

Internet – alapú, alacsony költségvetésű mobilrobot építése oktatási célokra

Domonkos Márk ¹, Dr. Fekete Róbert Tamás ²
Mechatronika, Optika és Gépészeti Informatika Tanszék
BME, GPK
Budapest, Magyarország
domonkos@mogi.bme.hu ¹, frt@mogi.bme.hu ²

Absztrakt— A dolgozat címében is megjelenő témát járjuk körül. A projekt, ahogy a cím is említi, egy költséghatékony, internet – alapú mobilrobot tervezése és elkészítése, mellyel könnyen valósíthatók meg a különböző mobil robotikai feladatok. Ezt tettük elsősorban oktatási célokra és későbbi továbbfejlesztés esetén akár etorobotikai kutatások elősegítésére. A tervezés során fontos volt a lehető legnagyobb mértékű univerzalitás, egyszerűség és az alacsony anyagi keret. Jelen dolgozat ezen projekt koncepciójának vázolata.

Kulcsszavak—*mobilrobot, alacsony költségvetés, STM32, Raspberry Pi, etorobotika;*

I. BEVEZETŐ

A robotok már régóta könnyítik meg életünket. Az eredetileg Sci-Fi lény, melynek keresztapja Karel Čapek, és szó szerinti fordítás alapján cseh nyelvről annyit tesz munkavégzés, nem feltétlenül az a kétlábon járó humanoid, amire az átlagember gondolna.

A technológiai fejlődést szemügyre véve észrevehető bizonyos ko incidenciák. Az elektromosság jelenlétét legtöbbször már csak akkor tudatosítjuk igazán, ha valami hiba lép fel vele kapcsolatban. A néhány éve még hatalmas újdonságnak számító önjáró robotporszívóval is csupán akkor törődünk ma már, ha az javításra vagy esetlegesen más segítségre szorul. Hogy egy kissé ne ennyire hardware közeli példát is említsünk, Rodney Brooks, az MIT Mesterséges Intelligencia Laboratórium igazgatójaként kijelentette, hogy tévhit, miszerint a mesterséges intelligencia megbukott. Nap, mint nap használjuk és észre sem vesszük, hogy egyes helyeken a repülőgépeket ilyen rendszer irányítja, a Microsoft termékeinek többsége szintén alkalmazza, sőt mesterséges intelligenciával küzdünk valahányszor egy videójáték világában elmélyedünk [1].

Mivel valószínűsíthető, hogy a robotok is hasonló utat fognak bejárni, semmiképpen sem ördögtől való, ha a szakma diákjai minél behatóbban képesek tanulmányozni a robotok viselkedését, irányításuk esetleges nehézségeit. A projektünk során éppen ezért egy olyan platform létrehozására tettünk kísérletet, ahol az adott diák a lehető legtöbb információt képes begyűjteni és a lehető legegyszerűbben képes a robottal kommunikálni.

Mai gyakorlat merőben eltér ugyan az eredeti oktatás formájától. Napjainkban a prezentáció kivetítővel, jó

minőségű ábrákkal történik, a diákok számára pedig elérhető legalább az előadott prezentáció oldalai, jobb esetben egy jegyzet a tananyagról. Ugyancsak fontos elem az interaktivitás [2]. A tanulás pszichológiáját tanulmányozva azonban észre vehetjük, hogy az előadás hallgatása, illetve a szemléltető ábrák figyelése, azaz a tanulás passzív formája messze elmarad az aktív, azaz élmény alapú, oktatástól [3].

Az élmény alapú oktatás egyik nagy hátulütője lehet, az erőforráshiány. Az ilyen jellegű problémákat igyekeznek megszüntetni a tanszéken „online” mérések segítségével. Ilyenkor a diákok a saját időbeosztásuk alapján határozzák el, hogy mikor „látogatnak” el a laborba. Ezáltal minden diák egyénileg végezheti el a mérést, úgy, hogy a gép kihasználtságát ezzel növeltük és ugyanakkor a diákok számára is egy kényelmesebb alternatíva keletkezett. Mi sem mutatja jobban ez utóbbi állítás igazságtartalmát, mint hogy a diákok döntő többsége a méréseit az éjszakai órákban végzik.

A jelenlegi tanszéki mérés egy DC motor mérését teszi lehetővé. A diákok első nekifutásra ismerkedhetnek a D/A és az A/D átalakítókkkal, majd eljuthatnak a PI szabályzó megtervezésétől egészen a komplexebb szabályzó elemek, mint a csúszómód szabályozás megtervezéséig. A diákok ilyenkor valós képet kapnak a motor működéséről, melyet egy kis ablakban szemlélhetnek, továbbá a mérés eredményeit letölthetik és kiértékelhetik [4]. A tanszéken több ilyen irányú projekt működik, vagy dolgoznak rajta, mint például a fényességvizsgálat [5].

A soron következő fejezetben az előzetes terveket, elgondolásokat tárgyaljuk. Ezek után a 3. fejezetben bemutatásra kerül a robot sajáttervezésű energiamodulja. A 4. fejezetben az ultrahangos érzékelő és a reflexiós elven működő érzékelő felhasználását mutatjuk be. Ezt követően a beavatkozásszerv témakörét tárgyaljuk. A hardware bemutatását ezek után a Raspberry Pi és a Mikrovezérlő c. fejezetben zárjuk. A hardware rész befejezte után áttérünk a szoftverre, amit először a mikrokontroller programjának koncepciójával boncolgatunk, majd áttérünk a Raspberry Pi előkészítésével kapcsolatos beszámolóra. A szoftveres rész lezárásaként a Node – Red -el, mint a kezelő felület létrehozásához szükséges programozással foglalkozunk. Lezárásként pedig a továbbfejlesztési lehetőségeket boncolgatjuk nagyon röviden, majd összegzésként egy rövid összehasonlítást végzünk néhány már kész termékkel.

II. TERVEK

Az egyszerű elérés és kezelés érdekében úgy gondoltuk, hogy egy webes server létrehozása lenne a legideálisabb számunkra. Ezen csupán az érzékelők adatait láthatjuk és a beavatkozóelemek felé támasztott igényeinket közvetíthetjük.

Az építés folyamán szükségesnek láttuk, hogy a robot kezelője képet kapjon a robotról, ehhez egy webkamera beépítését tartottuk célszerűnek. Ezért szükségünk volt egy olyan eszközre, ami képes a webkamera jelét lekezelni, és ugyanakkor ezzel egyidőben a már említett servert fentartani. Ennek érdekében döntöttünk úgy, hogy a robot fő irányító egysége egy Raspberry Pi beépítésével valósul meg.

A robotnak továbbá szüksége volt egy egységre, mely az érzékelők legkülönbözőbb jeleit – impulzusok száma, analóg jel nagyság, impulzusok szélessége – képes fogadni és előfeldolgozni, valamint esetlegesen a beavatkozók számára is tud biztosítani vezérlőjeleket. Erre a célra egy STM32F103C8T6 típusú mikrovezérlőt használtunk, soros kommunikációval illesztve a Raspberry Pi -hoz.

A robotot felszereltük 6 darab ultrahangos érzékelővel, melyek a tereptárgyakat figyelik és 4 darab a vízszintessel szöveget bezáró reflexiós elven működő érzékelővel. Ez utóbbinak a szerepe a sarkokban való elakadás elhárításának, illetve a szakadékok elkerülésének elősegítése.

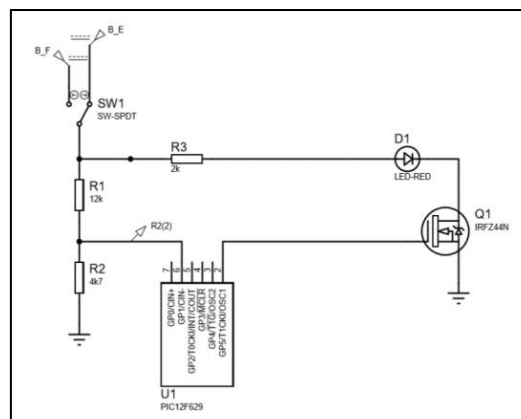
III. ENERGIAMODUL

A robotnak szüksége van megfelelő táplálásra, amit az energiamodul biztosít. Ennek központi eleme a 4 darab sorba kötött 18650 -es típusú lítium – ion akkumulátorcella. Ezek névleges feszültsége a megszokott 3,7 V, nominális kapacitásuk pedig 3500 mAh.

Az akkumulátorpakkhoz egy saját tervezésű akkumulátormegfigyelő rendszert terveztünk és építettünk, mely áll egy feszültségosztóból, FET tranzisztorból és egy Microchip PIC12F629 mikrovezérlőből. Mivel a mikrovezérlő nem képes elviselni az akkumulátorpakk feszültségét ezért egy feszültség szabályzó egységet is beépítettünk a megfelelő bekötésben.

A mikrovezérlő a feszültségosztóból érkező analóg feszültséget átalakítja számmá. Ez alapján kapcsolja a mikrovezérlő a FET tranzisztort, figyelembevéve a feszültség osztó tulajdonságait úgy, hogy az említett kapcsolás akkor következzen be, ha az akkumulátorok feszültsége 3 V alá esik.

Kihangsúlyozandó, hogy ez egy egyszerűsített megfigyelő egység a költségek alacsonyan tartása végett. Az fenti 1. ábrán láthatjuk a bekötés szimulációját, melyet ezek után meg is építettünk a már említett megfelelő kiegészítésekkel.



1. ábra: Szimuláció

Használat közben ugyan már nem tudjuk mélykísütésbe vinni az akkumulátorpakkunkat, tölteni viszont még nem tudjuk. A piacon fellelhető sok megoldás közül egy programozható töltőt választottunk. Ez az iMAX B6AC nevet viseli. Noha beépíthető lett volna más integrált áramkörös megoldás is, mellyel az akkumulátorok töltése némiképpen egyszerűsödött volna, az ára azonban nem volt kielégítő.



2. ábra: Energiamodul beépítve

A teljesen kész modult az akkumulátorokkal, és kiegészítő elemeivel a 2. ábrán láthatjuk, már felszerelt állapotban a robot alján. A helyválasztás több szempont alapján történt. A berendezés alján elhelyezkedés az akkumulátorok esetleges meghibásodása esetében kisebb eséllyel okozhatnak sérülést. További előny, hogy így felszabadul hely a platform tetején és esztétikusabbá is teszi a robotot.

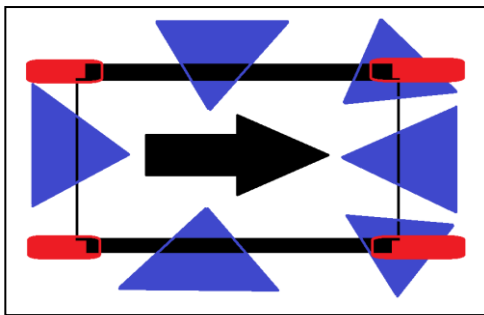
IV. ÉRZÉKELŐK

A robot építése során sokszor használtuk az „oszd meg és uralkodj” elvét, esetünkben ez itt is igaz az érzékelőkre. A robot távolságérzékelők két típusával van felszerelve, mindegyik típust célorientáltan választottuk. Ultrahangos elven alapuló érzékelő a nagyobb tereptárgyak érzékelésére és infravörös fény segítségével felhasználó szenzor a közelebbi

csapdák és az ultrahangos érzékelők elhelyezéséből eredően „láthatatlan” csapdák elkerülésére, mint például a szakadékok.

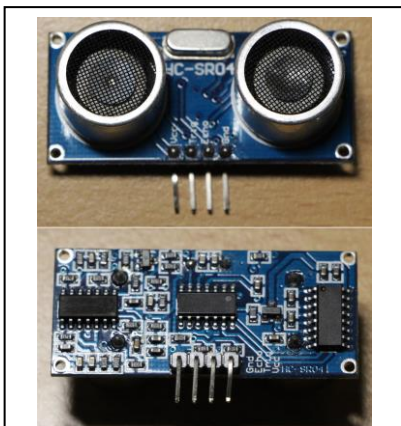
A. Ultrahangos elven működő érzékelő

HC – SR04 típusú ultrahangos szenzort használtunk fel, szám szerint 6 darabot. 3 darab került a robot elülső felére, a főmenetirányba, illetve attól jobbra és balra 30° eltéréssel. A robot elsődleges menetirányára merőlegesen szintén került egy – egy érzékelő, az elsődleges menetiránynak háttal szintén egy. Nem tartottuk fontosnak a robot hátoldalára felszerelni szintén 3 érzékelőt, mivel az elsődleges cél a robot előre történő haladása tehát ebből következően feltételezhető, hogy a robot többé – kevésbé „ismeri” a mögötte elterülő terepviszonyokat. Amennyiben az első feltétel mégsem teljesülne, akkor viszont a reflexiós elven alapuló érzékelők biztosítanak még további védelmet. Noha a szenzor ára nem jelentős, költségsökkentés szempontjából szintén előnyös a kevesebb érzékelő. A 3. ábrán látható az elrendezés vázlata.



3. ábra: Szenzorok elhelyezkedése a roboton (kékkel: ultrahangos, pirossal: reflexiós)

A szenzor felépítése, noha bonyolult, használatát mondhatni direkt a hobbistákra gondolva, egyszerűre készítették. Kimenete impulzus szélesség alapú, ezért nincs szükség analóg – digitális átalakításra a jel értelmezéséhez. Ez nagyban megkönnyíti az Arduino alapú vagy Raspberry Pi alapú projektekben való alkalmazását, így a hobbi projektek körében rendkívül népszerű érzékelő. A 4. ábrán látható maga az eszköz.

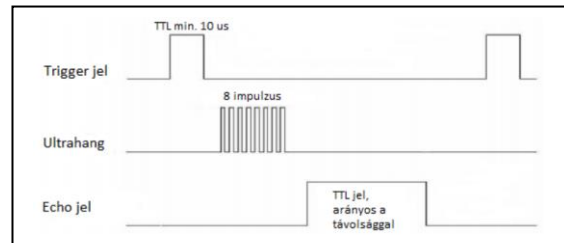


4. ábra: HC – SR04 (fent: előnézet, lent: hátlap)

A HC – SR04 táplálása 5 V feszültséggel történik melynél tipikusan 15 mA áramot fogyaszt. A mérhető távolság 20 – 4000 mm között van, melynek mérési szöge 15° [6].

Működtetéséhez szükséges egy Trigger impulzus, aminek a minimális szélessége 10 μ s. Az érzékelő ezt követően küldi ki a 8 darab 40 kHz frekvenciájú ultrahangot. A művelet végén az Echo jelre várunk, illetve mérjük a szélességét, ami arányos a tárgy távolságával [6].

A folyamat az 5. ábrán látható. A jel szélességéből és a hang terjedési sebességéből az (1.) közelítő összefüggés kapható, ahol t a mért impulzus szélessége és d a tárgy távolsága.



5. ábra: HC – SR04 használata

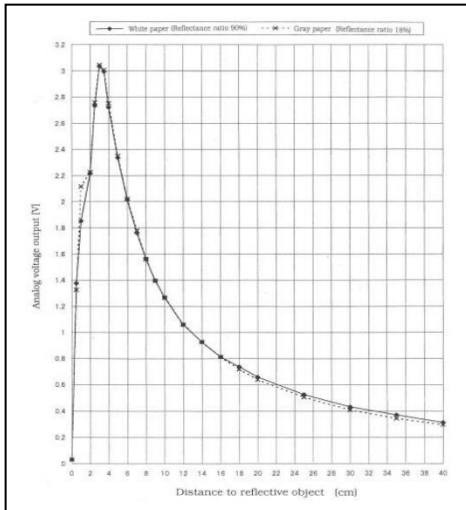
$$d \approx 0,5 \cdot t \cdot 340 \quad (1.)$$

B. Reflexiós elven működő érzékelő

A reflexiós elvű érzékelőt 4 darab SHARP GP2Y0A41SK0F típus testesíti meg. A szenzor infravörös fény kibocsátásával majd annak reflexiójával képes a tárgy távolság meghatározására (PSD érzékelő segítségével). Ennek függvényében analóg jelet bocsajt ki, mely esetben szükséges az átalakítás.

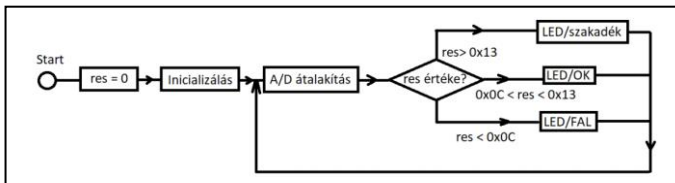
Táplálása 4,5 V és 5,5 V közötti feszültséggel történik, ilyenkor tipikusan 12 mA, maximálisan pedig 22 mA áramot fogyaszt. A szenzorral mérhető tárgy távolság 40 mm és 300 mm közötti tartományban található, mely alatt a kimeneti feszültségkülönbség tipikusan 2,25 V [7].

Az érzékelő adatlapján leolvasható, egy bizonyos tartomány, ahol a jellegzőbe irányt változtat, ezzel azonban különösebben nem kell foglalkozni, mivel az adott tartományban az eszközt nem használjuk. A 6. ábrán látható a távolság – kimeneti feszültség diagram.



6. ábra: SHARP GP2Y0A41SK0F távolság – kimeneti feszültség diagram [7]

Az adatlap, illetve az előzetes kutatás során talált információk figyelmeztettek, hogy az érzékelő nem feltétlen alkalmas a tárgytávolság mérésére nem merőlegesen elhelyezett tárgyak esetében, ahogyan ez esetünkben is megfigyelhető. Ennek tesztelésére készítettünk egy mérési elrendezést, mely az érzékelőből, egy mikrovezérlőből, egy állványból és egy mérendő tárgyból állt. A mérést elvégeztük fehér, illetve nem fehér tárggyal is. A mérés során kapott eredményekből azonban arra következtethetünk, hogy ez a probléma megkerülhető. A mérés eredményét látva az érzékelő alkalmas az általunk meghatározott feladat elvégzésére. A méréshez használt program vázlatát a 7. ábra mutatja.

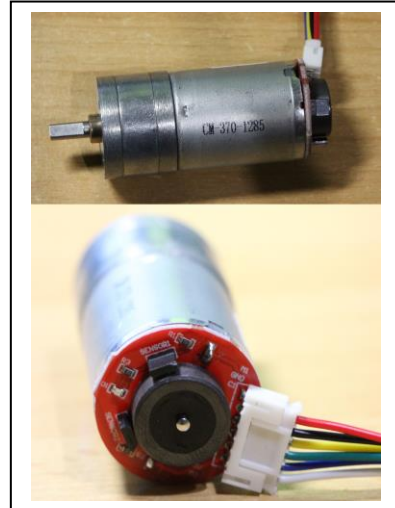


7. ábra: Tesztprogram

V. BEAVATKOZÓSZERV

A hely és költségcsökkentés érdekében a robotot két áttétellel és encoderrel egybeépített egyenáramú motor hajtja. Ezeket külön tudjuk vezérelni a motorvezérlő elektronika segítségével, kétféle forgási értelemmel. Mivel az említett motorról rendkívül kevés információ állt rendelkezésünkre, ezért néhány alapvető mérést elvégeztünk, mint például egy adott PWM kitöltési tényező függvényében mért encoder impulzusok száma, adott időintervallum alatt, amiből egy közelítő feszültség – fordulatszám karakterisztika kapható. A motor elérhető paraméterei a kapocsfeszültség, ami 12 V, névleges kihajtótengelyi szögsebesség, 915 RPM, indító- és felvett árama pedig ≤ 750 mA és ≤ 150 mA.

Az encoderes mérés érintésmentesen, Hall szondával, történik, minek eredményeképpen két csatornán kapunk impulzusokat. Az encoder táplálását külön kell megoldanunk 3,3 V táplálással. A motort az áttétellel és az encoderrel a 8. ábrán láthatjuk.



8. ábra: A kiválasztott motor (fent: a teljes motor, lent: az encoder)

Mivel a mikrovezérlő nem képes a megfelelő teljesítmény leadására, ezért egy L298n motorvezérlő modult is beépítettünk, noha ettől olcsóbb és kisebb helyigényű megoldások is léteznek.

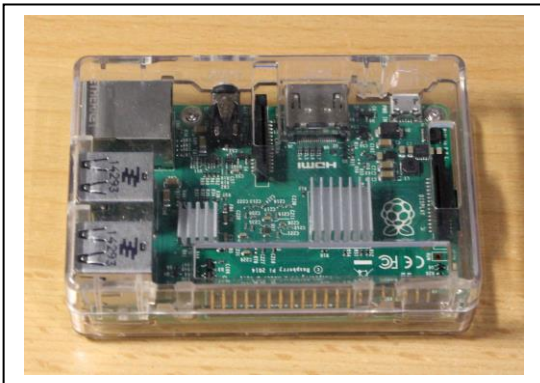
VI. RASPBERRY PI

Az említett eszköz már inkább nevezhető termékcsaládnak révén a rengeteg variánsának. Jelen projektben egy Raspberry Pi 2 B+ modellt használunk fel minden előnyével és hátrányával együtt. Az egykártyás számítógépet eredetileg a számítástudományok népszerűsítésére fejlesztették ki főként a fejlődő országokba. Az eszköz azonban a hobbi – építők körében is nagyon elterjedt. Jelen esetben ott tartunk, hogy a kiválasztott típust főleg csak a specifikációnk határozza meg, hiszen csak, hogy a processzort említsük, a variánsok terén a 700 MHz és az 1,2 GHz közötti órajel frekvencia és a több magos megoldások között is válogathatunk. Variálhatunk továbbá még a csatlók mennyisége és típusai között is [8].

Esetünkben a Raspberry Pi 2 alábbi tulajdonságait használjuk ki. Mivel nem rendelkezik beépített WiFi csatlakozóval ezért egy USB porton keresztül csatlakozó WiFi csatlót fogunk beépíteni. Ahogyan az már említésre került a robotot fel kell szerelni egy kamerával is. Erre a célra szintén egy USB porton keresztül csatlakozó eszközt találtunk. Felmerülhet a RaspiCam alkalmazása, ezt azonban jelen projekt esetében a költségek csökkentése érdekében elvetettük, előnyeiről a későbbiekben számolunk be, a továbbfejlesztési lehetőségek tárgyalásánál. Szükségünk lesz továbbá még a kivezetéseknél a soros kommunikáció érintkezőire, minek segítségével az STM32 mikrovezérlővel fogunk kommunikálni. A projekt során egy 8 GB kapacitású microSD kártyán tároljuk az operációs rendszert, ami esetünkben a Raspbian OS. Ez egy Debian alapú operációs rendszer,

esetünkben a 2017-11-29 kiadás, ami Stretch alapú, 4.9 Linux kernellel rendelkezik [9].

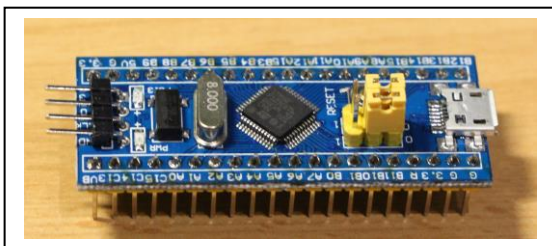
Természetesen egy műanyag doboz segítségével védjük az eszközt a fizikai behatásokkal szemben. Erről a 9. ábra is tanúskodik.



9. ábra: Raspberry Pi 2 dobozban

VII. MIKROVEZÉRLŐ

Több lehetőséget összehasonlítva az STM32F103C8T6 típus mellett döntöttünk. Pontosabban annak egy modul verziója mellett, ami a Minimum System Board nevet viseli, vagy a köznyelvben csak „blue pill” modulként emlegetik. A modul előnye a rendkívüli olcsósága és viszonylag kis mérete. További előnye még, hogy a gyártó által forgalmazott programozó interface szintén sokkal olcsóbb, néhány más gyártó által forgalmazottával összehasonlítva. Az alábbi 10. ábrán láthatjuk a modult.



10. ábra: STM32 „blue pill” modul

A mikrokontroller egy közepes sűrűségű ARM 32-bit Cortex – M3 alapú mikrovezérlő, melynek a maximális CPU órajel frekvenciája 72 MHz -ben van meghatározva és képes akár az 1,25 DMIS/MHz teljesítményre. A szériában 64 – 128 kbyte memória és 20 kbyte SRAM található [10] [11].

A tápellátás igénye a processzornak 2 V és 3,6 V feszültség között van, ezt azonban a modul segítségével 5 V vagy picivel az fölé is állíthatjuk köszönhetően a modulban elhelyezett 3,3 V -os regulátornak.

Az általunk használt főbb perifériák az analóg digitális átalakítók, időzítő egységek, közvetlen memória elérési periféria és a soros kommunikációért felelős kommunikációs periféria, valamint természetesen az I/O kivezetések.

A. Analóg digitális átalakító

Mivel a reflexiós elven működő érzékelő analóg jelet biztosít számunkra, ezért szükségünk van annak absztrahálására, valamilyen számértékké konvertálására, a feldolgozás érdekében.

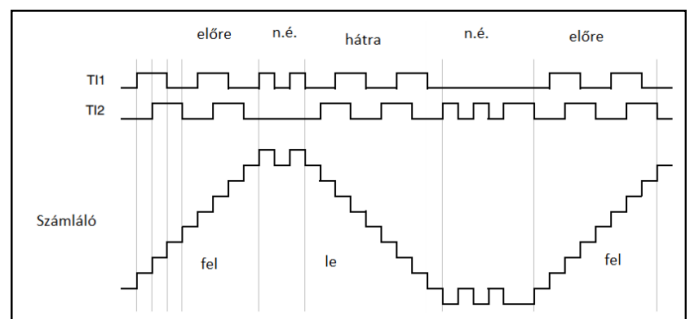
A két átalakítóperiféria közül csupán az egyiket használjuk, Continuous scan mode -ban. Ilyenkor az átalakító a rákötött négy csatornát – az általunk megadott sorrendben - letapogatja, majd az utolsóval végezvén az elsőre ugrik vissza. Az eredmény DMA (Direct Memory Access) segítségével kapjuk meg. A mintavételezési időt kellően nagyra – 71,5 ciklus – választottuk [10] [11].

B. Timer perifériák

A mikrovezérlőben egy Advanced Control Timer (TIM1) és további három általános célú (TIM2, TIM3, TIM4) időzítő periféria található. A Timer perifériák esetünkben több funkciót is el fognak látni. Szükségünk van egy adott frekvenciájú „órajelre”, ami segíti az időzítéseket, a motorokra kiadott PWM jelet is timer egységekkel oldjuk meg, sőt még az encoder impulzusait is az időzítő periféria segítségével fogjuk el.

A PWM jelet a Timer1 biztosítja, ami egy 16 bites, automatikusan újratöltődő számláló, programozható előosztóval. A megfelelő konfigurációk segítségével két 8 bites felbontású PWM jel generálása történik, kHz -es nagyságrendben lévő frekvenciával. Ezeket, a megfelelő kivezetések segítségével, a motorvezérlő modulra kötjük.

Kihasználjuk a periféria egy másik számunkra rendkívül hasznos lehetőségét, ami az Encoder Interface nevet viseli. Ez mondhatni egy egyszerű fel – le számláló, adott logikával. A TIM3 és TIM4 két – két csatlakozójára kötjük az encoder két – két csatornáját és a belső logika, illetve a saját beállításaink, mondhatni elvégzik a többi munkát. Ez a fajta interface sokban megkönnyíti az encoderekkel való munkát egy ilyen projekt során, mivel ezek után csupán annyi a dolgunk, hogy lekérjük az interfésztől a „lépések” számát. A működés vázlatát a 11. ábra illusztrálja. A vázlatból is jól látszódik, hogy az interface biztosítva van több módon is az esetleges hibák ellen, az ábrán ezek közül jól látható a bizonytalan állapot ignorálása [10] [11].



11. ábra: Encoder interface működési sémája

TIM2 feladata mondhatni a legegyszerűbb, csupán adott frekvenciával biztosít egy „órajelet” amihez szervezzük a robot működését.

C. Kommunikációs periféria

A Raspberry Pi -vel való kommunikációra a mikrovezérlő USART egységét fogjuk használni. Ez az egyik legegyszerűbb kommunikációs protokollok egyike. Esetünkben nem probléma a viszonylagos kis adatátviteli sebesség, mivel az általunk kommunikált adatok mennyisége kicsi, és keletkezésük üteme sem lehet túlzottan nagy.

D. I/O

Az ultrahangos érzékelők kezelését egyszerűen az I / O kivezetésekkel oldjuk meg. A szenzor Trigger érintkezőjét a mikrovezérlő előzőleg konfigurált kimeneteire – 5 darab –, míg ezzel szemben az Echo érintkezőkre a mikrovezérlő bemeneteit kötjük. Ennek és a már tárgyalt ultrahangos érzékelőknek a kezelési tulajdonságaiból építkezve tervezhetjük meg a kiszolgáló szubrutint.

VIII. A MIKROVEZÉRLŐN FUTÓ PROGRAM

Az STM32 -re megírt program egy többszintes, megszakításokkal ellátott, végtelen ciklus. A főprogram végtelen ciklusként 0,5 s -onként elküldi a begyűjtött információkat a Raspberry Pi felé.

A program másik fő része a 20 Hz -es megszakítássorozat lekezelése, melyben az adatok gyűjtésének nagyrésze történik. Itt kérdezzük le az encodereket, majd a kapott adatokat formázzuk kiküldhető formátumúra. Itt történik továbbá a kívánt PWM jel kiadása, valamint az ultrahangos érzékelők adatainak lekérése. Ez utóbbi esetben egyesével lekérdezi az ultrahangos érzékelőket, majd ezekből távolságokat számol a megfelelő összefüggéssel.

Egy másik megszakítás az analóg digitális átalakítást kezeli le. Itt tulajdonképpen a négy reflexiók elven működő szenzor kimenetét kiolvassuk, ezek után eldöntjük, hogy az érzékelő előtt „semmi”, „szakadék” vagy „fal” van. Végül az eredményből egy kódot állítunk elő, mely segítségével pontosan beazonosítható, hogy melyik érzékelő esetében találtunk problémás helyzetet és melyik érzékelő esetében nincs semmilyen akadály.

A 20 Hz -es megszakítás egy további fontos részleme, ahol a motorok mozgásának irányát állítjuk be. Itt egy feltételes elágazással kerülünk szembe. Amennyiben nincsen semmilyen akadály, akkor egyszerűen engedélyezzük a motorok adott irányba történő elmozdulását. Azonban, ha akadályba ütköznénk, a robot annak függvényében, hogy az irányítás ember által, vagy software által történik a robot megáll az akadály észlelésekor, illetve ignorálja a vészjelzést. Ennek oka, hogy az esetleges károkat könnyebben tudjuk elkerülni, úgy, hogy a felhasználó saját beavatkozását nem bíráljuk felül.

Természetesen a program végső összeállítása előtt a perifériák tesztelése és a szenzorok és beavatkozásszervek működtetésének tesztelése végett tesztprogramokat készítettünk minden részletre kiterjedően.

IX. RASPBERRY PI SZOFTVERE

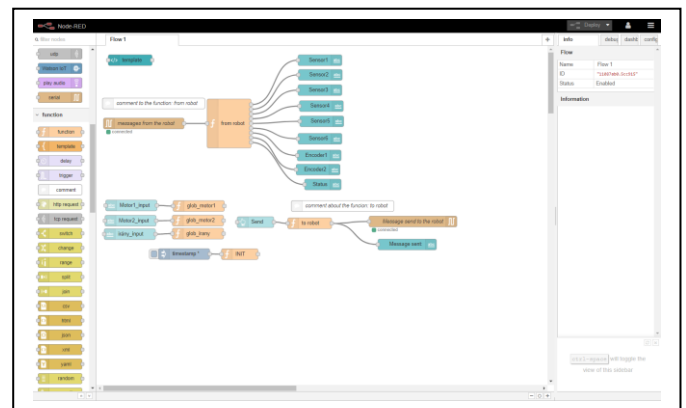
A Raspberry Pi -on az operációs rendszer frissítése után szükséges néhány beállítást elvégeznünk. Elsőként a WiFi csatoló élesztését, csatlakozási lehetőséget, majd a statikus IP cím beállítását. Ezt követően csatlakoztatjuk a webkamerát, melyhez a Motion szoftvert fogjuk alkalmazni. Ha a Motion -t kellőképpen konfiguráltuk, akkor a Raspberry Pi IP címe alatt a 8081 -es porton megtalálhatjuk a kamera képét a böngészőből. Ezt követi a Node – Red installálása és frissítése. Ha ezzel is megvagyunk, akkor még néhány alapvető beállítás, jelszó, felhasználónév, szükséges igény szerint a programozás kezdte előtt.

X. NODE – RED

A Node – Red egy adatfolyam alapú programozási nyelv, melyet eredetileg az IBM mérnökei fejlesztettek ki IoT projektek gyors prototípezéséhez. Később nyílt forráskódúvá tették [12].

A saját programunkat három fő egység alkotja. Az első egység egy Template node -ból áll, melynek feladata, hogy a webkamera képét a megfelelő formátumban továbbítsa a kezelői felületre. Második egység feladata, a mikrokontrollerből kapott adatok formázása és megjelenítése. Ehhez a mikrovezérlőből kapott adatokat szét kell válogatni, majd a megfelelő UI elembe továbbítani, illetve a kezelőszoftvernek megküldeni. Harmadikként pedig a felhasználó által küldött parancsokat kell elfogni, formázni, majd továbbítani a mikrokontrollerhez. A program a 12. ábrán látható.

Ajánlott még a programba beiktatni TCP node -okat, hogy a szoftveres úton történő kommunikációját a felhasználónak megkönnyítsük. Ilyenkor egy egyszerű csatlakozó algoritmussal már képesek vagyunk a robotot irányítani



szoftvert létrehozni.

12. ábra: Node – Red program.

XI. TOVÁBBFEJLESZTÉS

Az elkészült robotot a 13. ábrán láthatjuk. Az alábbiakban pontokba szedve összegezzük a platform továbbfejlesztésének általunk vélt főbb irányait.

- Raspberry Pi 3 vagy egyéb hasonló kategóriás Raspberry Pi használata, a beépített WiFi és a nagyobb teljesítmény elérésére.
- RaspiCam használata, a képfrissítés és minőség javítására és a buszrendszeren történő akadályozások elkerülésére.
- FET tranzisztor alapú motorvezérlő hídkapcsolás beépítése az L298n modul helyett, alacsonyabb fogyasztás érdekében.
- A mikrokontroller nyomtatott áramköri lapjának áttervezése kisebb helyigény érdekében, a motorvezérlő hídkapcsolással való integrálása.
- Kész akkumulátorpakk használata.
- Befoglalóméretetek csökkentése a robot vázánál.
- Motion finomhangolása, esetleges lecserélése más szoftverre.
- Windows 10 IoT Core alapú operációsrendszer / applikáció használata.

1. táblázat: A robot anyagköltségei

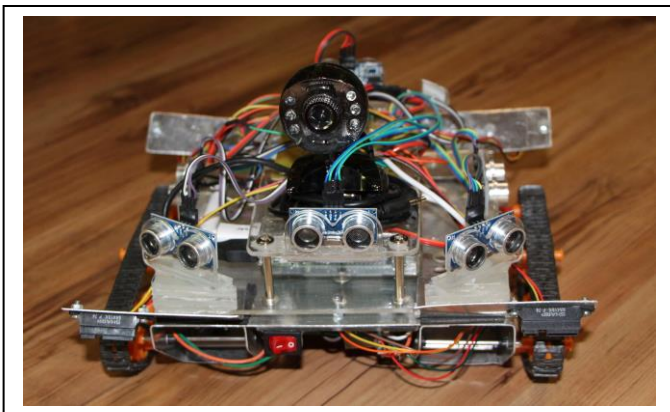
Megnevezés	Ár [€]	Megnevezés	Ár [€]
Vázszerkezet	15	2x DC motor	20
Raspberry Pi 2 B+	40	4x Sharp IR	15
Egyéb mech. elemek	10	STM + NYÁK	5
Webkamera	20	WiFi csatoló	10
6x HC SR04	5	Akkumulátorok	15
Energiamodul egyéb alkatrészei	15	Összesen	170

KÖSZÖNETNYILVÁNÍTÁS

EFOP-3.6.3-VEKOP-16-2017-00001: Tehetséggondozás és kutatói utánpótlás fejlesztése autonóm járműirányítási technológiák területén - A projekt a Magyar Állam és az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

HIVATKOZÁSOK

- [1] Kurzweil, R., A szingularitás küszöbén, Ad Astra kiadó, 2014, p. 945
- [2] Fedak V, Bauer P, Hájek V, Weiss H, Davat B, Manias S, Nagy I, Korondi P, Miksiewicz R, Duijsen P, Smékal P. „Interactive E-Learning in Electrical Engineering” In: Proceedings of the 15th International Conference on Electrical Drives and Power Electronics EDPE-2003. Konferencia helye, ideje: Podbanské, Szlovákia, 2003.09.24-2003.09.26. Košice: Technical University of Kosice, 2003. pp. 24-31.(ISBN:808906177X)
- [3] H. Weiss, A. Schmidhofer, A. Schmid, V. Hajek, B. Davat, S. Manias, I. Nagy, P. Korondi, K. R. Jordan, R. Miksiewicz, V. Fedak, P. Smekal, P. Bauer, P. van Duijsen, “Animated and Interactive E-Learning Concept and Realization,” in Proceedings of the IASTED International Conference on Web-Based Education (WBE), 2004
- [4] G Sziebig, B Takarics, P Korondi, “Control of an embedded system via internet”, in IEEE Transactions on Industrial Electronics 57 (10),pp. 3324-3333, 2010
- [5] K. Samu, B. Thamó Internet based light quality measurement RECENT INNOVATIONS IN MECHATRONICS 4:(1) pp. 1-4. (2017) eISSN: 2064-9622
- [6] Elecfreak.com, „Ultrasonic ranging module HC - SR04, adatlap,” [Online]. Elérhető: <http://www.electroschematics.com/wpcontent/uploads/2013/07/HCSR04-datasheet-version-1.pdf>. [Hozzáférés dátuma: 2018.01.28.].
- [7] SHARP, „GP2Y0A41SK0F adatlap,” [Online]. Elérhető: <https://www.pololu.com/file/0J713/GP2Y0A41SK0F.pdf>. [Hozzáférés dátuma: 2018.04.30.].
- [8] Raspberry Pi Foundation, „raspberrypi.org,” [Online]. Elérhető: <https://www.raspberrypi.org>. [Hozzáférés dátuma: 2018.01.22.].
- [9] i.sz., „raspbian.org,” [Online]. Available: <https://www.raspbian.org>. [Hozzáférés dátuma: 2018.01.23.].
- [10] STMicroelectronics, RM 0008 Reference manual, 2011, p. 1096.
- [11] STMicroelectronics, STM32F103x8 / STM32F103xB adatlap, 2015, p. 117.
- [12] „Node-Red hivatalos oldala,” JS - Foundation, [Online]. Elérhető: <https://nodered.org/>. [Hozzáférés dátuma: 2018.02.03.].
- [13] Mondada, F. et al., „The e-puck, a Robot Designed for Education in Engineering”, in: Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions, 2009



13. ábra: A kész robot

XII. ÖSSZEGZÉS

Az alábbi 1. táblázatban található a robotunk anyagköltsége. Az értékeket kerekítettük. Látható, hogy a kész robot elkészítési költsége hozzávetőlegesen 170 €, munkaköltséget itt szándékosan nem számoltunk. Egy érdekes oktatásban használni kívánt megoldás az e-puc, mely szintén egy a mi robotunkhoz hasonlós mobilrobot, melyről a [13] -ban olvashatunk a cikkben 250 € -s költségről számolnak be, illetve a hasonló jellegű eszközöket 550 € költségre teszik.

A piacon elterjedt eszközök árának alsó kategóriáját sikerült elérnünk, így a K-team Khepera IV többszöröse az általunk bemutatott robotnak. Hasonló a helyzet akár a MIR robotokkal is. Tény viszont, hogy itt az eszközök célja is más. A LEGO Mindstorms termékcsalád termékei viszont hasonló célokra tervezett játékszettként körülbelül 300 € ellenértékben vásárolható meg.