel mozog, evégett letért a kijelölt útvonalról, ezt a kerekeken két encoder tárcsával orvosoltuk, továbbá, ekkor merült fel, hogy a két szenzor bár elegendő azonban még is kevés, és túl távol vannak egymástól ezáltal a hirtelen kifordítás végett a robot szintén letért a kijelölt útról, ami ugye bár nem kívánatos. Ezt a problémát több szenzor felhelyezésével került kivédésre. Mind két robot esetén 50 mérés került elvégzésre, egy időben külön pályán, amelyek eredményeit a táblázat foglalja össze.

3. táblázat: Az 50 db mérés átlagának teszteredményei

	Robot 1		Robot 2	
	Pontosság	Hiba	Pontosság	Hiba
Útvonal 1	81%	19%	78%	22%
Útvonal 2	76%	24%	83%	17%

14. Összegzés

A teszteredmények alapján megállapítható, hogy a projekt elérte a célját, a kialakított robotok ha nem is tökéletesek, de a feladat szempontjából a projekt sikeresnek mondható. A hallgatók közös munkája, a csapatban való munkavégzés tapasztalatai segítik őket a későbbi feladataik megoldása során is. A projekt során olyan eszközök kerültek felhasználásra, melyek olcsón beszerezhetők, így a bemutatott megoldásban szereplő ötletek mások számára is, akár hasonló projektmunka során felhasználhatók.

Irodalomjegyzék

Banzy, M., & Shiloh, M. (2014). *Getting started with Arduino: the open source electronics prototyping platform*. Maker Media, Inc.

Chu, H. C., & Hwang, G. J. (2010). Development of a project-based cooperative learning environment for computer programming courses. *International Journal of Innovation and Learning*, 8(3), 256-266.

Gutkind, L. (2006). *Almost human: Making robots think*. WW Norton & Company.

Katona, J. et al (2016). Speed control of Festo Robotino mobile robot using NeuroSky MindWave EEG headset based brain-computer interface. In *2016 7th IEEE international conference on cognitive infocommunications (CogInfoCom)* (pp. 000251-000256). IEEE.

Katona, J. et al (2017). Hand controlled mobile robot applied in virtual environment. *World Academy of Science Engineering and Technology*, 11, 1430-1435.

Katona, J., & Kovari, A. (2016). A Brain-Computer Interface Project Applied in Computer Engineering. *IEEE Transactions on Education*, 59(4), 319-326.

Moursund, D. G. (1999). *Project-based learning using information technology*. Eugene, OR: International society for technology in education.

Moubarak, P. M., & Ben-Tzvi, P. (2011). Adaptive manipulation of a hybrid mechanism mobile robot. In *2011 IEEE International Symposium on Robot and Sensors Environments (ROSE)*, 113-118.

Moubarak, P., & Ben-Tzvi, P. (2012). Modular and reconfigurable mobile robotics. *Robotics and autonomous systems*, 60(12), 1648-1663.

Kovari, A. (2009). Hybrid Current Control Algorithm for Voltage Source Inverters. In *2009 First IEEE Eastern European Conference on the Engineering of Computer Based Systems*, pp. 65-70.

Kovari, A. (2015). Effect of leakage in electrohydraulic servo systems based on complex nonlinear mathematical model and experimental results. *Acta Polytechnica Hungarica*, 12(3), 129-146.

Orvis, K. L. (Ed.). (2008). *Computer-Supported Collaborative Learning: Best Practices and Principles for Instructors: Best Practices and Principles for Instructors*. IGI Global.