

**Szegedi Tudományegyetem
Informatikai Tanszékcsoport**

SZAKDOLGOZAT

Zana Roland

2013

**Szegedi Tudományegyetem
Informatikai Tanszékcsoport**

**Négykerekű mobilrobot vezérlőelektronikájának
tervezése és építése**

Szakdolgozat

Készítette:

Zana Roland
mérnök-informatika
szakos hallgató

Témavezető:

Dr. Mingesz Róbert
egyetemi adjunktus

Szeged
2013

Feladatkiírás

Egy mobil robot építése során a mechanika mellett alapvető fontosságú maga a működést vezérlő elektronika. Ennek feladata a szenzorok jeleinek érzékelése és feldolgozása, az aktuátorok meghajtása, kapcsolattartás a külvilággal (számítógépek, robot társak) valamint a vezérlő szoftver futtatása.

A szakdolgozat célja egy négykerekű robot működéséhez szükséges szenzorok kiválasztása és tesztelése, a vezérlőelektronika megtervezése, megvalósítása, tesztelése és felprogramozása, valamint a fejlesztési folyamat részletes dokumentálása. Az elkészített robothardver alkalmas kell, hogy legyen útvonalkövetésre, akadályelkerülésre, kommunikációra a külvilággal. Maga az elkészített robot, valamint az építése során szerzett tapasztalatok további szakdolgozattémák alapjául szolgálnak majd.

Tartalmi összefoglaló

- **A téma megnevezése:**

A témám egy négykerekű mobilrobot vezérlőelektronikájának tervezése és építése.

- **A megadott feladat megfogalmazása:**

A szakdolgozatom témájának megfogalmazása alapján, a feladatom egy olyan négykerekű mobilrobot szenzorainak és aktuátorainak kiválasztása és letesztelése, amelyek egy robot funkcionális működéséhez és mozgásához elengedhetetlenek. Továbbá a vezérlőelektronika megtervezése, megvalósítása, tesztelése és felprogramozása. Fontos volt, hogy olyan hardvert építsek meg, mely a későbbiekben is továbbfejleszthető, felhasználható. A teljes fejlesztési folyamat dokumentálása volt a szakdolgozatom egyik legfőbb feladata.

- **A megoldási mód:**

A megoldási mód első fázisa a szükséges információk felkutatása volt. A szenzorok és aktuátorok megismerése, azok felhasználása. A robot rendszerének blokkvázlata jó kiindulási alapot biztosított a robot vezérlőelektronikájának megtervezésében. Ez után sor került számos integrált áramkör adatlapjának tanulmányozására. Ezek tették lehetővé az áramkörök megtervezését. A nyomtatott áramköri lapok kialakítása után, lehetőség nyílt a teljes mértékű tesztelésre, majd a vonalkövető funkció megírására.

- **Alkalmazott eszközök, módszerek:**

A munkám során az áramköröket tesztelés céljából elsőnek próbapanelen raktam össze. Az egyes részfeladatok működését a C8051F120 és C8051F410 mikrovezérlők fejlesztő kit-jeivel teszteltem. Az ütközésmentesítő szenzorok letesztelésekor LabView-t és a cDAQ nevű mérőműszert használtam fel. Az áramkörök és a panelek tervezésére az Eagle nevű programot használtam. A programozás során a Silabs Laboratories által kifejlesztett Confirutaion Wizard 2 és az Integrated Development Environment (IDE) programokat használtam fel.

- **Elért eredmények:**

Elért eredményeim közé sorolnám a kiválasztott szenzorok és aktuátorok letesztelését. Továbbá az áramkörök, és a panelek megtervezését, forrasztását, tesztelését, a rendszer összerakását és a rendszer teljes mértékű letesztelését. A vonalkövetést megvalósító szoftver írása volt az utolsó eredményeim egyike.

- **Kulcsszavak:**

szenzorok, aktuátorok, mikrovezérlő, áramkörtervezés, paneltervezés, vonalkövetés

Tartalomjegyzék

Feladatkiírás.....	3
Tartalmi összefoglaló.....	4
Tartalomjegyzék	5
Bevezetés	7
1. Irodalmi áttekintés	8
1.1. Szenzorok.....	8
1.1.1. Ütközésmentesítő szenzor.....	8
1.1.2. Vonalkövető szenzorok.....	9
1.1.3. Ultrahang alapú távolságszenzor	9
1.1.4. Odometria optokapu használatával	10
1.1.5. Mozgás követése inerciaszenzorok segítségével	10
1.2. Aktuátorok.....	12
1. 2. 1. DC motorok vezérlése	12
1. 2. 2. Szervo motorok vezérlése	14
1.3. A C8051F120 mikrovezérlő bemutatása.....	14
1.3.1. A mikrovezérlőbe épített ADC leírása.....	14
1.3.2. A mikrovezérlő időzítő áramkörei	16
1.3.3. Programozható számlálótömbök (PCA) kezelése.....	17
1.3.4. Kommunikáció az SPI interfészen keresztül	18
1.4. A C8051F410 mikrovezérlő bemutatása.....	22
2. Szenzorok tesztelése	23
2. 1. Ütközésmentesítő szenzor tesztelése.....	23
2. 2. Vonalkövető szenzorok tesztelése.....	25
2. 3. Ultrahang alapú távolságszenzor tesztelése	25
2. 4. Optokapu tesztelése.....	27
3. Aktuátorok tesztelése.....	28
3. 1. Szervo motor tesztelése.....	28
4. Mikrovezérlők és funkcióik.....	30
4. 1. C8051F120 mikrovezérlő.....	30
4. 1. 1. Timer konfigurálása	30
4. 1. 2. A PCA konfigurálása	31
5. Áramkör és paneltervezés.....	32
5. 1. A robotvezérlő elektronika blokkvázlata	32
5. 2. A fő vezérlőpanel	32

5. 2. 1.	Áramkörtervezés	33
5. 2. 2.	A nyomtatott áramkör kialakítása.....	37
5. 3.	Az ütközésselkerülő és vonalkövető panel	37
5. 3. 1.	Áramkörtervezés	38
5. 3. 2.	A nyomtatott áramkör kialakítása.....	40
5. 4.	Az inerciaszenzoros panel.....	41
5. 4. 1.	Áramkörtervezés	41
5. 4. 2.	A nyomtatott áramkör kialakítása.....	42
5. 5.	A motorvezérlő panel	43
5. 5. 1.	Áramkörtervezés	43
5. 5. 2.	A nyomtatott áramkör kialakítása.....	45
6.	Megvalósított funkciók.....	47
6.1.	Vonalkövetési funkció	47
6.1.1.	A C8051F410 mikrovezérlő szerepe a funkció megvalósításában	47
6.1.2.	A C8051F120 mikrovezérlő szerepe a funkció megvalósításában	48
7.	Összegzés	55
7. 1.	Eredmények.....	55
7. 2.	Továbbfejlesztési lehetőségek.....	57
	Irodalomjegyzék	58
	Nyilatkozat.....	59
	Köszönetnyilvánítás.....	59
	Mellékletek	60

Bevezetés

Az elektronika kialakulása óta rohamosan fejlődött. Ez a fejlődés tette lehetővé a szenzorok és aktuátorok megjelenését, melyeket ma is használunk.

A későbbi fejlődés során megjelentek a mikrovezérlők. A mikrovezérlőket egy csipes számítógépeknek is nevezik. A hosszú fejlődési folyamata során, ezek a csipek egyre jobban integrálódtak, egyre több feladat elvégzésére lettek képesek. Ma már mikrovezérlőket számos elektronikai eszközben, és műszerben használunk. Jelen vannak az autóipartól kezdve a számítógép iparon át, a háztartási eszközökig, mai életünk szinte elképzelhetetlen lenne a mikrovezérlők létezése nélkül.

A mikrovezérlők az évek során nagy fejlődésen mentek keresztül, de az 1980-as években megjelent, az Intel gyártotta 8051-es mikrovezérlő család még ma is használatos [13]. Sokrétű felhasználása, és a könnyű C programozása, kompatibilitása miatt fent tudott maradni a piacon. A Silabs cég gyártotta C8051F120 és C8051F410 típusú mikrovezérlők is erre a családra épülnek.

Az elektronika, és mechatronika fokozatos fejlődése lehetővé tette a robotok megjelenését. A robotok felhasználása a mai életben nagyon szerteágazó. Robotok segíthetik az emberek munkáját, bejuthatnak, megvizsgálhatnak olyan területeket is, ahova az ember nem, vagy csak nehézkesen tudna eljutni. A robotok ma már nagyon pontos, precíziós műveletek végrehajtására is képesek, például az autóiparban.

A kezdetleges, és kevés szenzort tartalmazó négy- illetve többkerekű mobilrobotok megjelenése óta, nagy utat tett meg ez a robot típus, hiszen napjainkban már a Marsra is eljuttatták őket, ott adatokat gyűjtve és a Földre továbbítva gyarapítják az emberek ismereteit.

A fentebb felsorolt innovációk segítették az én munkámat is, hogy létrehozhasam a saját robotomat.

Az elmúlt egy évben kezdtem el érdeklődni a mikrovezérlők és a robotika iránt, melynek kombinálásaként választottam ezt a szakdolgozatot. A munkám során, fokozatosan, adott funkcionkénti előrehaladásban, a végső célom egy olyan robot megépítése, mely a hardver adottságai révén számos feladatra használhatóvá és továbbfejleszthetővé válik. További célom egy olyan szoftver írása, mely rávilágít a rendszer összetettségére, a mikrovezérlők sokrétű felhasználására, ennek a kritériumnak felelne meg az útvonalkövető algoritmus szoftveres megoldása is.

1. Irodalmi áttekintés

Ebben a fejezetben szeretném ismertetni azokat az ismereteket, melyek a szenzorok, aktuátorok és mikrovezérlők működésének tesztelése előtt a munkámat segítették.

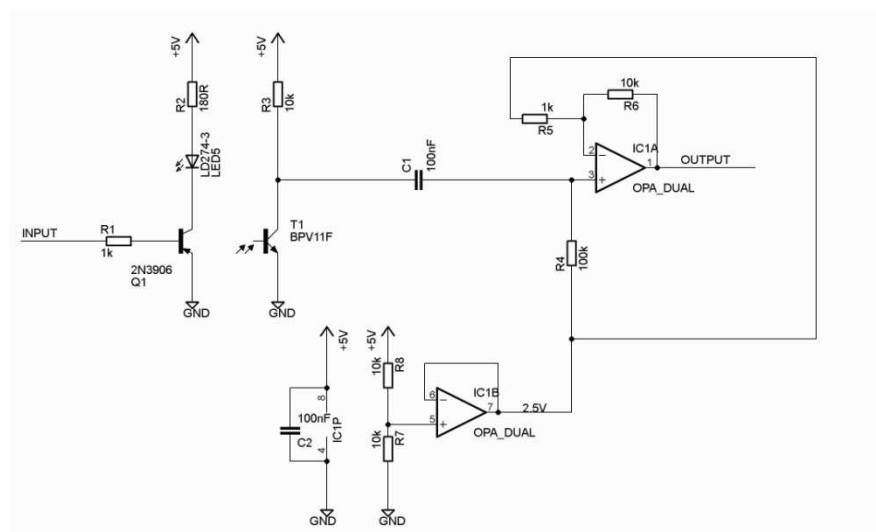
1.1. Szenzorok

A szenzorok, olyan eszközök, melyek a külső, környezetben lévő fizikai mennyiségeket átalakítják elektromos jelekké. Ilyen fizikai mennyiség lehet a szögsebesség, gyorsulás, hőmérséklet, stb. Az így létrejött elektromos jeleket analóg elektronikával jelkondicionáljuk, majd A/D konverterekkel alakítjuk át digitális jelekké.

1.1.1. Ütközésmentesítő szenzor

A robot működése során, az akadályok kikerülésére érdekében elkerülhetetlennek számítanak az ütközésmentesítő szenzorok. A feladat célja az volt, hogy a szenzorral képesek lehessünk 10-15 centiméteres nagyságban megfelelő jelet érzékelni a mikrovezérlő analóg bemenetén, mely kellőképpen olcsón kivitelezhető. Nem volt elvárás az, hogy pontosan meghatározható legyen a tárgy távolsága, ezzel csak azt tudjuk érzékelni, hogy a hatótávolságon belül van-e valamilyen tárgy, avagy nincs. A fotótranzisztor érzékeny a külső fényviszonyokra, ennek a hatásnak a kiküszöbölésére lock-in méréstechnikát érdemes használni.

A szenzor kapcsolási rajza látható az 1.1. ábrán. A kapcsolási rajzhoz az EDAQ530 nevű oktatási mérőműszer fotódetektorként, pletizmográfként használatos szenzorának a kapcsolási rajzát vettem alapul [4].



1.1. ábra: Az ütközésmentesítő szenzor kapcsolási rajza [Forrás: 4]

Az 1.1. ábrán lévő *INPUT* névvel ellátott vezeték jelenti a bemenetet. A bemenetre PWM (impulzus szélesség modulált) jelet küldünk, melynek segítségével a PNP típusú tranzisztor kapcsolgatni fogja az infravörös tartományú fényt kibocsátó LED-et.

A LED közvetlen közelében került elhelyezésre a fotótranzisztor. A fotótranzisztor a külső fizikai mennyiséget, jelen esetben a fényerősséget alakítja át elektromos jellé, mely a fotótranzisztor kollektor lábán jelenik meg.

A fotótranzisztor jelének a továbbítása során megtörténik a DC feszültség leválasztása egy kondenzátor és egy ellenállás segítségével, erre azért van szükség, mert csak analóg tartományban szeretnénk mérni és így a DC feszültség nem fogja zavarni a mérést. Továbbá a fotótranzisztor jelét tízszeresére erősítjük fel két ellenállás és egy műveleti erősítő segítségével.

Két sorosan kapcsolt ellenállás segítségével feszültségosztót alakítunk ki, mely a + 5 V-ból, + 2,5 V feszültséget állít elő. Ezt az átalakított feszültséget használjuk fel a DC leválasztáshoz, mely referenciapontként fog funkcionálni. Ezáltal a kimenő jel a GND és + 5 V feszültségtartományba fog esni, amelyet a compactDAQ (cDAQ) mérőműszer analóg bemenetére kötve mérni tudunk.

Az 1.1. ábrán lévő *OUTPUT*-tal jelzett kimenetet már ráköthetjük a cDAQ mérőműszer analóg portjára.

1.1.2. Vonalkövető szenzorok

A vonalkövető szenzorok, azok a szenzorok, melyekkel a követendő vonalat detektálni tudjuk. Ennek a legegyszerűbb módszere egy infravörös LED-ből és egy fotótranzisztor párosból kialakított szenzor. Ilyen optikai szenzor van beépítve a TCRT5000 típusú eszközbe is [14], ezek kerültek felhasználásra.

Működési elve egyszerű: Ha a feszültséget megkapja a LED folyamatosan kibocsátja az infravörös fényt, melyet a fotótranzisztor érzékelni fog.

1.1.3. Ultrahang alapú távolságszenzor

Az ultrahang alapú távolságszenzor egy olyan szenzor, melynek segítségével távolságot mérhetünk [6]. A szenzor egy kibocsátó (transmit) és egy fogadó (receive) egységből áll. A kibocsátó egység fogja az ultrahangot kibocsátani, míg a fogadó fogja a tárgyról visszavert ultrahangot venni.

A szenzor a robot „szeme” lesz, azért esett erre a szenzorra a választásom, mert megbízhatóan, pontosan és viszonylag gyorsan kapunk adatokat a távolságról. Nem befolyásolja a fényerősség és a hatótávolsága nagynak mondható. Az általam választott szenzor hatótávolsága 3 méter, a látószöge pedig 0° és $\pm 20^\circ$ között jónak mondható.

1.1.4. Odometria optokapu használatával

Az optokapuk működésük alapján két csoportba sorolhatóak. Az egyik a reflexiós, amikor a kibocsátott fény visszaverődését érzékeljük, a másik, pedig a transzmissziós, amikor a kibocsátott fénnel szemben érzékeljük a fényt [5, 15]. A két csoport közül a transzmissziós optokaput valósítottam meg. Az optokapuk olyan optikai szenzorok, melyeknek az egyik végében egy fényt kibocsátó LED, a másik végében (a LED-del szemben) egy fotótranzisztor van. Ezek a szenzorok a robot két hátsó kerekének a tengelye mellett fognak helyet foglalni és a távolságkövetésben, odometriában lesz szerepük.

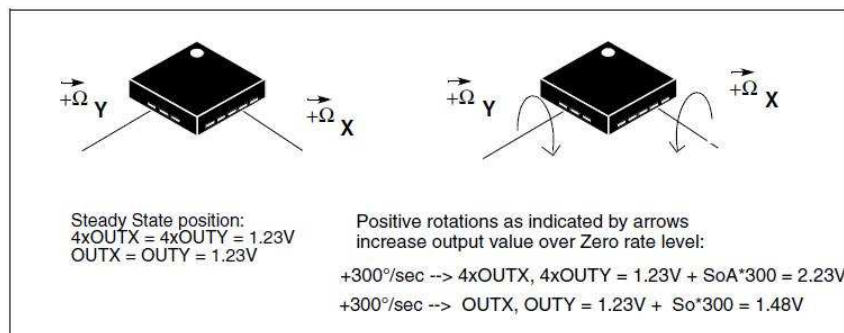
A motorok tengelyeire lesz rászerelve egy olyan, kör alakú kis papír ún. enkóderkerék, mely adott térközönként kivágásokat tartalmaz. Ez a kerék a LED és fotótranzisztor között fog helyet foglalni. Az így kialakított kerék funkciója az, hogy ha az enkóderkerék pont a kapuk között lesz, akkor leárnyékolja a LED és a fotótranzisztor közötti helyet, ezért az optokapu kimenetén alacsony jel lesz. Ellenkező esetben, ha az enkóderkerék kivágott rése lesz az optokapu kapui között, akkor a kimeneten magas állapot lesz. Ennek az elvnek az alapján a szenzorunk digitális kimenettel fog rendelkezni. Ezen módszer alapján a mikrovezérlővel összekötve tudomásunk lesz arról, hogy hányszor volt magas, és hányszor volt alacsony állapotban az optokapu kimenete, így egy használható adatot kapunk a robot mozgásának nagyságával kapcsolatban. A kivágások gyakoriságának növelésével kaphatunk nagyobb felbontást a kerék elfordulásáról.

1.1.5. Mozgás követése inerciaszenzorok segítségével

Inerciaszenzorok közé sorolandóak a gyorsulásszenzorok, és a giroszkópok. Ezekben a szenzorokban egy kisméretű, tehetetlen tömeg van, amelynek a gyorsulás, illetve a szögelfordulás okozta változását mérni tudjuk.

A giroszkóp, olyan eszköz, mellyel adott tengely körüli szögelfordulást lehet detektálni [5]. A robot egyik paneljére két különböző típusú giroszkóp került fel, mivel az érzékelést mind a három tengelyen szeretnénk vizsgálni.

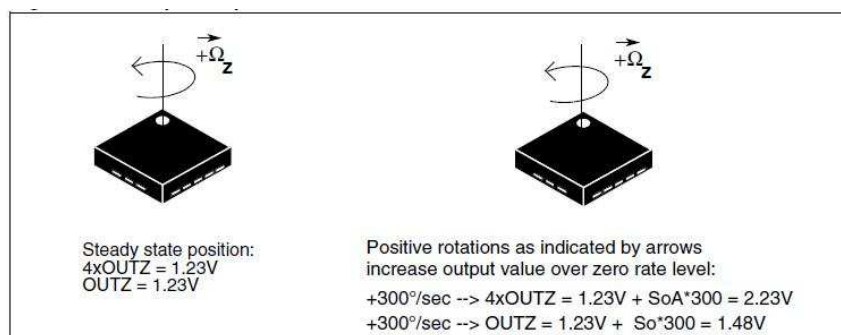
Az LPR530AL típusú giroszkóp, X (pitch) és Y (roll) tengelyen lévő szögsebességet tudja mérni, $\pm 300^\circ/\text{s}$ tartományban. A szenzor analóg kimenettel rendelkezik. A giroszkóp tengelyeit, és kimenetének értékeit mutatja be az 1.2. ábra.



1.2. ábra: Az LPR530AL giroszkóp tengelyei és a kimeneti jel értékei [Forrás: 9]

Az ábráról látható, hogy a szenzor nyugalmi állapotában X és Y tengelyeken is 1,23 V értékű feszültség olvasható. A szenzor kiépítése miatt választhatunk, egy négyszeresen felerősített jel és az erősítetlen jel között, én az utóbbit választottam. Az erősítetlen jel esetében az 1.2. ábrán jelzett So változó a Sensitivity-t jelenti, vagyis az érzékenységet, melynek az értéke $0,83 \text{ mV}/^\circ/\text{s}$.

A másik giroszkóp egy LY530ALH típusú szenzor, mely csak a Z (Yaw) tengely mentén érzékeli a szögelfordulást. Ennek a szenzornak is az érzékelési tartománya $\pm 300^\circ/\text{s}$. Szintén analóg kimenettel rendelkezik, és szintén használhatjuk a négyszeres erősített kimenetet is, de én az erősítetlen jelet használtam fel. Az 1.3. ábra mutatja be a giroszkóp tengelyének az állását és a kimeneti jel feldolgozásához szükséges képletet.



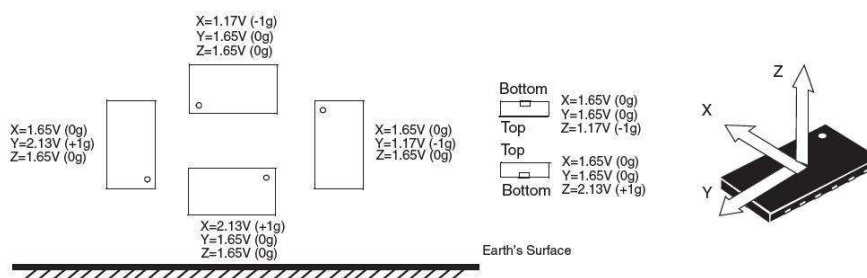
1.3. ábra: Az LY530ALH giroszkóp tengelyének állása és a kimeneti jel értékei [Forrás: 10]

Látható, hogy a két giroszkóp hasonlatos egymáshoz. Alapállapotban ez is 1,23 V feszültségjelet ad a kimenetére. A kimeneti jel értelmezése is ugyanaz, az So változó itt is a Sensitivity-t érti, ami szintén $0,83 \text{ mV}/^\circ/\text{s}$.

A fentebb ismertetett giroszkópok alapján, belátható, hogy a robot mozgásának, kanyarodásának az érzékelése során a Z (Yaw) tengely menti szögelfordulás érzékelése lesz a leghasznosabb.

A gyorsulásmérő szenzorok, olyan eszközök, melyekkel a tárgy Földhöz viszonyított helyzete érzékelhető [6]. Ez azon alapszik, hogy a Föld minden testet a tömegközéppontja felé gyorsít g ($\sim -9,81 \text{ m/s}^2$) gyorsulással.

A LIS302SG gyorsulásszenzor X, Y, Z tengely mentén képes érzékelni a gyorsulást, így a vízszintessel bezárt dőlést is tudjuk vizsgálni. A szenzor $\pm 2 \text{ g}$ gyorsulásig használható és analóg kimenetekkel rendelkezik. A szenzor orientációit és annak a kimenetre okozott hatásait ábrázolja az 1.4. ábra.



1.4. ábra: A LIS302SG gyorsulásszenzor orientációi és annak hatása a kimenetre
[Forrás: 7]

Az ábráról leolvasható, hogy ha az adott tengelyen nem érzékel gyorsulást, akkor a kimeneten 1,65 V feszültség jelenik meg.

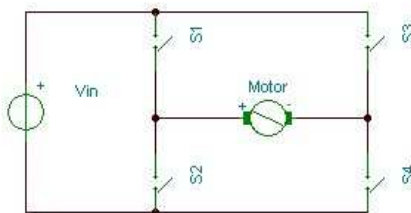
1.2. Aktuátorok

Az aktuátorok olyan eszközök, melyek adott elektromos jeleket képesek átalakítani fizikai mennyiségekké, így képesek beavatkozást generálni a környezetükre. Az aktuátorok legnagyobb csoportját teszik ki a motorok, azon belül foglalnak helyet a DC és a szervo motorok [15]. Ezek ismertetése következik.

1. 2. 1. DC motorok vezérlése

A DC motorok egy pozitív és egy negatív pólussal rendelkeznek. A motorok meghajtására L293DNE típusú integrált áramköröket használtam [11], ennek az ismertetése következik.

Az L293DNE integrált áramkörbe beépítettek 2 darab H-híd kapcsolást. A H-híd kapcsolás ábrája látható az 1.5. ábrán.



1.5. ábra: A H-híd kapcsolás

A H-híd kapcsolás 4 kapcsolóból áll, ezzel tudjuk a motort vezérelni. Értelem szerűen, ha mind a négy kapcsoló nyitott állapotban van, akkor a motor nyugalmi, stop állapotban van. Ha az $S1$ és az $S4$ jelzéssel ellátott kapcsoló zárt állapotba kerül, és az $S2$, $S3$ kapcsoló nyitott állapotban marad, akkor a motor jobbra kezd el forogni. Ha pedig az ellenkezőleg kapcsoljuk a kapcsolókat, tehát az $S2$, $S3$ zárt és az $S1$, $S4$ kapcsoló nyitott állapotba kerül, akkor a motor a bal irányba kezd el forogni.

Tehát az L293DNE IC két beépített H-híd kapcsolást tartalmaz, ezért kettő motor meghajtására képes. Az IC kimeneti portjai diódákkal védettek. Mivel a motorok + 6 V feszültséggel működnek a nagyobb teljesítmény miatt, így képesek 470 mA áramot is felvenni terhelt állapotukban, ezért az IC megfelelő földelése és zavarsszűrése elengedhetetlen.

Egy motor esetében fogom leírni az L293DNE működési elvét, mert a többi motorra ezek ugyanazok. A motor két pólusát az L293DNE IC 1Y és 2Y (Output) lábára kell csatlakoztatni. Ennek a motornak a vezérlése az 1-2EN (Enable) jelzésű lábbal történik. A motort magas állapotú jellel lehet bekapcsolni, és alacsonnyal kikapcsolni, de ez a vezérlési mód nem a legmegfelelőbb. Lehetőség nyílik, hogy a motort a mikrovezérlővel az 1-2EN lábra kiadott PWM jellel hajtsuk meg, ezáltal a motor sebessége állítható, a kiadott impulzus szélességével. A PWM jel a motort ki-/be fogja kapcsolgatni, ezért a motorok kondenzátorral való védelme elengedhetetlen. Az 1A és 2A (Input) jelzésű lábakra kiadható jelek variációit láthatjuk az 1.6. ábrán a táblázatban.

1A lábra adott jel	2A lábra adott jel	Motor állapota
Alacsony	Alacsony	STOP
Alacsony	Magas	HÁTRAMENET
Magas	Alacsony	ELŐREMENET
Magas	Magas	TILTOTT

1.6. ábra: Az 1A és 2A lábakra adott jelek, és annak hatásai a motorra [Forrás: 11]

A többi motor vezérlése ugyanígy működik.

A robot kialakítása miatt, a motorok differenciál meghajtásúak [15]. Ez azt jelenti, hogy a motorok fix helyzetben helyezkednek el, és minden egyes motor, csak a hozzá csatlakoztatott kereket tudja hajtani. Ez által a kanyarodás megoldható úgy, hogy ha a jobb illetve a bal oldali kerekeket ellenkező irányba vezéreljük.

1. 2. 2. Szervo motorok vezérlése

A robot fejének a mozgatására szervo motort használok [15]. Ezek a motorok a számukra kiadott impulzusszélességgel hajthatók. Előnyük, hogy könnyen meghatározott állásba lehet őket vezérelni. Meghajtására 6 V-os feszültséget fogok használni, így a szögsebességük $0,16 \text{ s}/60^\circ$ -ra növekszik. Mindegyik típusnál meg van adva, hogy mekkora, azaz impulzus melyet hiba nélkül még végre tudunk velük hajtani. Ez a tartomány az én esetemben, 0,5 ms (milliszekundum) és 2,5 ms közötti lehet. A szervoknál található egy ún. középpállás pozíció, ha az ennek megfelelő impulzusszélességet adjuk ki, akkor a szervo a középpállásba fog bemozdulni, ez a kipróbált szervonál 1,5 ms szélességet jelent.

1.3. A C8051F120 mikrovezérlő bemutatása

A főmikrovezérlő olyan egység, amely képes, kommunikációt létesíteni egy, vagy több másik mikrovezérlővel, a főbb feladatokat ellátni, a rendszer megfelelő működését koordinálni.

A főmikrovezérlő a Silabs gyártotta C8051F120-as mikrovezérlő lett [2]. A választást segítette, hogy ez a mikrovezérlő rendelkezik minden funkcióval, mely a robot vezérléséhez, továbbfejlesztéséhez szükséges. Van benne két darab ADC (analóg-digitális konverter, 12 és 10 bites), DAC (digitál-analóg konverter), 56 digitális input/output port, PCA (Programmable Counter Array), Timerek, továbbá többféle kommunikációt elősegítő egység (két darab UART, SPI, I2C).

A felhasznált funkciók ismertetése következik.

1.3.1. A mikrovezérlőbe épített ADC leírása

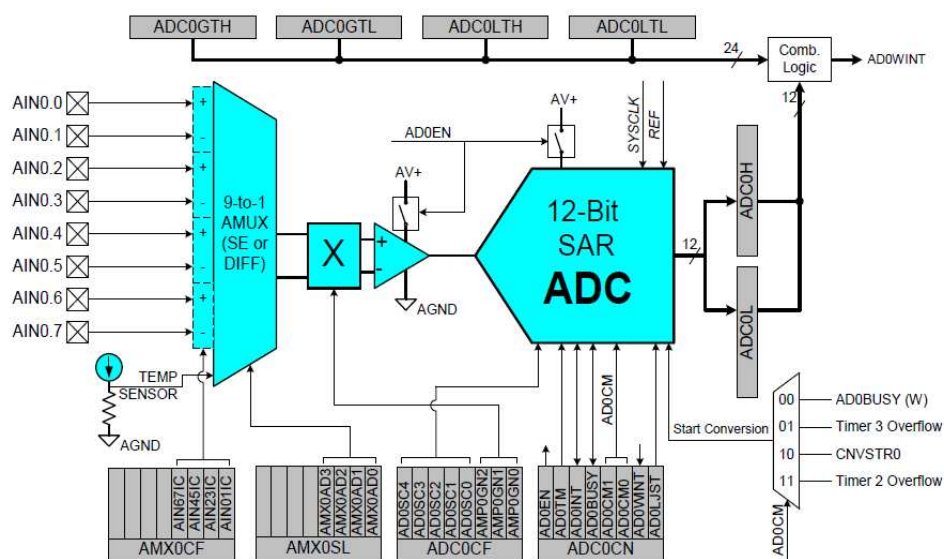
Az analóg digitális konverter egy olyan egység, amely a bejövő folytonos, analóg feszültségjelet alakítja át egy digitális számmá [16]. Az ADC a 0 V (GND) és az előre beállított referenciafeszültség között végzi el a konvertálást. A mérés felbontásától függ, hogy ezt a feszültségtartományt hány részre osztja. Például, ha a konverter 12 bites,

akkor a 0 V és a referenciafeszültség közötti tartományt 0-4095 közötti számokra osztja fel. Az (1.1.) képlet alapján számolható ki a digitális szám az analóg bemeneti feszültségjelből.

$$Z = \frac{U_{be} \cdot 2^b}{U_{ref}} \quad (1.1.)$$

Az (1.1.) képletben szereplő U_{be} , az analóg feszültségnek felel meg, a b pedig a bitek számát jelenti, vagyis az ADC konverzió felbontását. Az U_{ref} az adott referenciafeszültség, amellyel a konverzió megtörténik. A Z értéke, az a digitális szám, mely az ADC átalakítás végeredménye.

A következőekben a C8051F120 mikrovezérlő ADC konverterének a működéséről fogok írni. Ehhez az elvhez hasonlóan működik a C8051F410 mikrovezérlő 12 bites ADC komparátora is, ezért ezt nem ismertetem. Az F120 mikrovezérlőben kettő ADC átalakító áll a rendelkezésünkre, egy 12 bites és egy 8 bites. A nagyobb felbontás érdekében én a 12 bites ADC-t használtam. Az ADC blokkvázlata látható az 1.7. ábrán.



1.7. ábra: A C8051F120 mikrovezérlő 12 bit-es ADC blokkvázlata [Forrás: 2]

Az ADC0 esetében az ADC konverzióhoz szükséges referenciafeszültséget a VREF0, vagy a DAC0 (digitális-analóg átalakító kimeneti) portjáról kaphatja. Az F120 mikrovezérlő rendelkezik belső referenciafeszültséggel is, melynek az értéke 1,2 V, amely egy kétszeres erősítésen megy keresztül, így 2,4 V feszültség jelenhet meg a

VREF porton, amit a tervezéskor összekötöttem a VREF0 porttal. A továbbiakban ezt a referenciafeszültséget fogom használni.

Az 1.7. ábrán jól látható, hogy az ADC-nek 8 bemeneti csatornája van, és egy hőmérsékletérzékelő bemenete. Az ADC működése az AD0EN bittel kapcsolható be, illetve ki, ez a bit az ADC0CN regiszterben foglal helyet. A bemenetek közül egy multiplexer segítségével tudjuk kiválasztani az aktuálisan mérendő csatornát, az AMX0SL regiszterrel tudunk a vizsgálandó csatornák közül váltani, míg az AMX0CF regiszterrel a konfigurációk közül választhatunk. A vizsgálandó csatorna kiválasztása után, a jelet felerősíthetjük egy erősítő segítségével, melyet az ADC0CF regiszter AMP0GN2-0 bitek segítségével állíthatunk be. Az erősítés mértéke lehet 0,5, 1, 2, 4, 8, 16. Az AD0SC4-0 bitekkel az ADC órajele állítható, melynek a frekvenciája kisebb, vagy egyenlőnek kell lennie, mint 2,5 MHz.

Az ADC konverzió maximális sebessége 100 ksps (ezer minta per másodperc, kilo samples per second). A konverziót négy féle módon kezdhethetjük meg, ezeket lehet az AD0CM1-0 bitekkel beállítani. A négy féle mód: magas állapotba állítani az AD0BUSY bitet, vagy a Timer2 számláló túlcsordulásakor, vagy a Timer3 számláló túlcsordulásakor, vagy az előre beállított CNVSTR0 porton lévő felfutó élre. Mikor a konverzió véget ért, akkor az AD0BUSY flag alacsony állapotba kerül, és beállítja az AD0INT bitet, ezzel jelezve, hogy az átalakítás véget ért. A kapott adat megjelenik az ADC0H (felső bitek) és ADC0L (alsó bitek) regiszterekben, melyek lehetnek jobbra és balra igazítottak is, melyet az AD0LJST bittel állíthatunk be.

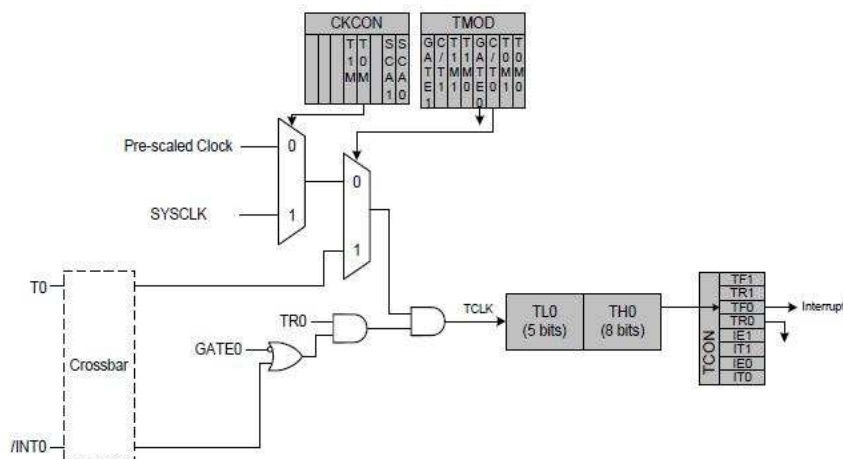
Továbbá beállíthatóak még különféle ablakozási módok is, melyek nem kerülnek ismertetésre.

1.3.2. A mikrovezérlő időzítő áramkörei

A timer működése egyszerűen úgy fogalmazható meg, hogy az előre beállított órajellel számol felfelé egyesével, majd ha már ez a számolt szám nem fér el a számláló regisztereiben, akkor túlcsordul és kezdi előlről a számolást. A timer-ek segítségével lehetővé válik a pontos időzítés, jelalakok szélességének a megmérése is.

A C8051F120-as mikrovezérlőnek 4 timere (számlálója) van. Ezek közül a Timer0-át és annak is a 16 bites módját fogom bemutatni.

Az 1.8. ábra mutatja a Timer0 blokkvázlatát. Az ábra a 13 bites számlálóra vonatkozik, de ezzel megegyezően működik a 16 bites számláló is. A különbség abban nyilvánul meg, hogy a TL0 és a TH0 regiszterek 8-8 bitet képesek tárolni.



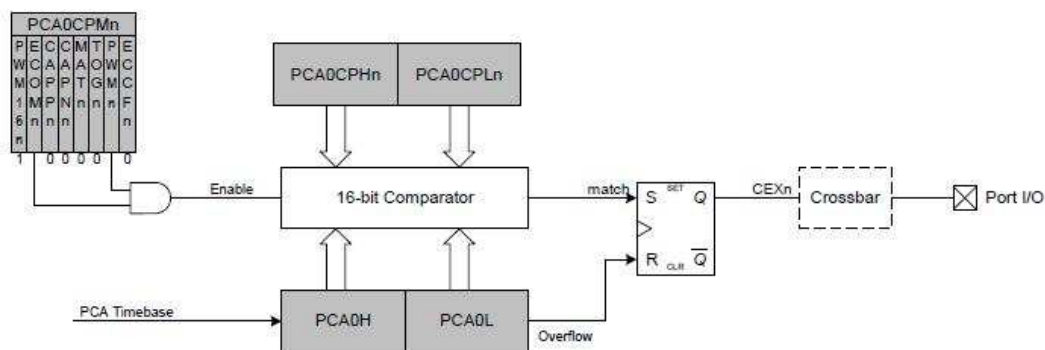
1.8. ábra: A Timer0 blokkvázlata [Forrás: 2]

Az 1.8. ábrán látható, hogy a Timer0 beállításait a CKCON, TMOD és a TCON regiszterekkel végezhetjük el. A CKCON regiszterben lehet beállítani többek között a Timer0 órajelét, a példa során ez a systemclock (24,5 MHz) 12-ed része volt. A TMOD regiszterben lehet beállítani a Timer módját. A TCON regiszterben a számláló engedélyezése és a megszakítások állíthatók be.

1.3.3. Programozható számlálótömbök (PCA) kezelése

Mikrovezérlőn a PCA (Programmable Counter Array) egység segítségével állíthatunk elő PWM jelet. A PWM jel, más néven, impulzusszélesség modulált jel, a nevében szereplő impulzus szélességétől függően más és más hatást érhetünk el.

Az impulzusszélesség moduláció, más néven a PWM módnak a 16 bites változatát fogom bemutatni. A PCA lényegében felfogható egy számlálónak (Timer), mely a beállított órajel-ciklusonként növeli az értékét. A 16 bites PWM mód működési ábrája látható az 1.9. ábrán.



1.9. ábra: A 16 bites PWM működése [Forrás: 2]

Az ábráról leolvasható, hogy PCA0CPLn (LSB) és a PCA0CPHn (MSB) regisztereknek adhatunk értéket, így az alsó és felső 8 bit összegzésével elérhető a 16 bites vezérlés. Figyelnünk kell arra, hogy elsőnek a PCA0CPLn regiszter tartalmát írjuk, és aztán csak a felső 8 bitet tartalmazó regisztert. A PCA számolni fog 0-tól 65535-ig (ez a 16 bites tartománynak felel meg) az előre beállított órajel-ciklussal. A 16-bites komparátor segítségével, ezt a számolt értéket mindig összehasonlítja a PCA0CPLn és a PCA0CPHn regiszterekben található két 8 bites szám összegével. Ha a számolt érték elérte a regiszterekben tárolt értéket, akkor az előre beállított digitális kimeneti porton magas (1-es) érték lesz, és mindaddig ez az érték marad, míg a számláló túl nem csordul, azaz 65535-ig. Minden más esetben a kimeneten alacsony (0-ás) érték lesz. Ezek ismeretében az impulzusszélességet könnyen változtathatjuk.

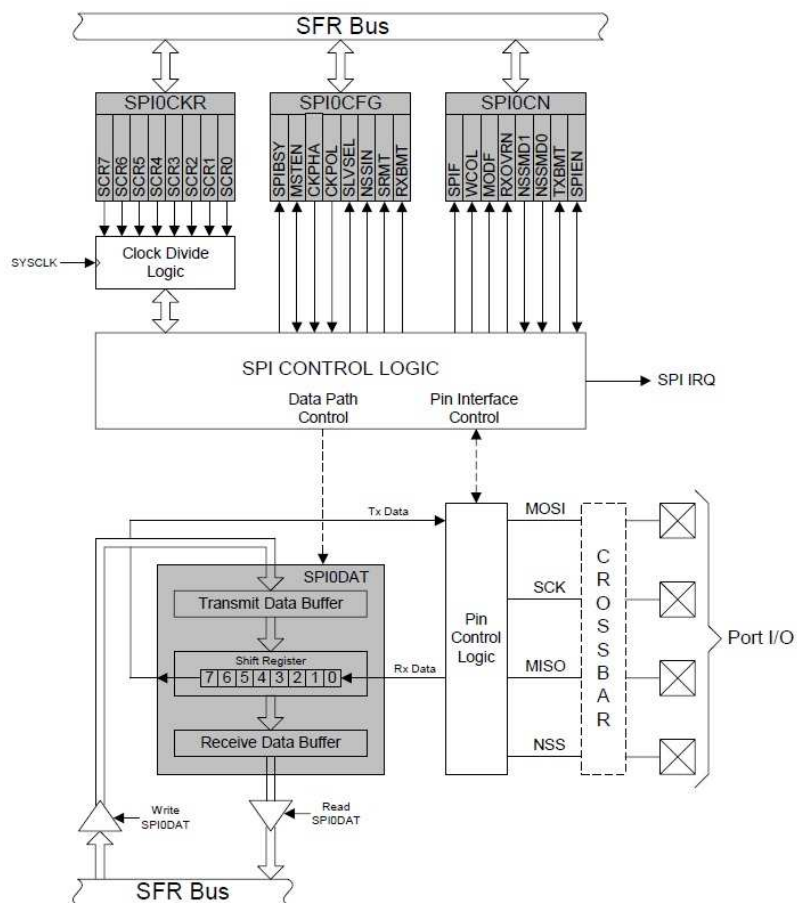
1.3.4. Kommunikáció az SPI interfészen keresztül

A két mikrovezérlő (C8051F410 és C8051F120) közötti kommunikáció megoldásának számos módja áll rendelkezésre, ebből én az SPI interfészen keresztüli kommunikációt valósítottam meg, ezt fogom bemutatni.

Az SPI0 (Serials Peripheral Interface), egy teljes duplex (oda-vissza irányuló adatátvitelt biztosító) szinkron soros busz. Az SPI0 interfész kezelni tudja a master és a slave eszközöket három, ill. négy vezetéken, továbbá lehetőség van több eszköz kezelésére is.

Az én esetemben egy master és egy slave eszköz van, ezek sorban a C8051F410 és a C8051F120 mikrovezérlők lesznek. Azért a C8051F410 mikrovezérlő lett a master egység, mert ő fogja kezdeményezni az adatok elküldését és lényegesen többször használjuk adatküldésre, mint az F120 SPI0 interfészét.

Az 1.10. ábrán látható az SPI0 blokk diagramja.



1.10. ábra: A C8051F120 SPI0 blokk diagramja [Forrás: 2]

Mivel én a négy vezetékes kommunikációt választottam, ezért az 1.10. ábrán jelzett MOSI, SCK, MISO, NSS port mindegyikére szükség lesz. Ezek rövid bemutatása következik.

MOSI (Master Out, Slave In): A MOSI port a master eszköz kimenetét és a slave eszköz bemenetét köti össze. Ez a port szolgál a master és a slave eszköz közötti soros adatátvitelre. A legnagyobb helyiértékű bit kerül elsőnek elküldésre, más néven ez a most-significant bit (MSB).

MISO (Master In, Slave Out): A MISO port a master eszköz bemenetét és a slave eszköz kimenetét köti össze. Ez a port teszi lehetővé, hogy a slave eszköz tudjon adatot továbbítani a master eszköznek. Itt is elsőnek a legnagyobb helyiértékű bit kerül elküldésre.

SCK (Serial Clock): Az SCK porton keresztül továbbítódik az órajelel, ez a port a master eszköz kimenete és a slave eszköz bemenete. Ez a port használatos az adatátvitel szinkronizálására a MOSI és a MISO portok között. Az SPI0 interfész generálja ezt az órajelet, amikor az eszköz masterként van konfigurálva. A slave eszköz ezt az órajelet

figyelman kívül hagyja, amikor nincs kiválasztva ($NSS=1$) a négy vezetékes megvalósításban.

NSS (Slave Select): Ez a port a slave eszközök kiválasztására szolgál. Három lehetséges beállítás érhető el az NSSMD1 és az NSSMD0 bitek helyes beállításával, amelyek az SPI0CN regiszterben foglalnak helyet. Ha a két bit nulla, akkor három vezetékes üzemmódba kerülünk, tehát az NSS portra nincs szükségünk. Ha az NSSMD1 bit 0, és az NSSMD0 bit 1, akkor négy vezetékes üzemmódban vagyunk és az NSS port a slave eszközön bemenetként van konfigurálva. A master eszközön, az NSS porton lévő jel magas állapotból alacsony állapotba kerülése megszünteti a master eszköz funkcióit, így több master eszköz használható egy SPI buszon keresztül (multi-master mode). Ha az NSSMD1 bit 1 és az NSSMD0 bit 1 vagy 0, akkor az NSS port kimenetként van definiálva. Az NSSMD0 bittel tudjuk konfigurálni, hogy milyen jel jelenjen meg az NSS porton. Ez a beállítás csak akkor alkalmazható, ha az SPI0 interfész master eszközként működik.

A továbbiakban szeretném ismertetni a master és slave módok működését.

- **Master mód működése**

A master eszköz kezdeményezi az összes adatátvitelt az SPI buszon keresztül. A Master engedélyező flag-gel (MSTEN bit a SPI0CN regiszterben) tudjuk engedélyezni a master módot. Master módban, ha egy bájtnyi adatot írunk az SPI0 adatregiszterébe (SPI0DAT), akkor ez az adat az adó pufferbe kerül. Ha az SPI shift (léptető) regisztere üres, akkor az adó pufferből a shift regiszterbe kerül az adat, és az átvitel megkezdődik. A master eszköz azonnal megkezd az adat shiftelését (léptetését) a slave eszköz felé a MOSI porton keresztül, feltéve, ha van órajel az SCK porton. Az SPIF bit az SPI0CN regiszterben, akkor állítódik be magas állapotba, ha az adat elküldődött, ezzel megszakítást is vezérelhetünk. Ezzel a bittel tudjuk vizsgálni még azt is, hogy mikor áll készen a master eszköz az olvasásra. Az adatküldés során a legnagyobb helyiértéken lévő bit kerül elsőnek a shift regiszterbe. Ha az adatot teljesen átléptettük a regiszterbe, akkor a fogadó pufferbe kerül, ahol az SPI0DAT regiszter kiolvasásával fogjuk megkapni. A master üzemmódnak három lehetséges változata van, a fentebb említett NSSMD1 és NSSMD0 bitek beállításával.

- **Slave mód működése**

Ha az SPI0 engedélyezve van, és nem master módban van, akkor az eszköz slave módban lesz. A slave eszközt, a master eszköz órajele időzíti. Az SPI0 bit számlálója az SCK (Serial Clock) port éleit számolja meg. Ha egy bájt átléptetődött a shift regiszteren keresztül, akkor a SPIF bit beállítódik 1-re, és az adat a fogadó regiszterbe kerül. Az adat kiolvasható az SPI0DAT regiszter kiolvasásával. A slave eszköz nem kezdeményezhet adatküldést. Tehát elsőnek a master eszközzel kell adatot küldenünk, majd a slave eszközzel kiolvassuk, ez után az adat regiszterbe (SPI0DAT) beleírhatjuk az elküldeni kívánt adatot a slave eszközről a master eszköznek. Majd a master eszközön ez után kiolvashatjuk az adat regiszter tartalmát, ami a slave eszközről küldött adat lesz.

A slave módnak két változata van, a három, ill. a négy vezetékes.

Az alapbeállítás a négy vezetéket igénylő mód, ami akkor aktív, ha az NSSMD1 bit nulla, míg a NSSMD0 bit egyes, az SPI0CN regiszterben. Ekkor az NSS port bemenetként van definiálva. Amikor az NSS port állapota alacsony, akkor az SPI0 engedélyezve van, ha az NSS magas állapotba kerül, akkor az SPI0 kikapcsolásra kerül. A bitszámláló az NSS porton jelentkező lefutó élre resetelődik.

A három vezetékes mód, akkor aktív, ha az NSSMD1 bit és az NSSMD0 bit is nulla. Ebben a módban az NSS port nem kerül használatba. Fontos megjegyezni, hogy ez a mód csak akkor használható, ha csak egy slave eszközzel szeretnénk kommunikálni. A három vezetékes módban nincs lehetőség a bitszámláló resetelésére (nullázására) az NSS porton keresztül, amivel detektálhatjuk egy bájtnyi adat érkezését. Az egyetlen módja a bitszámláló kinullázására az, hogy ha az SPI0-át letiltjuk, majd újra engedélyezzük az SPIEN (SPI Enable) bit segítségével.

- **Az SPI órajelének beállítása**

Az SPI0CFG (SPI0 Configuration Register) CKPHA (SPI0 Clock Phase) és CKPOL (SPI0 Clock Priority) bitek négy féle módot biztosítanak az órajel fázisának és polaritásának helyes beállítására. CKPHA bittel állíthatjuk az SPI0 fázisát, ha 0, akkor az adat az órajel periódusának az első élére lesz központosítva, míg ha a bit 1-es, akkor a második élre lesz központosítva. A CKPOL bittel az SPI0 órajel polaritását tudjuk állítani, ha 0, akkor az órajel alacsony állapotban van a tétlen állapotokban, míg ha a bit 1-es, akkor magas állapotban lesz az órajel, ha nincs szükség rá. Fontos megjegyezni

azt, hogy az SPI0 órajelének a fázisa és a polaritása a master és slave eszközön meg kell, hogy egyezzen.

Az SPI0 órajelét beállíthatjuk az SPI0CKR (SPI0 Clock Rate Register) regiszter segítségével. Mivel a slave eszköz, a master eszköztől kapja az órajelet, ezért ez a regiszter slave módban figyelmen kívül hagyható. A Master eszközön maximálisan beállítható adatátviteli sebesség (bits/sec) az órajel felével egyezik meg, vagy a 12,5 MHz-cel. Ha csak a master eszközről szeretnénk adatot továbbítani, akkor a slave eszköz az adatot az órajel egy negyedével tudja olvasni. Ha oda vissza szeretnénk adatot küldeni (full-duplex), akkor a maximális sebesség (bits/sec) a master eszköz maximálisan beállítható órajel frekvenciájának az egy tizede.

1.4. A C8051F410 mikrovezérlő bemutatása

Ennek a mikrovezérlőnek a fő célja az ütközésmentesítő és vonalkövető szenzorok jeleinek átalakítása, feldolgozása, és a kapcsolat megteremtése a főmikrovezérlővel. Ezt a működést látja el a Silabs gyártású C8051F410-es mikrovezérlő [3]. Az F410 rendelkezik PCA egységgel, mellyel PWM (impulzus szélesség modulált) jelet adhatunk ki, továbbá 12 bites ADC konverterrel, és SPI kommunikációval. Ezeket a funkcionális egységeket nem ismertetném, mert a működési elvük hasonló a C8051F120-as mikrovezérlő megfelelő egységeivel, melyek ismertetésre kerültek.

2. Szenzorok tesztelése

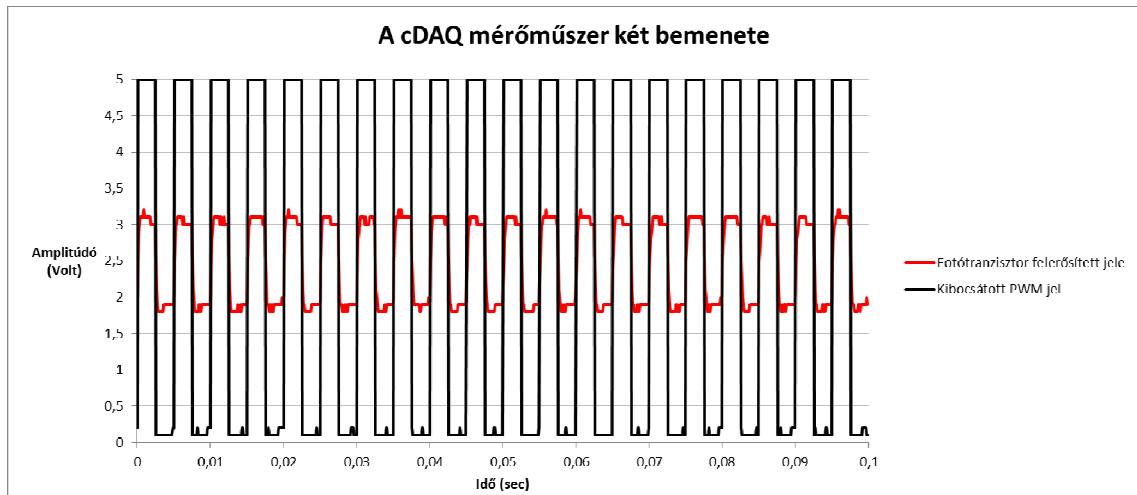
A robot szenzorjainak a felsorolása: ütközésmentesítő szenzorok, vonalkövető szenzorok, ultrahang alapú távolságszenzor, optokapuk, giroszkópok, gyorsulásmérő szenzorok, ezek tesztelési bemutatása következik.

2.1. Ütközésmentesítő szenzor tesztelése

A tesztelés során CompactDAQ (továbbiakban: cDAQ) műszert használtam, és a hozzá tartozó LabVIEW példaprogramot kis mértékben átalakítva. A cDAQ előnye, hogy képes többféle jelet kibocsátani, és közben több bemenetén mérni is egy időben, ezt a tulajdonságát használtam ki én is. A műszerrel kibocsátottam egy négyszögjel sorozatot (PWM jelet) az 1.1. ábrán lévő *INPUT* jelzésre, majd ezt a kibocsátott jelet visszavezettem a cDAQ bemenetére, és továbbiakban ezt a jelet használtam referenciajelként. Az 1.1. ábra kapcsolási rajzán *OUTPUT*-tal jelzett kimenetet pedig rákötöttem a műszer analóg bemenetére, ez a már felerősített jele a fotótranzisztornak.

A programban a zajszűrés érdekében, ún. lock-in méréstechnikát alkalmaztam. Ez a technika felfogható egy keskeny sávú sáváteresztő szűrőként is. A lényege az, hogy a kapott jelet egy referenciajellel megszorozva majd időben átlagolva a kimeneten egy szűrt, kevésbé zajos jelet kapunk, ez a jel lesz a mérés végeredménye. A program használatakor a referenciajel az a négyszögjel volt, a -1 és 1 közötti tartományba beleszorítva, melyet az 1.1. ábra *INPUT* jelzésre bocsátottam ki. A zajos jel pedig az *OUTPUT*-tal jelzett helyről jövő jel volt. Ezeket összeszorozva, majd átlagolva megkaptam a lock-in szűrés utáni jelet.

A 2.1. ábrán jól látható, hogyan alakulnak a lock-in méréstechnika alkalmazása előtti jelalakok, ha a tárgy 5 centiméter közelében van. Az ábrán a fekete színű jel a kibocsátott négyszögjel, más néven a referenciajel. A piros színű jelzést kaptam a fotótranzisztor felerősített jeleként.



2. 1.ábra: A cDAQ mérőműszer két bemeneti jel

Nagyobb távolságok tesztelésekor azt figyeltem meg, hogy minél nagyobb a távolság, annál kevesebb a zaj hatása és a távolságok növekedésével az érzékelhető jel amplitúdója csökken. Az első tesztelések TCRT1000 típusú egybeépített, LED és fotótranszisztor párost tartalmazó szenzorral történt, de a mérések során ezzel nem tudtam kellő távolságokat érzékelni, ezért a végleges szenzor egy LD274-3 infravörös LED és egy BPV11F fotótranszisztor párosból állt. Ezzel a kettőssel érzékelhetővé vált a körülbelüli 15 centiméteres távolság is, ami már megfelelő az ütközések elkerülésére.

A szenzor hátrányaihoz sorolandó, hogy a külső fényhatás zavarja a mérendő jelet, ezért a panelre szereléskor megfelelő árnyékolás szükséges. A panel tesztelése során zsugorcsöves árnyékolást alkalmazva a vizsgálandó jel nagyon kicsinek bizonyult, ezért az árnyékolás nem volt megoldható.

A szenzort tesztelése közben láthatjuk a 2.2. ábrán.



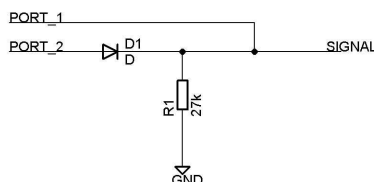
2. 2.ábra: Az ütközésmentesítő szenzor tesztelése közben

2. 2.Vonalkövető szenzorok tesztelése

A tesztelés során fekete vonalat használtam az útvonalhoz, így minél jobban érzékelt a fotótranzisztor az alatta lévő vonalat, annál kisebb feszültség jelent meg a kimeneten. A fotótranzisztor kimeneti jelét a mikrovezérlő analóg-digitális konverterével átalakítva felhasználható jelet kapunk. A szenzor bekötési rajza az Elülső panelnél, az 5.3. fejezetben kerül ismertetésre.

2. 3.Ultrahang alapú távolságszenzor tesztelése

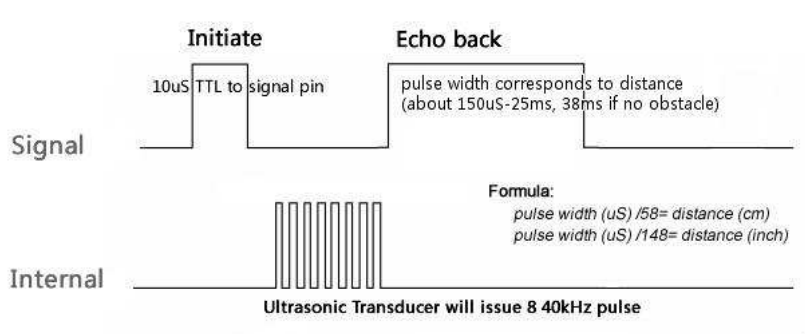
A szenzoromnak a kialakítása olyan, hogy csak négy lábat vezettek ki a tűkesorra, ezek sorrendben: GND, Vcc, NC (Not Connected), SIG (Signal). A szenzor működtetése + 5 V-tal oldható meg. A lábak eloszlása alapján megfigyelhető, hogy csak egy láb van a kibocsátandó jel küldésére és a visszavert jel fogadására. Ezért arra a kimeneti portra, mellyel kibocsátjuk a megfelelő hosszúságú jelet, arra egy egyutas egyenirányító kapcsolást kell raknunk, így nem kell megváltoztatni a PORT_2 láb irányát a jel fogadásához. A PORT_1 lábon érzékeljük a visszavert jelet. Ennek a kapcsolási rajza látható a 2.3. ábrán.



2. 3.ábra: Az egyutas egyenirányítás kapcsolási rajza

Az ábrán lévő *SIGNAL* kimenet csatlakoztatható az ultrahangos szenzor SIG nevű portjára.

A szenzor működési elve látható a 2.4. ábrán.



2. 4.ábra: Az ultrahangos szenzor működési elve [Forrás: 17]

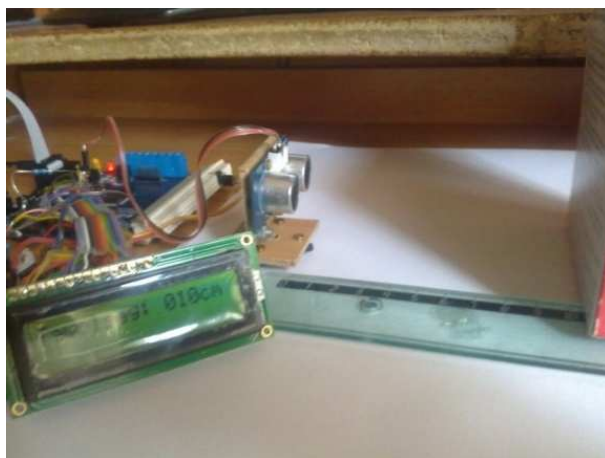
A 2.4. ábrán látható, hogy a szenzor megfelelő működése érdekében a 2.3. ábrán lévő *PORT_2* bemenetre kell kiadnunk a mikrovezérlővel a 10 μ s-os trigger jelet. A jel kiadása után a szenzor ezt 8 darab 40 kHz-es ultrahang impulzussal veszi tudomásul, majd ez után várunk a visszavert jelre. Ekkor a SIG port magas állapotba kerül és addig abban is marad, míg a jel vissza nem verődik a fogadó egységbe. A jel szélessége alapján kiszámolhatjuk a szenzor és a tárgy közötti távolságot.

A visszavert jel szélességének a megmérésére a C8051F120-as mikrovezérlőnek a Timer0 számlálóját használtam. A beállítások lehetőségét lentebb, a 4.1.1. fejezetben ismertetem. A timer által szolgáltatott TL0 és TH0 regiszterek tartalmát egyesítem, így kapok egy 16 bites számot. Mivel a timer a mikrovezérlő órajel-ciklusa a 24,5 MHz 12-ed részével számol egyesével, míg el nem éri a 65535-öt, ezért az egyesített számláló értékét meg kell szoroznunk 0,49 μ s-mal, mert ez felel meg egy művelet elvégzéséhez szükséges időnek. A hang terjedési sebessége szobahőmérsékleten 340 m/s, ezt átváltva megkaphatjuk, hogy 1 μ s alatt 0,034 cm-ert tesz meg. A szenzor működése olyan, hogy a visszavert jelben az az időintervallum is beleesik, míg a jel eljut a tárgyhöz, majd a tárgyról visszajut a fogadó egységbe, így a 0,034 cm-t el kell osztanunk kettővel. A (2.1.)-es képlet tartalmazza a távolság kiszámolását.

$$távolsag_{cm} = timer_{ertek} \cdot 0,49 \mu s \cdot 0,017 cm \quad (2.1.)$$

Továbbá az is látható a 2.4. ábrán, hogy ha a visszavert jel hosszúsága 38 ms, vagy több, akkor a tárgy a 3 méteres hatótávolságon kívül helyezkedik el.

A távolság kiíratására LCD modult használtam. A szenzort tesztelés közben láthatjuk a 2.5. ábrán.



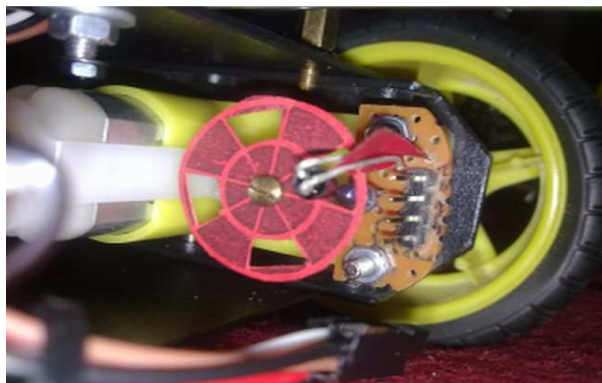
2. 5.ábra: Az ultrahangos szenzor tesztelése

2. 4. Optokapu tesztelése

Az optokapuk tesztelése során derült ki, hogy ezek gyártási méretük miatt nem építhetők a robotra, ezért egy saját optokapu került kialakításra egy infra LED és fotótranzisztor egymással szembefordított párosából, ezt láthatjuk a 2.6. ábrán.

A kerék tengelyére van rászerezve az enkóderkerék [15], így a kerék meghajtásával egy időben az enkóderkerék is forogni fog. Ezáltal a kerék elfordulásának a mértékéről kaphatunk adatot, amelynek az odometriában, távolságkövetésben lesz szerepe. A szenzor kimenetén digitális jel áll elő, melyet a mikrovezérlővel feldolgozhatunk.

A 2.6. ábrán látható, hogy a felhasznált enkóderkeréknek 4 kivágott rése és 4 nem kivágott rése van. Az így kialakított enkóderkerék egy körülfordulása során 4 impulzus keletkezik a szenzor kimenetén.



2. 6.ábra: Az optokapu beszerelt állapotában

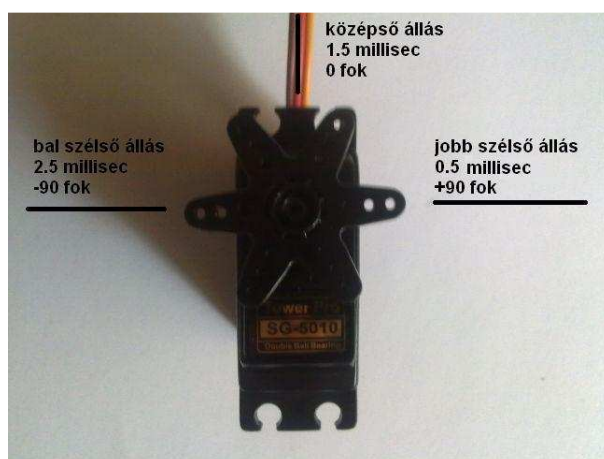
3. Aktuátorok tesztelése

Az aktuátorok közé tartoznak a DC motorok és a szervo motor. Ezek tesztelése során szerzett információk lejegyzése következik.

Mivel a motorok tesztelése során kapott információk teljesen megegyeznek a 1.2.1. fejezetben leírt elmélettel, ezért a motorok tesztelési ismertetésére nem kerül sor.

3.1. Szervo motor tesztelése

A tesztelés során 16 bites PWM módot használtam. A szervo vezérlését úgy szerettem volna megoldani, hogy a szervo beállításának a helyzetét meg lehessen adni fokban, előjelhelyesen. A 3.1. ábra tartalmazza a szervo és annak főbb állásait és a neki megfelelő értékeket.



3. 1. ábra: A szervo és főbb állásainak értékei

A 3.1. ábrán látható, hogy a középállást a 0°-nak, azaz a 1,5 ms-os impulzusszélességnek feleltettem meg. A - 90°, a 2,5 ms-os és a 90°, a 0,5 ms-os szélesség megfelelője.

A (3.1) képlet alapján kiszámítható a kívánt fok eléréséhez szükséges idő.

$$ido = \frac{fok \cdot 1000}{90} [\mu s] \quad (3.1.)$$

Mivel a jobb és a bal szélsőhelyzet közötti időbeli különbség 1000 μs és a szög, melyet foroghatunk maximum $\pm 90^\circ$, így az *ido* értéke, azaz idő lehet mely a kellő fok eléréséhez szükséges eltelt idő μs -ban. Ez az érték lehet pozitív, ill. negatív is, ez a fordulási irányt határozza meg.

A (3. 2.) képlet alapján kiszámítható a kívánt impulzusszélesség.

$$szelesseg_{impulzus} = 65535 - \left(\frac{1500}{0,49} - \frac{ido * 65535}{32125} \right) \quad (3.2.)$$

A (3.2.)-es képletben szereplő *ido* változó a (3.1.)-es képletben kiszámolható *ido* változóval egyezik meg. A zárójelben lévő tag első törtjének az értéke a középálláshoz (0°-hoz) tartozó érték. A zárójel második törtjében szereplő számok hányadosa az 1000 µs-os kimeneti magas jel értékének megfelelő szám. Mivel csak a középállástól számítva pozitív, ill. negatív irányban térhetünk el 1000 µs-es értékkel, ezért van erre szükség. Az *ido* változó értékével tudjuk skálázni a kivonandó értékét. A 16 bites PWM mód fordított működésére lesz szükségünk, ezért az egész kiszámolt értéket ki kell vonnunk a 65535-ből, mert azt szeretnénk, hogy a kimenet akkor legyen magas állapotban, amikor a számláló elérte a képlet alapján kiszámolt értéket és a számláló túlcsordulásáig ebben az állapotban is maradjon. Az így kiszámolt értéket még át kell alakítani bináris számmá (kettes számrendszer), majd szét kell bontani alsó és felső 8 bitre, és így értékül adható a PCA0CPLn és a PCA0CPHn regisztereknek.

4. Mikrovezérlők és funkcióik

A mikrovezérlők kiválasztásának az első kérdése az volt, hogy a robot működésére elég-e egy mikrovezérlő, avagy sem. Az ütközésmentesítő szenzorok, és a vonalkövető szenzorok egy panelen helyezkednek el és sok analóg portot foglalnak el, ill. ezek vizsgálata lelassította volna a főmikrovezérlőt. Továbbá elég sok vezeték kellett volna a panelek összeköttetésének a megvalósításához, ezért két mikrovezérlőt választottam ki a rendszer helyes működtetéséhez, így a főmikrovezérlőt nem terheli a sok ADC konvertálás, és elég csak négy vezetékkel összekötnünk a két panelt a kommunikációhoz. Továbbá így kapunk két univerzális panelt, amelyeket más rendszerekbe is beépíthetünk és külön-külön is működőképesek lesznek.

Most a mikrovezérlők felhasznált funkcióik kerülnek bemutatásra, melyek eltérnek az 1.3. fejezetekben leírt elméletektől.

4. 1.C8051F120 mikrovezérlő

A specifikus feladatokat ez a mikrovezérlő fogja ellátni, ezért ennek az ismertetésére kerül sor.

4. 1. 1. Timer konfigurálása

Egy adott példán keresztül fogom bemutatni a timer működését. A példa az ultrahang alapú távolságszenzor működése során a visszavert jel szélességének a megmérésére vált szükségessé. Az elméleti rész az 1.3.2. fejezetben került leírásra.

A TMOD regiszterben beállítjuk a Timer0 módját, a 16 bites módot, majd a GATE0 engedélyező bittel beállíthatjuk, hogy csak akkor számoljon a timer, ha engedélyezve van, és ha az /INT0 magas állapotban van. A port I/O-nál érdemes az /INT0-át arra a portra helyezni, ahol azt szeretnénk, hogy a timer vizsgálja ezt a bemenetet. A TCON regiszterrel tudjuk engedélyezni a számlálót. Az így beállított számláló akkor kezd el számolni, amikor az /INT0-ra beállított port magas állapotba kerül, és addig számol, míg újra alacsony állapotba nem kerül. Az így kiszámolt értéket a TL0 és a TH0 regiszterekbe helyezi, melyet, ha egyesítünk, akkor megkapjuk a 16 bites számot. Jelen esetben, mivel a timer órajele 2,04167 MHz, így a timer értékét meg kell szoroznunk 0,49 μ s-mal, az így kapott érték lesz annak az időintervallumnak az értéke μ s-ban, melyet a jel magas állapotban tölt el.

4. 1. 2. A PCA konfigurálása

A mikrovezérlő PCA egységének működésének elvét az 1.3.3. fejezetben ismertettem.

Egy példán keresztül bemutatom, hogy hogyan lehetséges kiszámítani a regiszterek tartalmát, ha a kívánt impulzusszélességet ismerjük, és a PCA órajelét is. Legyen a mikrovezérlő órajele 24,5 MHz, ezt a PCA beállításainál leosztjuk 12-vel, így a PCA órajele 2,04167 MHz lesz. Ha ennek vesszük a reciprokját, akkor megkapjuk, hogy egy művelethez 0,4898 μ s szükséges, melyet kerekítettem 0,49 μ s-ra. Tehát ha azt szeretnénk, hogy az adott kimeneti porton 1000 μ s-ig legyen magas a jel, a periódusidő további részében pedig alacsony, akkor a képlet (4.1.) az alábbi alakban írható fel:

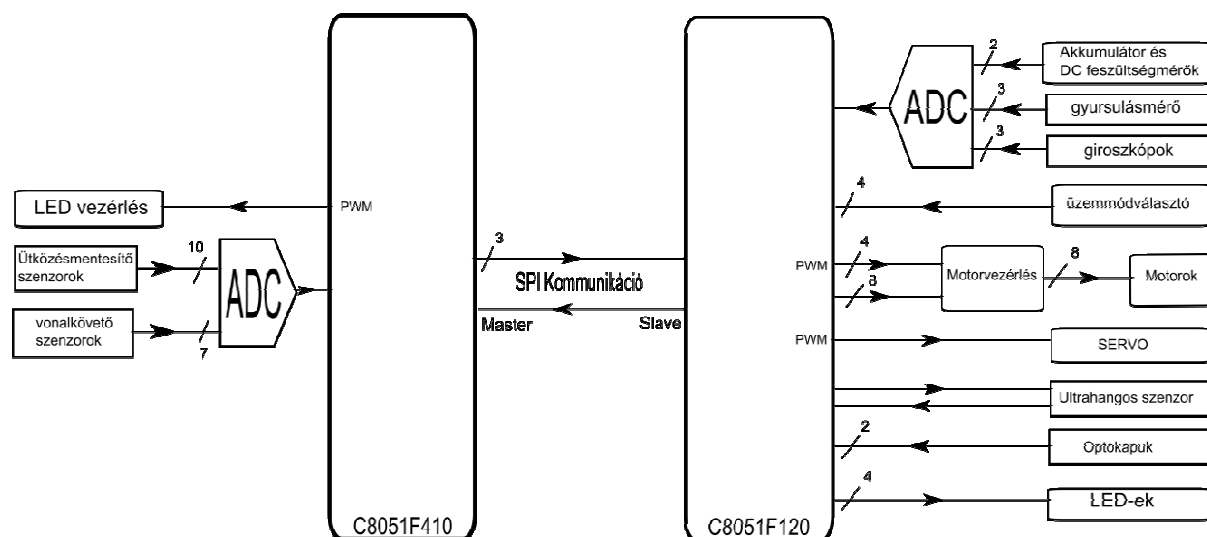
$$x = \frac{1000 \mu s}{0,49 \mu s} = 2040,816 \quad (4.1.)$$

A (4.1.)-es képlet alapján kiszámolt értéket ki kell vonnunk a 65535-ből, mivel most pont a PWM mód ellentettjét szeretnénk elérni, tehát az így kapott érték a 63494,184, ha a számláló eléri ezt a számot, akkor magas állapotba kerül, és addig ebben az állapotban marad, míg a számláló túl nem csordul, minden más esetben a kimeneten alacsony érték lesz. Tehát az így kapott érték a 63494,184, melyet kerekítenünk kell egészekre és át kell váltanunk 2-es számrendszerbe, 1111100000000110 ezt az értéket, ha szétbontjuk alsó és felső 8-bitre, akkor a kapott értékeket beírhatjuk a PCA0CPLn és a PCA0CPHn regiszterekbe.

5. Áramkör és paneltervezés

5.1. A robotvezérlő elektronika blokkvázlata

A blokkvázlat célja, hogy egyszerűen és átláthatóan megmutassa a rendszer funkcionális egységeit. Egy ilyen ábrát mutat be az 5.1. ábra.



5.1. ábra: A rendszer blokkvázlata

Az 5.1. ábrán látható, hogy mely funkciókat mely mikrovezérlő látja el. A vonalak mellett jelzett számok a vezetékek számát jelentik, ahol nincs számozás ott ez egyet jelent. A vonalakon szereplő nyilak jelentik azt, hogy bemenetről vagy kimenetről van szó. Látható továbbá, hogy melyek azok a szenzorok, amelyek jelének a feldolgozásához analóg-digitális átalakító (ADC) szükséges. Az ábra mutatja, hogy hol van szükségünk PWM (impulzus szélesség modulált) jel kibocsátására. A két mikrovezérlő SPI kommunikáción keresztül létesít kapcsolatot egymással, ahol jelölve van, hogy melyik a master és melyik a slave eszköz. A C8051F410 mikrovezérlő lett a master egység, mert gyakrabban fog ez a mikrovezérlő küldeni adatokat, mint az F120-as mikrovezérlő. Ennek abban van szerepe, hogy az F410-es mikrovezérlő figyeli és feldolgozza a szenzorok jeleit, majd ő fogja továbbítani az F120-as mikrovezérlőnek, aki a bejövő adatokra a kellő választ generálja majd.

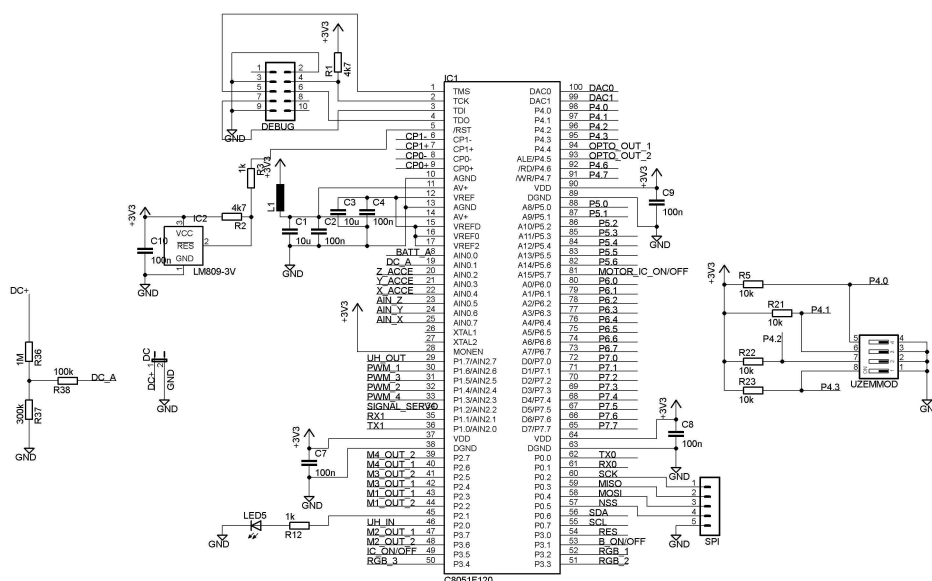
5.2. A fő vezérlőpanel

A fő vezérlőpanel főbb céljai, az egész rendszer vezérlése, összetartása, a csatlakozások megteremtése a többi panellel. Sorra bemutatásra kerülnek az egyes részei.

Először általánosságban az erőforrások alakulásáról, majd később az adott helyen következik be a pontosítás. A robot erőforrása egy Lithium-Polymer (Li-Po) akkumulátor lesz, mely kedvező tulajdonsága, hogy sokáig megtartja a benne rejlő feszültséget, és hogy nagy teljesítményű. A választott típusom 2400 mAh teljesítményre, és 8,4 V feszültség kiadására képes, a beszerzése az alábbi címről történt: <http://shop.modell.hu/>. Mivel az akku fix helyzetben lesz a robotban és nem szeretném minden egyes töltéskor kivenni a robotból, ezért a főpanelen kellett megoldani a töltést. A töltés a hálózati feszültségbe (+ 240 V) bedugott adapterrel történik, amely a kimenetén + 9 V-ot és 1,7 A áramerősséget állít elő. Ez a DC feszültség nem csak az akkumulátor töltésekor használható fel, hanem a robot tápellátására is szolgálhat.

5. 2. 1. Áramkörtervezés

A főpanel mikrovezérlője a Silabs cég által gyártott C8051F120 típusú mikrovezérlő lett, melynek ismertetése fentebb, az 1.3. fejezetben olvasható. A mikrovezérlő és a hozzá tartozó alkatrészek bekötési ábrája látható az 5.2. ábrán.

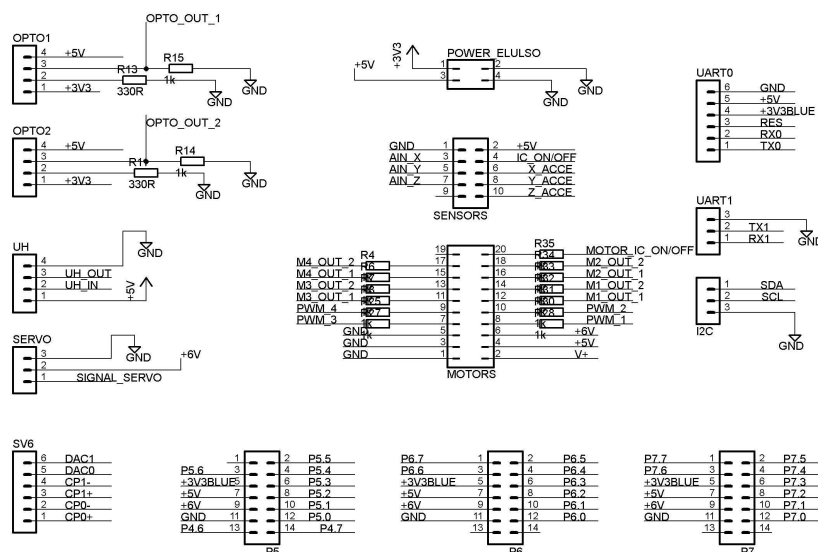


5.2. ábra: A C8051F120 mikrovezérlő és alkatrészének bekötése

Az 5.2. ábráról látható, hogy a mikrovezérlő + 3,3 V feszültségről fog működni. A mikrovezérlő programozása a DEBUG csatlakozón keresztül történik. Az SPI nevű csatlakozó az SPI-n keresztüli kommunikációt biztosítja az előző panellel. Az előző panelen lévő C8051F410 + 5 V feszültséggel működik, mivel a C8051F120 összes portja + 5 V toleráns, ezért a feszültség-átalakítás nem szükséges. Az ábrán helyet

kapott egy üzemmódválasztó kapcsolósor is, mivel a kapcsolók másik végeit felhúzó ellenállásokkal a + 3,3 V feszültségre kötöttem, ezért a zárandó kapcsoló magas jelet fog adni a mikrovezérlő megfelelő portjára, így fogjuk tudni, hogy melyik üzemmódot szeretnénk kiválasztani. A robot külső DC csatlakozóval rendelkezik, ezért annak a kimenetét egy tűskepárra vezetjük majd, ez után a DC feszültséget feszültségosztón és védőellenálláson keresztül vezetjük a mikrovezérlő egyik analóg bemenetére, így a DC feszültség szintjét tudjuk vizsgálni.

A főpanelen lévő csatlakozók bekötési rajzát mutatja be az 5.3. ábra.

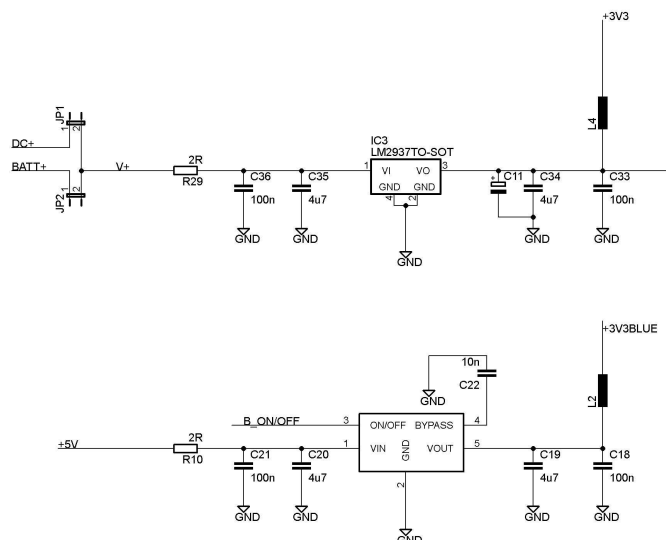


5.3. ábra: A főpanelhez tartozó csatlakozók

A panel csatlakozói közé sorolandóak az ultrahangos szenzor, az optokapuk, a szervó, a mikrovezérlő fel nem használt portjainak kivezetéseit tartalmazó csatlakozók. A mikrovezérlő más eszközzel való kommunikációját segítik az UART0, UART1, I2C (és az 5.2. ábrán látható SPI) csatlakozók.

A többi panellel való együttműködést segítik a SENSORS, POWER_ELULSO és a MOTORS nevű csatlakozók. A SENSORS csatlakozó a szenzoros panelhez biztosítja a kapcsolódást, a POWER_ELULSO pedig az elülső panelhez továbbíthatja a + 3,3 V és a + 5 V feszültségeket. A MOTORS nevű csatlakozó a motorvezérlő panelhez csatlakoztatható, portjai védőellenállással rendelkeznek, mivel a motorok nagy áramot (egyenként 470 mA) is felvehetnek, így a mikrovezérlőt védeni kell.

A főpanelen lévő feszültség átalakítók kapcsolási rajzát láthatjuk az 5.4. ábrán.



5.4. ábra: A főpanelen lévő feszültség átalakítók kapcsolási rajza

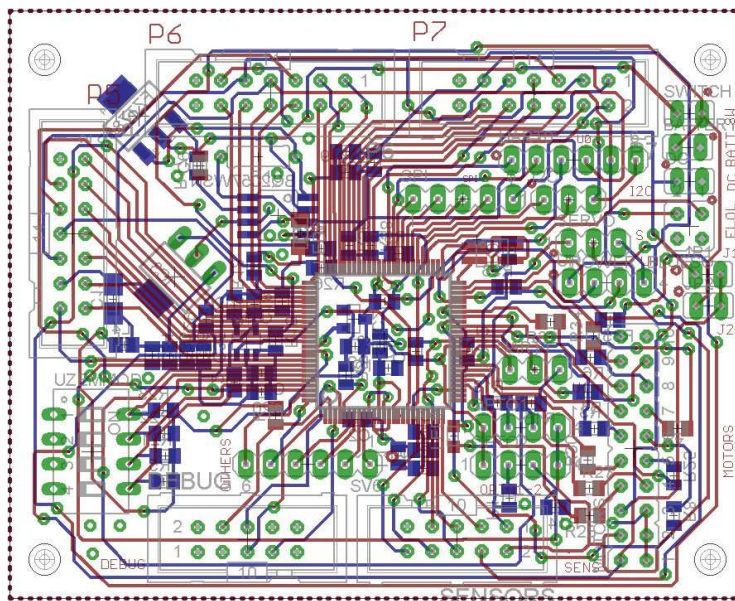
Az ábra felső részében a C8051F120-as működéséhez szükséges + 3,3 V-ot előállító LM2937 típusú IC látható, mely 500 mA-es és + 3,3 V-os kimenetet állít elő. Továbbá fontos itt megjegyezni, hogy a DC+ jelzés a DC csatlakozó, míg a BATT+ az akkumulátor pozitív pólusát jelenti. Ez a megoldás lehetővé teszi, a helyes jumpereléssel, hogy a robot működése vagy a DC feszültséget adó adatterről vagy az akkumulátorról történjen, a jumperek kimenete lesz a V+ jelzésű feszültség.

Az ábra alsó részénél látható kapcsolás, szintén + 3,3 V-ot állít elő, legfőképpen a bluetooth (vagy más) kommunikáció részére vált szükségessé. Ez az IC + 5 V-ról működik és a B_ON/OFF bemenet állapotával kapcsolható be-/ki.

A főpanel utolsó egysége az akkumulátor töltéséről gondoskodik, ennek a kapcsolási rajza látható az 5.5. ábrán.

5. 2. 2. A nyomtatott áramkör kialakítása

A fő vezérlőpanelen, a felső rétegen helyezkedik el a mikrovezérlő és néhány kisebb alkatrész, a csatlakozók és a LED-ek, az alsó rétegen pedig a kisebb alkatrészek. A panel tervezetét mutatja az 5.6. ábra.



5.6. ábra: A fő vezérlőpanel tervezete

Helytakarékosság és a roboton lévő furatok adott elhelyezkedése miatt, a panel mérete előre adott volt. A könnyebb csatlakozás érdekében szalagkábel csatlakozókat helyeztem el, melyek csak egyféleképpen csatlakoztathatóak az aljzatukba. A többi csatlakozón érdemes megjelölni az első port elhelyezkedését, a portok neveit, hogy véletlenül se csatlakoztassunk rosszul. A panel két rétege között érdemes minél több helyen a földelést átvezetni a két réteg között.

5. 3.Az ütközésselkerülő és vonalkövető panel

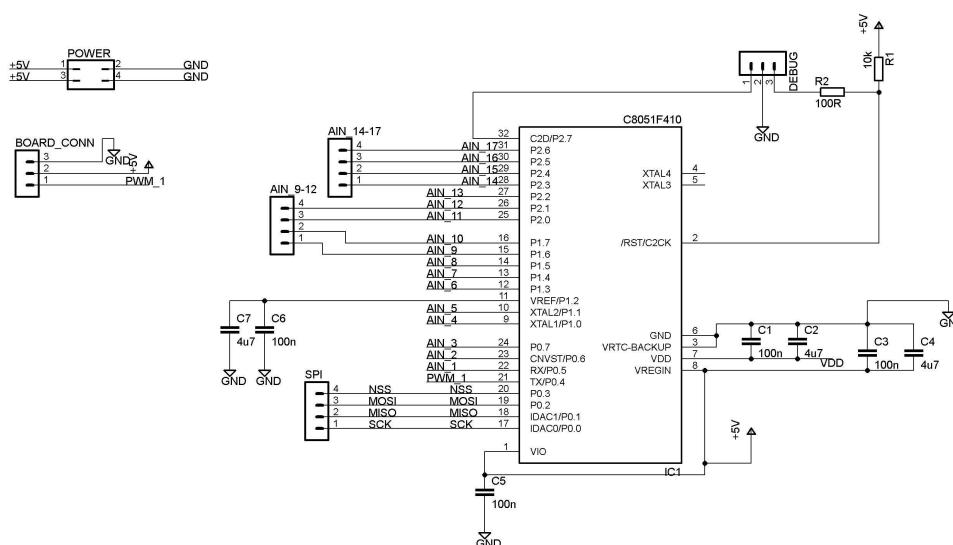
Az ütközésselkerülő és vonalkövető áramkör három főcsoportba rendezhető:

- a mikrovezérlő, és a hozzá tartozó csatlakozók
- az ütközésmegakozók szenzorokhoz tartozó alkatrészek
- a vonalkövető szenzorokhoz tartozó alkatrészek

Ezeket a csoportokat fogom bemutatni, a mikrovezérlővel kezdve.

5. 3. 1. Áramkörtervezés

A vezérlést a Silabs gyártmányú C8051F410 mikrovezérlő látja el. Azért erre esett a választásom, mert elegendő porttal rendelkeznek a funkciók elvégzéséhez, 12 bites ADC-vel (Analog-Digitális konverterrel) és SPI kommunikációval rendelkeznek. A mikrovezérlő bekötési rajzát láthatjuk az 5.7. ábrán.

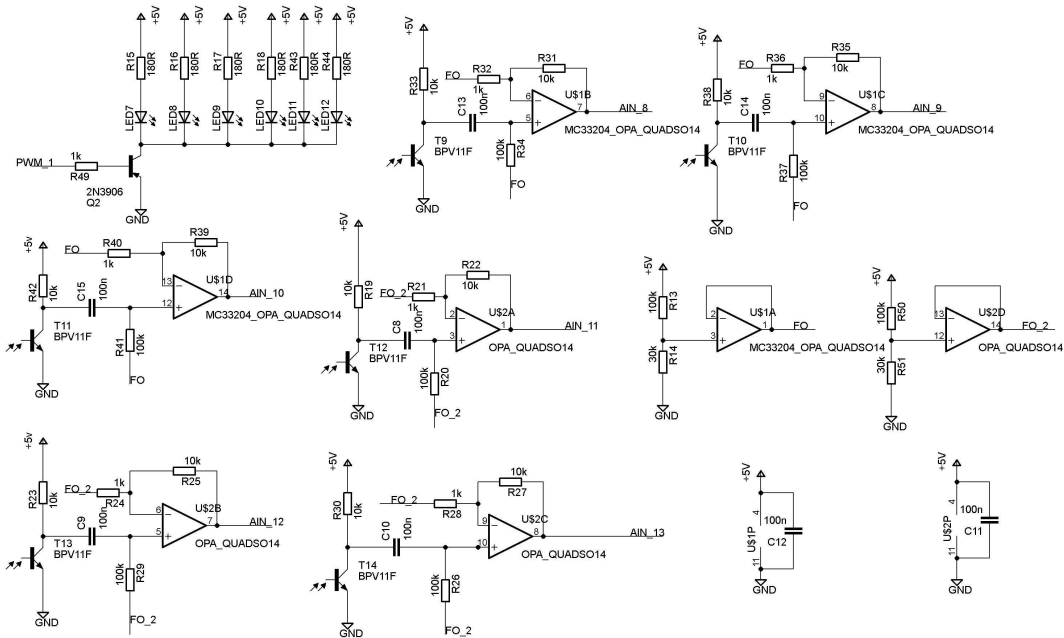


5.7. ábra: A C8051F410 mikrovezérlő kapcsolási rajza és a csatlakozók

Az 5.7. ábrán látható, hogy a mikrovezérlő programozása a 3 tús DEBUG adapterrel fog majd történni. Az áramkör működéséhez szükséges feszültséget a POWER csatlakozón fogja megkapni a főpanelről, a panel működése során csak + 5 V feszültségre lesz szükség. A már említett SPI kommunikáció számára szükséges portok kivezetése is megtörtént, ezt majd a főpanelre csatlakoztatjuk.

Látható még, hogy nem csak analóg bemenetekre van szükségünk, hanem egy PWM jelet is ki kell bocsátanunk. Ezzel az impulzussorozattal tudjuk majd működésbe hozni az infravörös fényt kibocsátó LED-eket.

Mivel a robot tolatása közben is szeretnénk, ha érzékelne a tárgyakat, ezért az áramkör kialakítása lehetővé teszi az analóg csatlakozók és a BOARD_CONN csatlakozó révén, hogy egy hátulra szerelhető panelen lévő szenzorokat is ugyanaz a mikrovezérlő tudjon kezelni. Tehát az elülső és a hátulsó panel közötti különbség csak a beforrasztandó alkatrészeken menyniségében lesz felfedezhető.

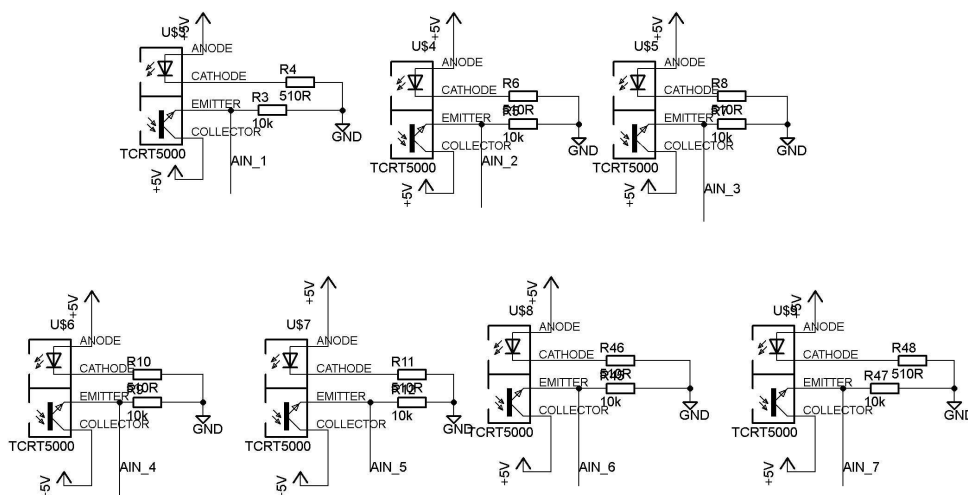


5.8. ábra: Az ütközésmentesítő szenzorok kapcsolási rajza

Az 5.8. ábrán látható, hogy a hat darab infravörös fényt kibocsátó LED-et PWM jellel fogjuk meghajtani. A 6 darab infravörös fényt kibocsátó LED miatt 6 fotótranzisztor és az ahhoz kapcsolódó alkatrészek bekötése látható még az 5.8. ábrán. Az ütközésmentesítő szenzornál, az 1.1.1. fejezetben leírtak alapján épül fel a kapcsolás, ezért itt már csak röviden ismertetem. A LED-ek be-/kikapcsolását egy PNP típusú tranzisztor teszi lehetővé. A fotótranzisztorok kimeneti jelét analóg jelként szeretnénk érzékelni, ezért a DC-feszültséget leválasztjuk egy kondenzátor és ellenállás segítségével. A szenzor nem képes nagy hatótávolságra, ezért a jelét 10-szeres erősítéssel fogjuk átvezetni. Az erősítés előtt még előállítjuk a +1,125 V-os feszültségosztó segítségével, mely létrehozható két sorosan kapcsolt ellenállással, az így létrehozott feszültség szolgál majd referenciapontként a DC-feszültség leválasztásában. Az erősítés után a mikrovezérlő analóg bemeneteire kötjük a szenzor felerősített kimeneti jelét, amely a referenciafeszültség miatt a GND és a +2,25 V feszültségtartományba fog esni, amely megegyezik az F410-es mikrovezérlő ADC konverterének a tartományával. A mikrovezérlő ADC konverterének az értéke, ha a tárgy hatótávolságon kívül van, megegyezik a +1,125 V referenciafeszültséghez tartozó

értékkel. A tesztelések során a LD274-3 infravörös LED és a BPV11F típusú fotótranzisztor páros bizonyult a legjobb választásnak.

A vonalkövető szenzorok, azok a szenzorok, amelyek a robot vonalkövetési funkciójához elengedhetetlenek lesznek. Az 5.9. ábra tartalmazza a szenzorok kapcsolási rajzát.



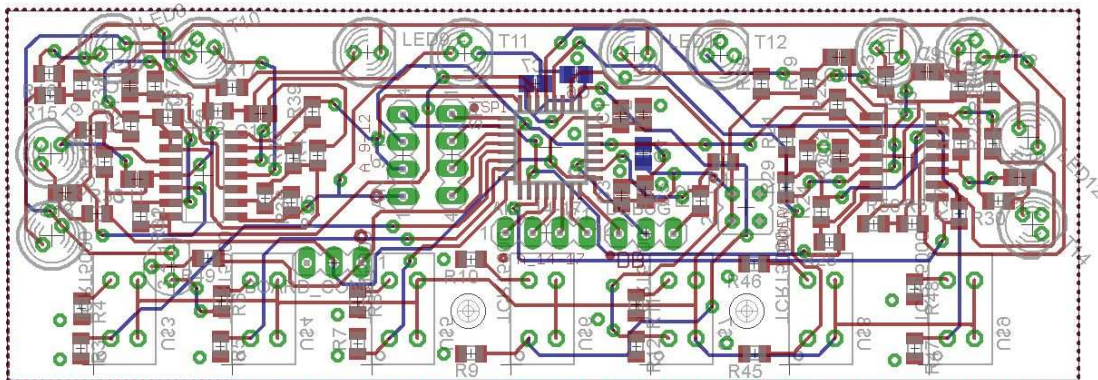
5.9. ábra: A vonalkövetéshez szükséges szenzorok bekötési rajza

Szenzoroknak TCRT-5000 típusú alkatrészt választottam, mely egybeépített infra LED-ből és fotótranzisztorból áll, mely a kialakítása révén le is van árnyékolva. A szenzorok meghajtása + 5 V feszültségről működik, az 5.9. ábra bekötési rajza alapján a fotótranzisztor emitter lábán megjelenik az elektromos jel, melyet a mikrovezérlő analóg bemenetére köthetünk. A TCRT-5000 előnye, hogy ha megkapja a kellő feszültséget, akkor a LED kibocsátja az infravörös fényt és a fotótranzisztor a visszavert jelet észlelni fogja, továbbá az árnyékolás gyárilag megoldott. Hátrányaként talán azt lehet elmondani, hogy majd a kész panelnek a roboton lévő elhelyezkedésénél ügyelni kell, hogy a padló (ahol a követendő vonal lesz) és a szenzor közötti távolság nem lehet nagyobb 5 milliméternél, mert ez a távolság tekinthető még megfelelőnek.

5. 3. 2. A nyomtatott áramkör kialakítása

A panel tervezése során elvárás volt, hogy egy olyan panelt tervezzek meg, amely más platformokra is kompatibilis lehet. A kialakítás során további elvárás volt, hogy egy olyan panel legyen a végeredmény, amely hosszú, de szélessége rövidnek mondható.

Ezeknek az elvárásoknak sikerült megfelelni, a kész panel 10 cm hosszú, és 3,45 cm széles lett. A tervezett panelt mutatja az 5.10. ábra.



5.10. ábra: Az ütközésselkerülő és vonalkövető panel tervezete

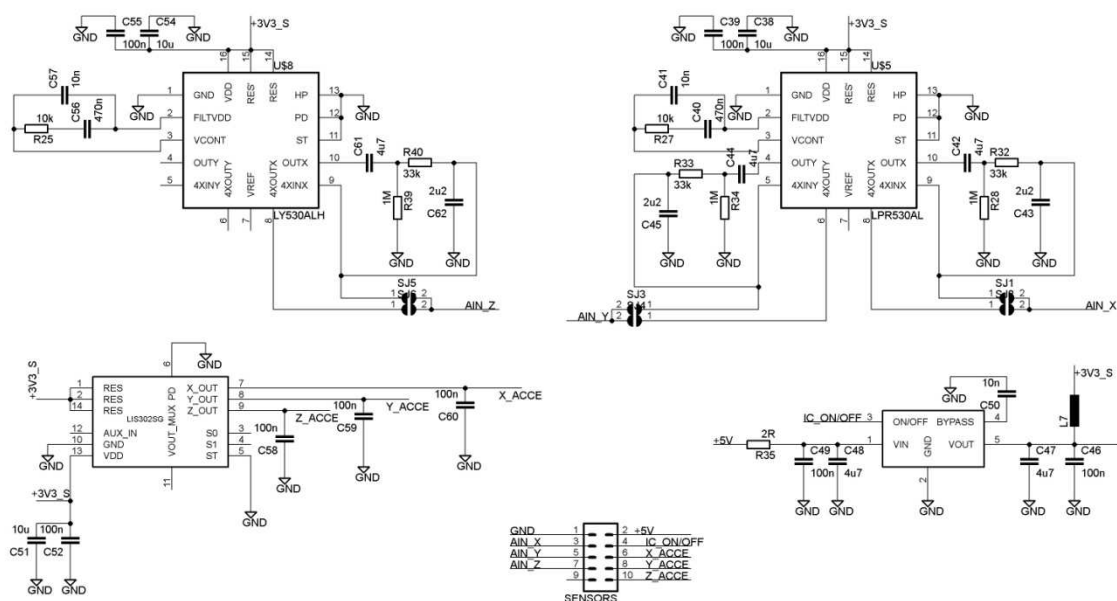
A tervezés során ügyelni kellett, hogy egy-egy LED és fotótranszisztor páros egymás mellett helyezkedjen el. Továbbá a kissé elforgatott szenzorokkal a panel érzékelési szöge növekedni fog. Lényeges, hogy a vonalkövető szenzorok a panel alján helyezkedjenek el, és azok is úgy forgatva, hogy lehetőleg az infravörös fényt kibocsátó LED legyen elől és mögötte a fotótranszisztor. Fontos még a furatok pontos elhelyezése a panel helyes rögzítése érdekében.

5. 4. Az inerciaszenzoros panel

Az inerciaszenzoros panel tartalmazza a két giroszkópot és a gyorsulásszenzort, melyek analóg kimenettel rendelkeznek.

5. 4. 1. Áramkörtervezés

A panel kapcsolási ábrája látható az 5.11. ábrán.



5.11. ábra: A szenzoros panel kapcsolási rajza

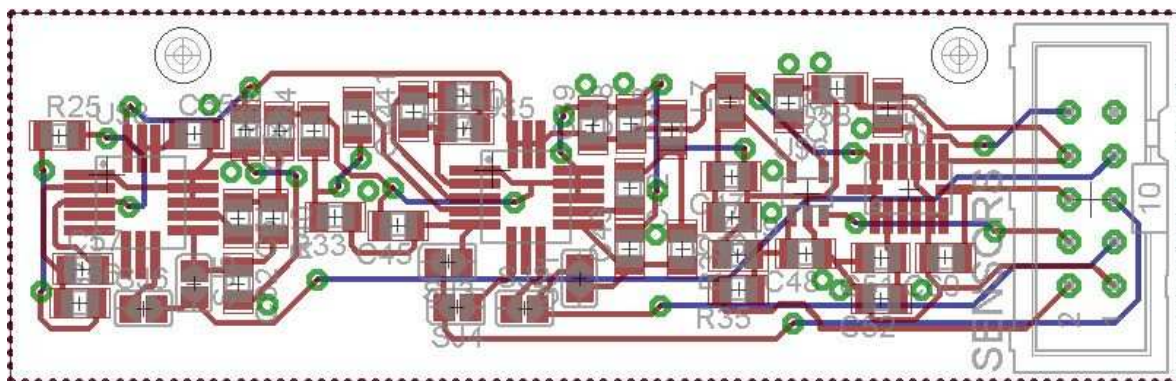
Az 5.11. ábra tetején látható a két giroszkóp szenzor. Mivel az LPR530AL típusú szenzor, csak X és Y tengely mentén tud érzékelni, ezért szükségessé válik egy LY530ALH típusú szenzor beépítése is, amely csak Z tengely mentén tud érzékelni. A giroszkópok analóg kimenetei a kellő szűrőkkel vannak felszerelve, forrasztással változtathatunk, hogy a 4-szeresen felerősített vagy az erősítetlen jelet dolgozzuk fel. A két szenzor ugyanazzal a ± 300 °/s sebességgel rendelkezik.

A gyorsulásszenzor egy LIS302SG típusú szenzor, mely szintén analóg kimenetekkel rendelkezik. Ez a szenzor ± 2 g gyorsulás érzékelésére képes, 3 tengely menti irányban.

Az 5.11. ábrán látható továbbá a szenzorokhoz szükséges feszültség előállító, mely a bejövő +5 V-ot alakítja át +3,3 V-ra. Ennek az IC-nek a működése az IC_ON/OFF bemenettel kapcsolható be-/ki. A szenzoros panel is a főpanelre lesz csatlakoztatható.

5. 4. 2. A nyomtatott áramkör kialakítása

Az 5.12. ábrán látható az inerciaszenzoros panel kialakítása.



5.12. ábra: Az inerciaszenzoros panel tervezete

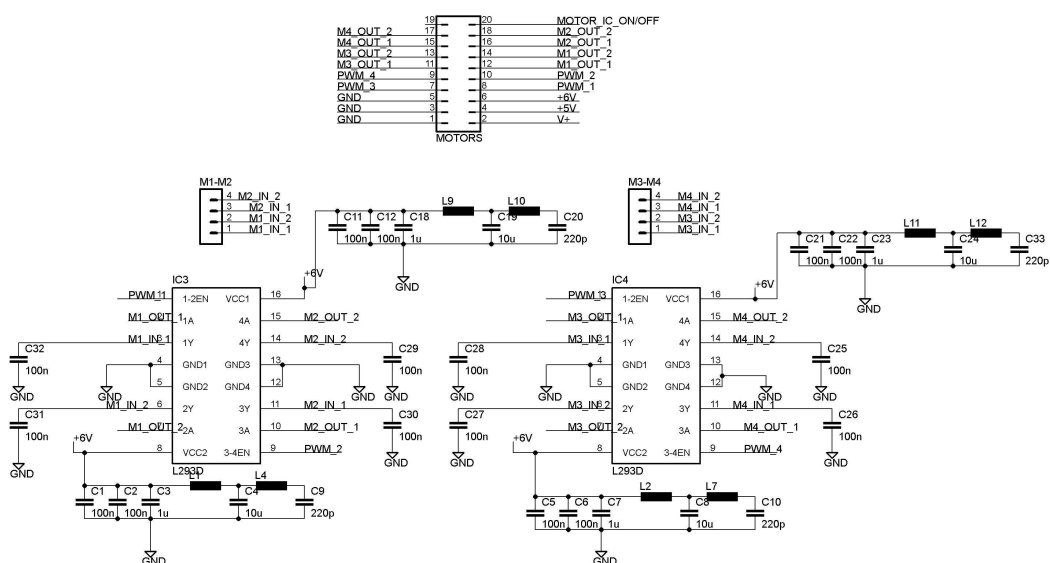
A panel közepén egy sávban balról jobbra helyezkedik el a két giroszkóp és a gyorsulásszenzor. A csatlakozó a panel szélére került, így könnyítve meg a kábelek elhelyezését. A jobb földelés érdekében érdemes a kondenzátorok közelében elhelyezni minél több, a két réteg közötti átmenetet.

5. 5.A motorvezérlő panel

A motorvezérlő panel feladatai közé sorolandó a négy darab motor vezérlése, és a + 5 V és + 6 V feszültségek előállítása és a főpanelhez való csatlakozás megteremtése.

5. 5. 1. Áramkörtervezés

A négy motor vezérlésére két darab L293DNE típusú IC-t használtam. Ennek a bekötési rajza látható az 5.13. ábrán. Továbbá a főpanelhez való csatlakozás is az 5.13. ábrán foglalt helyet.

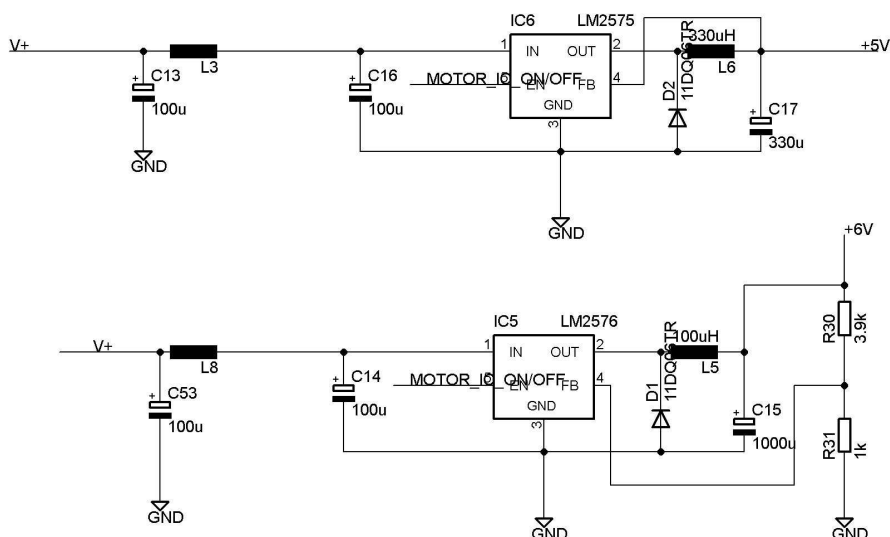


5.13. ábra: A motorvezérlő IC-k bekötése és a csatlakozók

A főpanel csatlakozóján keresztül érkezik a $V+$ feszültség (mely vagy az akku vagy az adapter feszültsége), továbbá itt történik a $+5\text{ V}$ és a $+6\text{ V}$ továbbítása is a főpanel felé. A csatlakozón keresztül válik megoldottá a motorok PWM meghajtása (szükségességét fentebb, az 1.2.1. fejezetben ismertettem), és a motorok vezérléséhez szükséges további jelek érkezése. Az M1-2 és M3-4 nevű csatlakozókra csatlakoztathatóak a motorok.

Az L293DNE típusú IC-k működési elve, fentebb az aktuátoroknál ismertette lett az 1.2.1. fejezetben. Ezek biztosítják a motorok helyes vezérlését.

A panel további feladata volt, a motorokhoz szükséges $+6\text{ V}$ előállítása, és a szenzoroknak, C8051F410-es mikrovezérlőnek szükséges $+5\text{ V}$ előállítása is. A DC-DC konverterek bekötési rajza látható az 5.14. ábrán.



5.14. ábra: A feszültséget átalakító IC-k kapcsolási rajza

Az ábrán jelzett $V+$ feszültség, mellyel az IC-k meghajtásra kerülnek, a főpanelről érkezik és vagy az akkumulátor, vagy annak a töltésére használandó DC feszültség lesz az. Az IC-eket védeni kell, az esetleges disszipációtól ezt megtehetjük kondenzátorral, a ferrit fojtónak a szűrésben van szerepe.

A $+5\text{ V}$ előállítása egy LM2575 típusú IC-vel történik [8]. Ez az integrált áramkör a bemenő feszültségből előállítja a kellő $+5\text{ V}$ -ot. Az EN (Enable) jelzésű lábán lévő MOTOR_IC_ON/OFF vezetéken a főpanelről jövő jellel tudjuk be- illetve kikapcsolni.

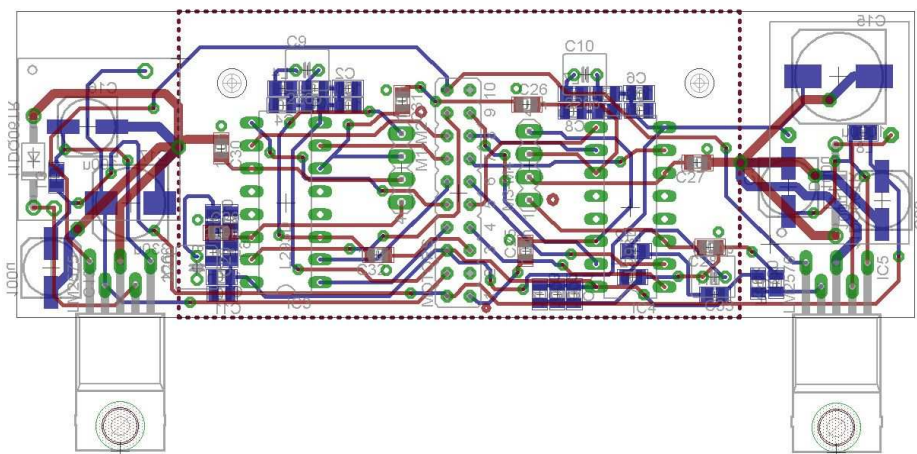
Mivel a +6 V-ról szeretnénk a motorokat meghajtani, amik egyenként felvehetnek 470 mA áramot, ezért az előállító IC-nek ezt az árammennyiséget elő kell állítania. A választásom egy LM2576 típusra esett, azért mert ez 3 A áramerősségű kimenettel rendelkezik [8], amely elég a 4 motor meghajtására. Ugyanúgy, ahogy az LM2575, ez az IC is az MOTOR_IC_ON/OFF jelzésű EN (Enable) lábra érkező jellel lehet be-/kikapcsolni (alacsony állapotra kapcsol be és magasra ki). Az IC kialakítása lehetővé teszi, hogy a kimeneti feszültség nagyságát, két sorba kapcsolt ellenállással vezérelni tudjuk, ezt hívják programozható kimeneti feszültség módnak. A kimeneti feszültség kiszámítására szolgál az (5.1.) képlet.

$$V_{OUT} = V_{ref} \left(1 + \frac{R_2}{R_1} \right) \quad (5.1.)$$

Ahol a képletben szereplő $V_{ref} = +1,23 \text{ V}$, míg az ellenállások az 5.14. ábra alapján az R30-as jelzésű ellenállás megfelel a képletben szereplő R_1 ellenállásnak, míg az R31 az R_2 ellenállásnak felel meg. Az értékek helyes behelyettesítésével megkapjuk kerekítve a +6 V-os kimeneti feszültséget.

5. 5. 2. A nyomtatott áramkör kialakítása

A panel tervezése során célszerűnek láttam a csatlakozók elhelyezését a panel közepén, tőle jobbról és balról az L293DNE vezérlő IC-k elhelyezését, majd a panel bal és jobb szélein a feszültségeket átalakító IC-k elhelyezkedését. Ennek a panelnek a tervét láthatjuk az 5.15. ábrán.

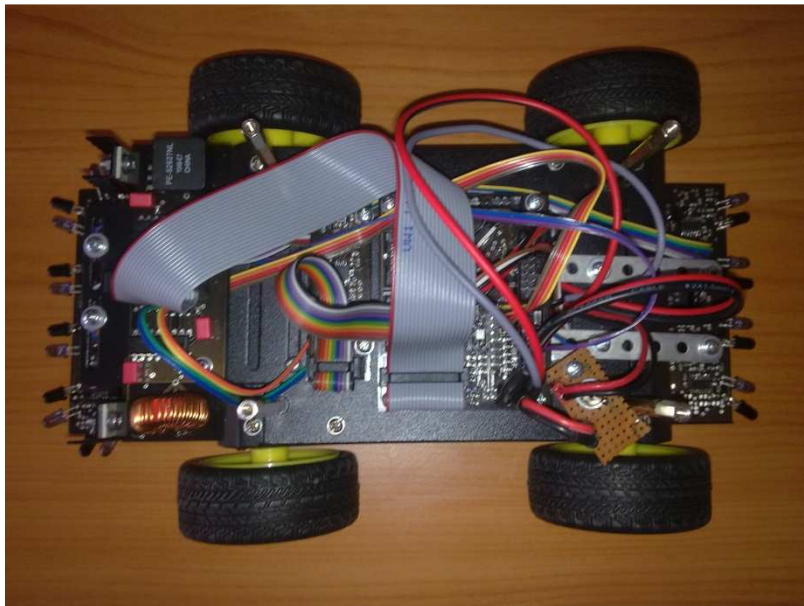


5.15. ábra: A motorvezérlő panel kialakítása

Mivel a motorok felvehetnek terhelt állapotukban, a már említett 470 mA erősségű áramot, ezért a panelen szélesebb vezetékekre volt szükség. Az L293DNE IC

két feszültség bemenetét a jobb hűtés érdekében két külön futó vezetékkel érdemes megvalósítani. Úgynevezett „heat sink” hűtést is kell alkalmazni, ez a GND lábak közelében hagyott földszigetekkel valósítható meg. Az IC-k közelében elhelyezkedő kondenzátorok is a zavarcsökkentést segítik. A feszültségeket előállító IC-k a panel jobb és a bal szélén láthatóak, ezeknél a földelést csillagpontos földeléssel kellett megoldani, melyekhez még szélesebb vezetékek tartoznak [8]. Az adatlap megadta a DC-DC konverterekhez használatos tekercsek típusát is, ezeknek a kimeneti feszültség zavarcsökkentésében, simításában van szerepük. [8]

A robot alvázának adatait tartalmazza az alábbi dokumentum [12]. A robot rendszerében felhasznált összes panel kész, bedrótozott változatát láthatjuk az 5.16. ábrán.



5.16. ábra: A robot paneljeinek összeszerelt, bedrótozott állapota

6. Megvalósított funkciók

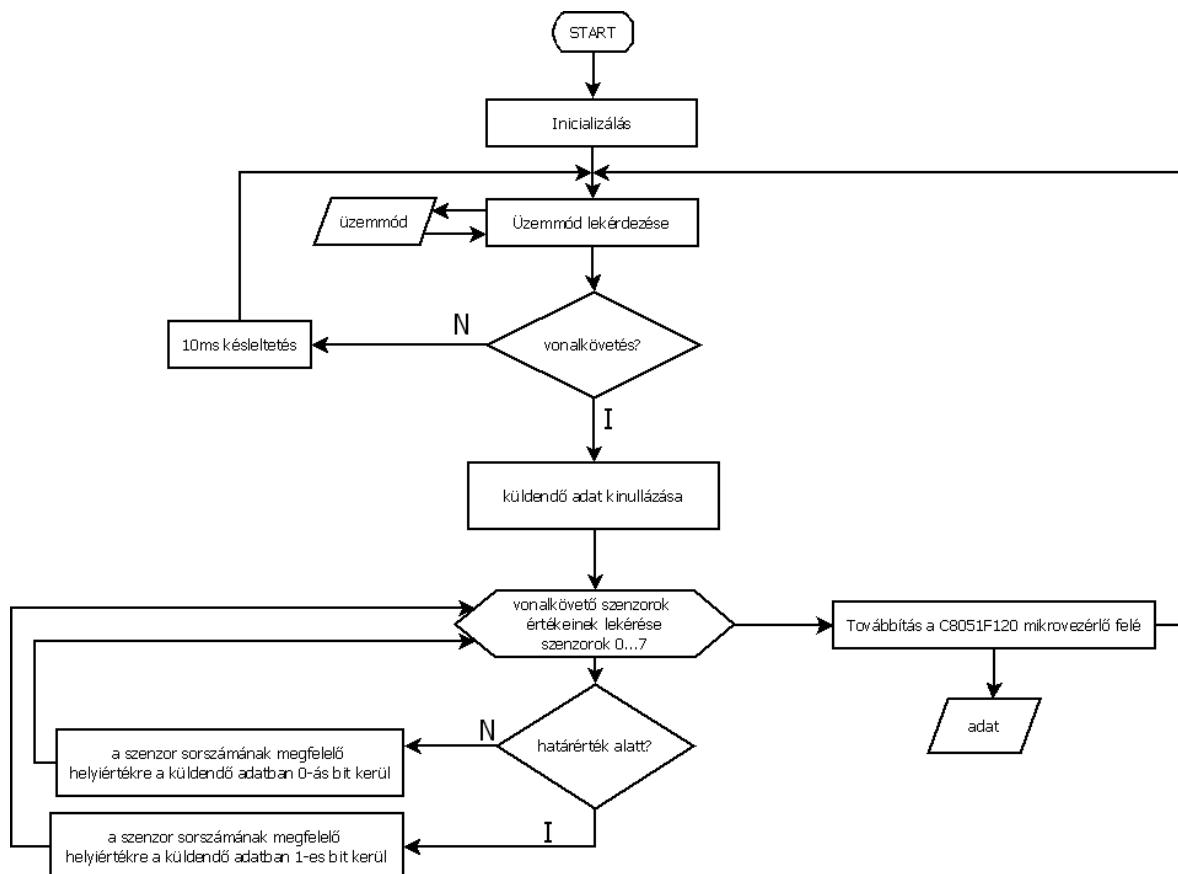
6.1. Vonalkövetési funkció

A vonalkövetési funkció segítségével a robotnak képesnek kell lennie az útvonalkövetésre. Ennek a funkciónak a leírása következik.

6.1.1. A C8051F410 mikrovezérlő szerepe a funkció megvalósításában

A C8051F410 mikrovezérlő fő feladatai, a már fentebb említett, vonalkövető és ütközésmentesítő szenzorok jeleinek érzékelése és feldolgozása, továbbá az SPI kommunikáció megteremtése a C8051F120 mikrovezérlővel.

A 6.1. ábrán láthatjuk a F410 mikrovezérlő működésének a folyamatábráját a vonalkövető funkció kibontásával együtt.



6. 1.ábra: A C8051F410 mikrovezérlő folyamatábrája, a vonalkövetés funkció kibontásával együtt

A 6.1. ábrán láthatjuk, hogy a START állapotból kiindulva, egy inicializálást követően, ahol a mikrovezérlő funkcióinak a beállítását végeztük el, eljutunk az

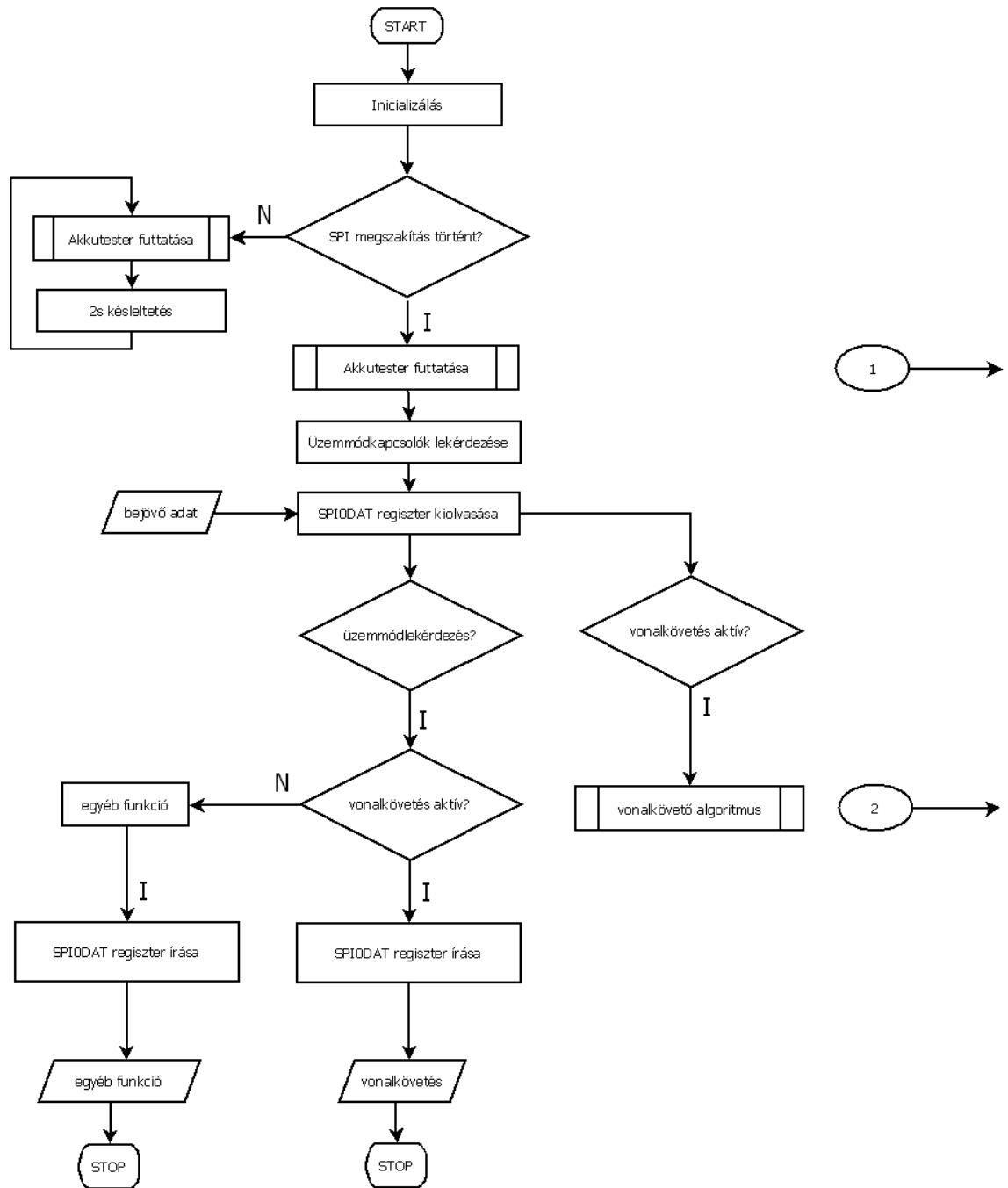
üzemmód lekérdezéshez. Ez a tevékenység, egy előre meghatározott byte-ot küld az F120 mikrovezérlőnek, ott ezt detektáljuk, és az üzemmódkapcsoló adatait beolvasva eldöntjük, hogy milyen üzemmód aktív ebben a pillanatban, ezt az adatot beleírjuk az SPI0 interfész adatregiszterébe (SPI0DAT), majd ezt visszaolvassuk a F410 mikrovezérlővel. Kettő opció közül választhatunk, vonalkövetés, vagy egyéb.

Ha a vonalkövetés aktív, akkor a küldendő adatot kinullázzuk, így ez minden egyes küldés előtt resetelődni fog. Mivel vonalkövető szenzorból hét darab áll rendelkezésemre, ezért a szenzorok jeleit egyesével beolvasva eldöntöm, hogy az előre definiált határértéknél kisebb-e, tehát a szenzor a vonal alatt van, így a szenzor sorszámanak megfelelő helyiértékbe 1-es értékű bit kerül. Ellenkező esetben, az adott helyiértékbe 0-ás értékű bit kerül. Erre azért van szükség, mert az SPI-n keresztül egyszerre, csak 8 bit küldhető, ennek a konverzióknak a segítségével elküldhetjük az összes szenzor értékét egyszerre. Az így átalakított adat kerül továbbításra az F120-as mikrovezérlő felé, majd kezdődik a ciklus előlről.

6.1.2. A C8051F120 mikrovezérlő szerepe a funkció megvalósításában

A mikrovezérlők gyakran generálhatnak, és kezelhetnek le megszakításokat, ennél a fejezet résznél fontos lejegyezni ezeknek az általános működési elvét. Megszakítást idézhet elő, valamilyen külső jel, ez lehet egy timer (számláló) túlsordulást jelző bit, egy ADC konverzió végét jelző flag, vagy éppen az SPI-n keresztüli sikeres adattovábbítás végét jelző bit is. A vonalkövetési funkciónál az utóbbi került felhasználásra. A futó programot bármikor megszakítás érheti, ha ez bekövetkezik, akkor a jel beérkezése előtti állapot elmentésre kerül, majd a megszakításkezelő rutin veszi át a program működését. A megszakításkezelő végrehajtja a bejövő megszakításra specifikált programrészt, majd ennek végrehajtása után, visszatér az elmentett, megszakítás előtti állapotba és a program futtatása folytatódik tovább.

Az F120-as mikrovezérlő folyamatábrája látható a 6.2. ábrán.



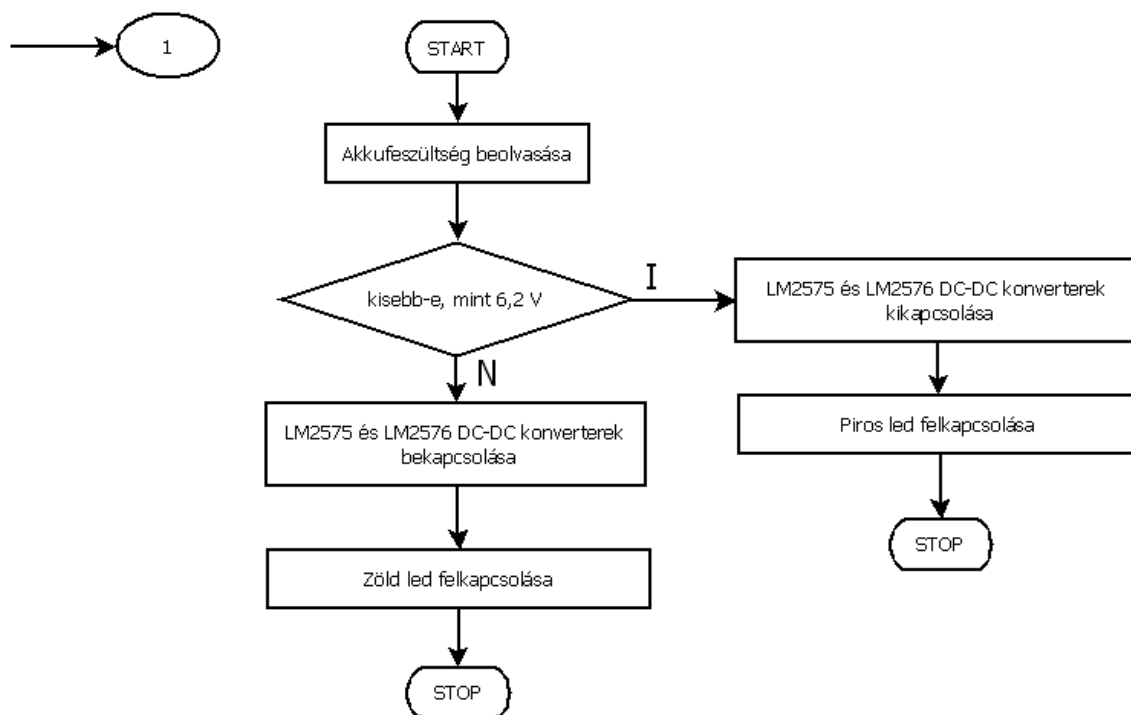
6. 2.ábra: Az F120 mikrovezérlő folyamatábrája

A 6.2. ábrán látható, hogy egy start állapotot követve elvégezzük a mikrovezérlő inicializálását.

Látható, hogy ha nem jön megszakítás (interrupt), akkor a program csak az akkumulátor töltöttségi szintjét fogja ellenőrizni és egy késleltetést hajt végre ciklikusan.

Az interrupt a program futása közben bármikor megtörténhet, ha ez megtörténik, akkor az SPI interrupt kezelő rutin lép életbe. Ez a megszakítás akkor jöhet létre, ha az

F410-es mikrovezérlő sikeresen elküldött egy adatot. A vételi oldalon, az interrupt beérkezése után, megvizsgáljuk az akkumulátor töltöttségi szintjét, ennek a folyamatnak a kibontása az 1-es sorszámmal történik. Az akku feszültség tesztelésére szolgáló folyamat ábráját a 6.3. ábra mutatja be.



6. 3.ábra: Az akku feszültségének tesztelésére szolgáló folyamat ábrája

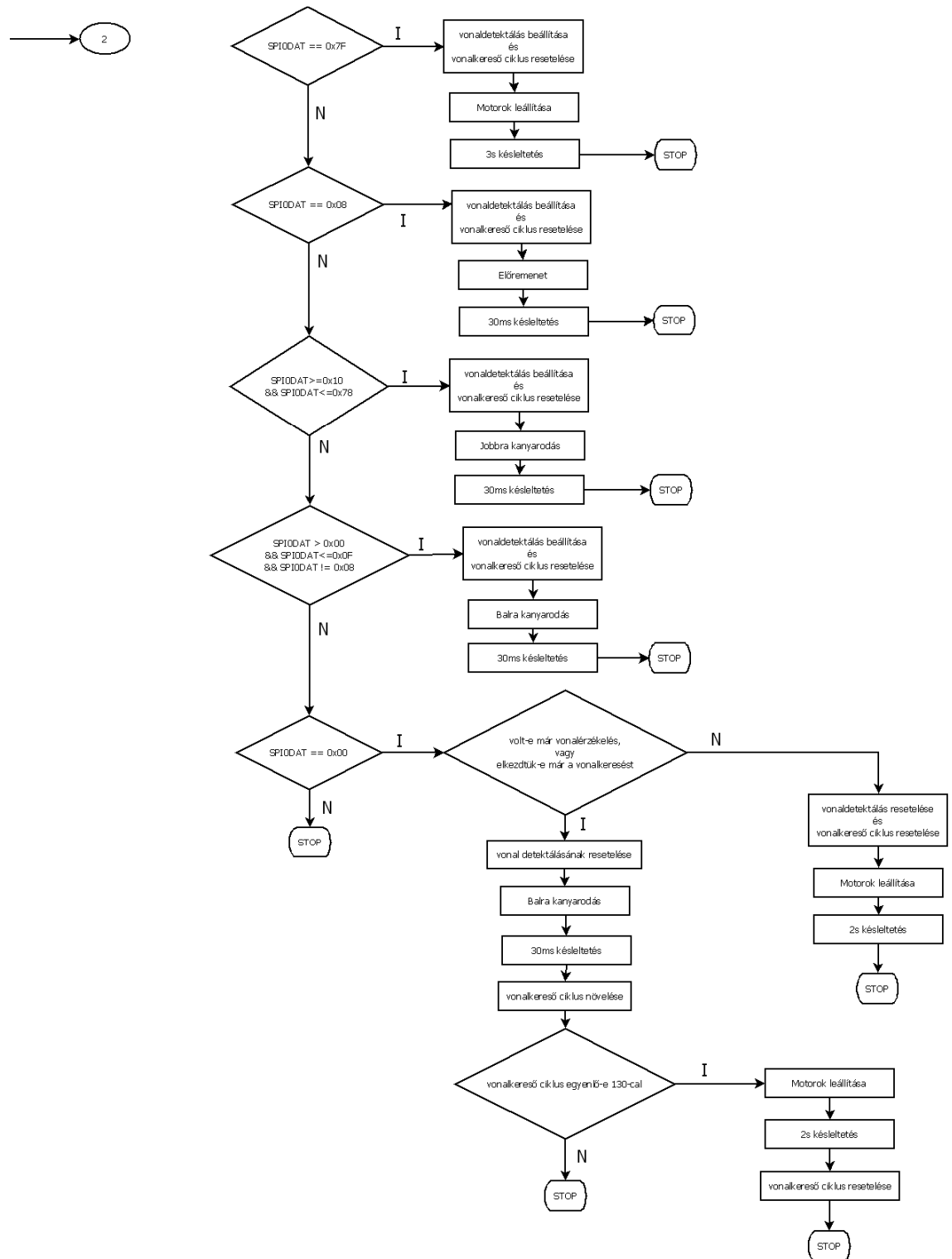
A 6.3. ábrán lévő folyamat szerepe, hogy ha már nincs elegendő feszültség az akkumulátorban, akkor ne engedje meg a dc-dc konverterek működését.

A 6.3. ábrán lévő folyamat lefutása után, továbbhaladva a 6.2. ábrai folyamatban, lekérdezzük az üzemmódkapcsoló állását, hogy melyik üzemmód éppen az aktív. Sikeres adatküldés esetén, valamilyen adatnak lennie kell az SPI0DAT regiszterben, vagyis az SPI adatregiszterében, ezt az adatot kiolvassuk.

Az adat kiolvasása után kerül eldöntésre, hogy melyik folyamat felé haladhatunk tovább. Ha az F410-es mikrovezérlő csak az üzemmódkapcsoló állását szeretné tudni, akkor a fentebb az F120 által lekérdezett üzemmód alapján visszaküldjük az aktív üzemmódot az F410-nek, így ő tudni fogja, hogy milyen adatokat kell lemérnie és a következőekben továbbítani.

Ha a vonalkövetési funkció aktív volt, akkor az F410 mikrovezérlő továbbította vonalkövető szenzorok adatai feldolgozásra kerülnek. Az alapján, hogy milyen adat érkezett be az SPI0DAT regiszterbe, vagyis az SPI0 interfész adatregiszterébe, más és

más hatása lesz a robot mozgásában. Ennek a vonalkövető algoritmusnak, a 6.2. ábrán, kibontására hivatkozott sorszáma a 2-es, ennek a sorszámnak a kibontott folyamatábráját láthatjuk a 6.4. ábrán.



6. 4.ábra: A vonalkövető algoritmus folyamatábrája

Még mielőtt ismertetném a funkciót, érdemes két változó hasznosságát lejegyezni. Az egyik változó akkor kerül beállításra, ha már egyszer vonalat detektáltunk, és csak akkor kerül resetelésre, ha már nem érzékeljük a vonalat. A másik fontos változóban tároljuk le, hogy elkezdtük-e már a vonal megkeresését, avagy sem. További részletezésre lentebb kerül sor.

Ha a bejövő adat 0x7F, ezt bináris számrendszerbe átváltva 1111111-t kapunk. A fentebb, a 6.1.1. fejezetben ismertetett átváltási módszer alapján, ez azt jelenti, hogy mind a 7 szenzor a vonal alatt van, ezért a motorokat megállítjuk, és 3 másodpercig késleltetünk, ennek szerepét majd lentebb ismertetem. Ha a bejövő adat 0x08, vagyis 1000, tehát csak a középső szenzor van a vonal alatt, ekkor egyenesen előre megyünk a robottal. Balra kanyarodunk, ha a bejövő adat 0x10 és 0x78 közötti zárt tartományban van, 100000 és 1111000 között, vagyis ha a középső szenzortól balra is bejelzett egy szenzor, addig a tartományig, ameddig csak a bal oldali és középső szenzor jelez be. Jobbra kanyarodás akkor történik, ha a bejövő adat 0x00 értéknél nagyobb és 0x0F értéknél kisebb vagy egyenlő, tehát ha 0 és az 1111 tartományban van, vagyis csak a jobb oldali szenzorok jeleztek be, kivéve, ha az adat a 0x08 számmal egyezik meg, mert akkor előre megyünk.

Ha a bejövő adat 0x00, tehát nem érzékelünk vonalat, akkor megnézzük, hogy volt-e már ez előtt vonaldetektálás vagy elkezdtük már a vonal megkeresését, erre azért van szükség, hogy ha nem volt vonaldetektálás még vagy vonalkeresés, akkor ne kezdje el a robot feleslegesen megkeresni a vonalat. Ha egyik feltétel igaz, akkor a vonaldetektálást reseteljük és elkezdünk balra forogni 30 ms-os késleltetés segítségével, ez után a vonalkereső ciklust növeljük eggyel. Észrevehető, hogy ha még nem találtuk meg a vonalat, akkor újra ebbe a ciklusba fogunk kerülni, mert már elkezdtük a keresést, és így balra foroghatunk tovább. Ha a keresési ciklus elérte a 130-as értéket, ez felel meg egy teljes kör megtételének, akkor megállítjuk a motorokat, a ciklusváltozót reseteljük, mert sajnos nem találtuk meg a vonalat és nem akarjuk, hogy a ciklusváltozó miatt újrakezdje a keresést.

Minden más esetben, ha a beérkező adat 0x00, akkor reseteljük a vonaldetektálást és a vonalkereső ciklus változókat és megállítjuk a motorokat, majd késleltetünk 2 másodpercig.

Ez a vonalkeresési funkció szükségessége, akkor vált észrevehetővé, amikor a robot a derékszögű kanyart szeretne volna bevenni, de ha pont a derékszög előtt mintavételezi a szenzor jeleit, mire reagálni tudna rá, addig a robot nem tud helyes

választ adni a motoroknak, ezért a robot nem fogja a kanyarodást elvégezni. A fentebb említett funkció segítségével, ha a robot túl is megy a vonalon, képes lesz korrigálni és visszatéríteni magát a vonalra.

Látható, hogy a motorok helyes beállítása után, az adott irányba fog mozogni a robot, de az előre, jobbra és balra kanyarodáskor kell egy tíz milliszekundum nagyságrendű késleltetés is, mert ennek hiányában a motor mozgása darabossá válik. A kipróbált folyamatban 30 milliszekundumos késleltetést alkalmaztam. Észrevehető, hogy nem csak a beállított késleltetés nagysága lesz végrehajtva, hanem addig, míg megérkezik az új adat, annak lekezelése meg nem történik, addig a motorok iránya nem fog változni.

A megírt vonalkövető algoritmust, az alábbi három pályán teszteltem, amelyek a 6.5., 6.6. és 6.7. ábrákon láthatóak. A robot ezeken a pályákon oda-vissza irányban is végig tudott menni. A pályákon szereplő függőleges kis szakasz, arra a célra szolgál, hogy ott az algoritmus alapján megálljon a robot.



6. 5.ábra: A vonalkövető algoritmus tesztelésének első pályája

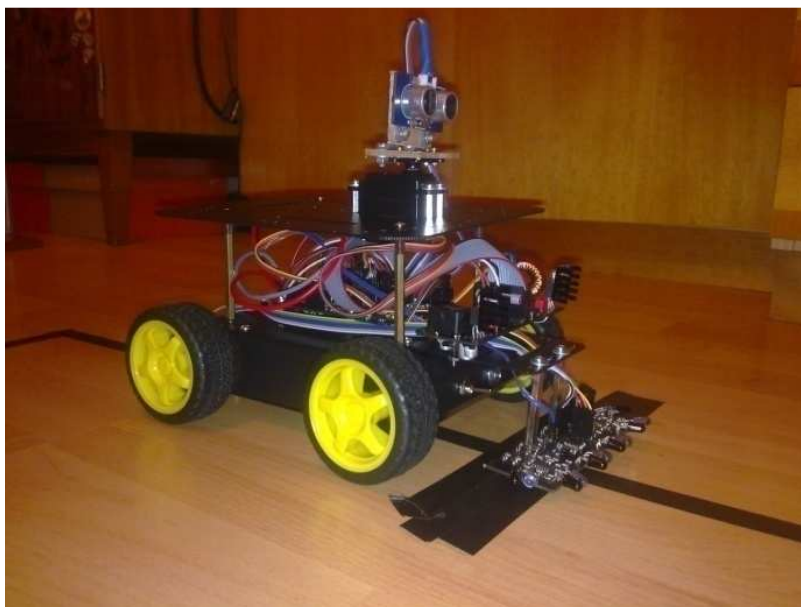


6. 6.ábra: a vonalkövető algoritmus tesztelésének második pályája



6. 7. ábra: A vonalkövető algoritmus tesztelésének harmadik pályája

A robotot a vonalkövetési funkciójának megállási pontjánál láthatjuk a 6.8. ábrán.



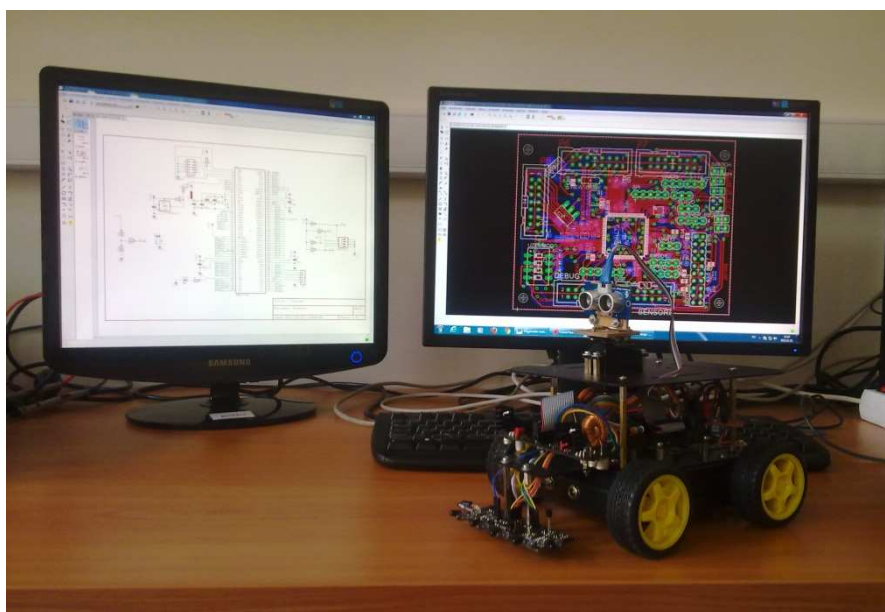
6. 8. ábra: A robot vonalkövetési funkciójának megállási pontjánál

7. Összegzés

A több hónapos laboratóriumi munkám során, behatóbb ismereteket szereztem az elektronika, áramkörtervezés, mikrovezérlők és robotika témakörökben. A megszerzett tudást a szakdolgozatom első részében az irodalmi áttekintés fejezeteiben ismerttettem. A szakdolgozatom második felében, a saját munkámban, a gyakorlatba is átültettem az elméleti tudást. A feladatok megoldása során törekedtem a legoptimálisabb megoldásokra, ezek a mérnöki szemléletmódomat is fejlesztették.

A robot, melyet terveztem és összeraktam, egy összetett rendszerként fogható fel, amelyben számos hiba befolyásolhatta a robot működését. Ezeknek a hibáknak a felderítése és kijavítása jó tapasztalatokat adott a jövőbeli, hiba mentesebb munkálatokban.

A robot teljes állapotát mutatja be a 7.1. ábra.



7.1. ábra: A robot végleges állapotában

7. 1. Eredmények

A munkám első fázisában felkutattam a felhasználni kívánt szenzorokat és aktuátorokat. ezek beszerzése után végezhettem el rajtuk a teszteléseket, hogy megbizonyosodjak arról, hogy ezek kellőképpen használhatóak és integrálhatóak a robot rendszerében.

Az ütközésmentesítő szenzor áramkörét próbapanelen raktam össze, majd méréseket végeztem rajta, ezek megkönnyítették a munkámat, amikor a szenzort tartalmazó panel áramkörét terveztem. Az ultrahang alapú távolságszenzort kipróbálás

után megbízhatónak tekintettem, így felhasználhatóvá válik a pontos távolságmérésben. Ebben a fázisban építettem meg az optokapukat is, melyeknek a távolságmérésben van szerepük.

Az aktuátorok fontos részét képezik a robotnak, közé tartoznak a motorok és a szervó. A motorok teszteléséhez szükséges L293DNE integrált áramkört tartalmazó áramkört próbapanelen építettem meg, mely hasznos információkkal szolgált a későbbi motorvezérlő panel áramkörének tervezésében. A szervó motor vezérlését is ebben a szakaszban oldottam meg, mellyel sikerült a kívánt elfordulási szöghöz társítani a megfelelő PWM jelet.

Következő lépésben meghatároztam a robot funkcióit, ez után lehetővé vált a blokkvázlat elkészítése, mely a robot építésének további szakaszaiban megbízható háttérrel nyújtott.

Tanulmányoztam a felhasználni kívánt integrált áramkörök adatlapjait, majd az alkatrészek beszerezhetőségét vizsgáltam. A pillanatnyilag, vagy egyáltalán nem beszerezhető alkatrészeket mellőztem, helyükre hasonló alkatrészeket kerestem.

A tesztelések során szerzett információk, a próbapanelen megépített kapcsolások, és az adatlapok alapján kezdtem el az áramkörök megtervezését. Ehhez nyújtott segítséget az Eagle program.

A kész áramköri kapcsolások alapján készítettem el a nyomtatott áramköri lapok tervezeteit, ehhez az Eagle program panelszerkesztő felületét használtam fel. Az elkészült paneleken rövidzár tesztet futtattam le, majd elkezdtem az alkatrészek beforrasztását. Az összes alkatrész beforrasztása után funkcionális és teljes körű tesztelést hajtottam végre a nyomtatott áramköri lapokon, így bizonyosodva meg arról, hogy a panelek külön-külön használhatóak, de még szükséges a rendszer teljes állapotában végrehajtott tesztelés is.

A robot rendszerének összerakását elvégeztem, a paneleket a megfelelő helyükre illesztettem, majd a paneleket megfelelően vezetékkeztem. Ez után volt lehetőségem a robot rendszerének teljes mértékű tesztelésére, amelynek végeredménye, hogy a megépített hardver megbízhatóan és biztonságosan működtethető és felhasználható.

Az utolsó eredményem, egy vonalkövetési funkciót ellátó program megírása volt. Ez az elkészült program jól példázza a két mikrovezérlő együttműködését és a rendszer összetettségét.

7. 2. Továbbfejlesztési lehetőségek

A robot áramköreinek és paneljeinek megtervezésének egyik fő célja az volt, hogy a rendszert tovább lehessen fejleszteni, erre a megépítésre került hardver képes.

Fejlesztési lehetőségek között sorolnám fel az ütközésmentesítés szoftveres megoldását, amely érdekében a hardvert nem kell átalakítani.

Számos további lehetőség adódhat még: A C8051F120 mikrovezérlő kommunikációs portjainak kivezetései lehetővé teszik, hogy külső kommunikációs egységet csatlakoztassunk a mikrovezérlőhöz. Ezáltal a számítógéppel való kommunikáció is lehetővé válik.

A robotra csatlakoztathatóak egy lapos számítógépek is, mint például a BeagleBoard, vagy a Raspberry Pi, melyek szintén a robot funkcióinak kibővítését jelenthetik.

A C8051F120 mikrovezérlő rendelkezik olyan I/O portokkal, melyeket a szakdolgozatom során nem használtam fel, de a panelen csatlakozók segítségével elérhetővé tettem. Ezek a portok is továbbfejleszthetőségi lehetőségként szolgálnak.

Irodalomjegyzék

- [1] BQ2057WSN akkumulátor töltő IC adatlapja (Megtekintés dátuma: 2012. 12. 03.):
<http://www.ti.com/lit/ds/symlink/bq2057w.pdf>
- [2] C8051F120 mikrovezérlő adatlapja (Megtekintés dátuma: 2012. 12. 03.):
<http://www.silabs.com/Support%20Documents/TechnicalDocs/C8051F12x-13x.pdf>
- [3] C8051F410 mikrovezérlő adatlapja (Megtekintés dátuma: 2012. 12. 03.):
<http://www.silabs.com/Support%20Documents/TechnicalDocs/C8051F41x.pdf>
- [4] Edaq530 oktatási mérőműszer kapcsolási rajza (Megtekintés dátuma: 2012. 12. 03.):
<http://www.noise.inf.u-szeged.hu/edudev/EDAQ530/Edaq530Design.pdf>
- [5] Harry N. Norton – Handbook of transducers, Prentice Hall PTR, New Jersey, 1989
- [6] Lambert Miklós – Szenzorok – elmélet és gyakorlat, Press+Print Lapkiadó, Budapest, 2009
- [7] LIS302SG típusú gyorsulásszenzor adatlapja (Megtekintés dátuma: 2012. 12. 03.):
http://www.st.com/internet/com/TECHNICAL_RESOURCES/TECHNICAL_LITERATURE/DATASHEET/CD00187369.pdf
- [8] LM2575 és LM2576 DC-DC konverterek adatlapja (Megtekintés dátuma: 2012. 12. 03.):
<http://www.datasheetcatalog.org/datasheet/SemtechCorp/mXvqxxtq.pdf>
- [9] LPR530AL típusú giroszkóp adatlapja (Megtekintés dátuma: 2012. 12. 03.):
<http://www.sparkfun.com/datasheets/Sensors/IMU/lpr530al.pdf>
- [10] LY530ALH giroszkóp adatlapja (Megtekintés dátuma: 2012. 12. 03.):
<http://www.sparkfun.com/datasheets/Sensors/IMU/LY530ALH.pdf>
- [11] L293DNE motorvezérlő IC adatlapja (Megtekintés dátuma: 2012. 12. 03.):
<http://idmax.free.fr/Aide/Stepper/l293.pdf>
- [12] Robot alvázának adatlapja (Megtekintés dátuma: 2012. 12. 03.):
<http://www.dfrobot.com/image/data/ROB0003/ROB03-Instruction%20ManualV2.0.pdf>
- [13] Roland Dilsch – A 8051 mikrokontroller család, Műszaki Könyvkiadó, Budapest, 1993
- [14] TCRT5000 optikai szenzor adatlapja (Megtekintés dátuma: 2012. 12. 03.):
<http://www.vishay.com/docs/83760/tcrt5000.pdf>
- [15] Thomas Bräunl – Embedded Robotics, Mobile Robot Design and Applications with Embedded Systems, Second Edition, Springer-Verlag Berlin Heidelberg, Germany, 2006
- [16] U. Tietze, Ch. Schenk – Analóg és digitális áramkörök, Műszaki Könyvkiadó, Budapest, 5. javított kiadás, 1993
- [17] Ultrahangos szenzor működésének elvi ábrája (Megtekintés dátuma: 2012. 12. 03.):
<http://www.seeedstudio.com/wiki/images/6/6b/Ultra-Sonic-seq.JPG>

Nyilatkozat

Alulírott Zana Roland mérnök-informatika BSc szakos hallgató, kijelentem, hogy a dolgozatomat a Szegedi Tudományegyetem, Informatikai Tanszékcsoport Műszaki informatika Tanszékén készítettem, mérnök-informatikus BSc diploma megszerzése érdekében.

Kijelentem, hogy a dolgozatot más szakon korábban nem védtem meg, saját munkám eredménye, és csak a hivatkozott forrásokat (szakirodalom, eszközök, stb.) használtam fel.

Tudomásul veszem, hogy szakdolgozatomat / diplomamunkámat a Szegedi Tudományegyetem Informatikai Tanszékcsoport könyvtárában, a helyben olvasható könyvek között helyezik el.

Szeged, 2012. 12.

.....

Aláírás

Köszönetnyilvánítás

Ez úton szeretném megköszönni, Dr. Mingesz Róbertnek, témavezetőmnek, és Mellár Jánosnak, tanszéki mérnöknek, a szakdolgozatom kapcsán nyújtott segítségüket. Szeretném megköszönni a zajlabor további dolgozóinak és szakdolgozóinak, akik a bent töltött munkanapokat színesebbé tették. Köszönet illeti a Műszaki Informatika Tanszéket, hogy biztosították nekem a munkám során használatos eszközöket és a kellő anyagi háttérrel.

Mellékletek

Egy darab CD, mely tartalmazza a kapcsolási rajzokat, és a programkódokat.