



TUDOMÁNYOS DIÁKKÖRI DOLGOZAT

MOBILROBOT TERVEZÉSE

Szerző(k): **Otlakán Péter**
villamosmérnök BSc. szak, I. évf.
Tóth Tamás
villamosmérnök BSc. szak, I. évf.
Üveges Krisztián Tibor
villamosmérnök BSc. szak, I. évf.

Konzulens: **Borsos Döníz**
tanársegéd

Tartalom

1.	Bevezetés	4
2.	Tervezési folyamat.....	6
2.1	A kezdet	6
2.2	Eddigi versenytapasztalatok.....	6
2.3	Specifikációk pontos meghatározása	7
3.	3D terv elkészítése [1][2][3][4][5]	10
4.	Elektronika	18
4.1	Logikai rendszerterv	18
4.2	Fizikai rendszerterv.....	19
4.2.1	18650 Lithium-Ion akkumulátor	21
4.2.2.	Akkumulátor vezérlő egység	22
4.2.3.	NodeMCU-32S ESP32	23
4.2.4.	Motorvezérlő.....	24
4.2.5.	GP2Y0A41SK0F infrás távolságérzékelő [24]	25
4.2.6.	MPU6050 gyorsulás érzékelő szenzor [15][16][25]	26
4.2.7.	MFRC522 RFID [14][23]	27
4.2.8.	IR- rés-opto [21].....	28
4.3	Teszt áramkörök.....	29
4.3.1.	Motorvezérlő és rés-optók	29
4.3.2.	B.M.S - Akkumulátor vezérlő egység [17][18][19][20]	30
4.4	Végző kapcsolás [22]	35
5.	Fizikai megvalósítás.....	36
5.1	3D nyomtatás	36
5.2	Nyomtatott áramkörök	37
5.3	Áramkör tesztelése.....	38
6.	Szoftver	41
6.1	RTOS ismertetése [6][7][8]	41
6.2	Hajtáslánc vezérlés és visszacsatolás szenzor adatok alapján.....	42
6.3	Útvonal feltérképezés és tájékozódás [8][11][12][13].....	43
6.4	Feltérképezett útvonal optimalizálás, végző gyors kör	44
7.	Fejlesztési lehetőségek.....	45
8.	Összefoglalás	46

9.	Források:	47
9.1	Ábrák:	47
9.2	3D modellek:.....	47
9.3	Szoftver:.....	48
9.4	Adatlapok:.....	49
10.	Csatolmányok:	51

1. Bevezetés

Az alábbi dolgozat témája egy labirintusban autonóm módon közlekedő, RFID (Radio Frequency IDentification - Rádiófrekvenciás Azonosítás) címkékkel és távolságméréssel tájékozódó robot tervezési folyamatai. A munka tartalma, a projekt kivitelezése lesz, hogyan gondolkodtunk, milyen problémákkal szembesültünk a tervezés és a megvalósítás során, ezeket hogyan oldottuk meg. A korábbi, hasonló munkák tapasztalataiból és beszámolások alapján igyekeztünk a feladat kihívásait a lehető leghatékonyabb módon abszolválni innovatív megoldásokkal.

A Kandó Kálmán Szakkollégium minden évben rendszeres szakszemináriumokat és programokat biztosít az egyetemi élet és szakmai tudás fejlesztésében. Így nyílt lehetőségünk a 2023-as Kutatók éjszakáján bemutatott labirintus versennyel és az Ipari projektek nevű tárggyal megismerkedni. A verseny szabadon választható tárgyként felvehető, egyetemi hallgatóknak szánt rendezvény. A cél egy olyan robot megalkotása, amely a préselt falapokból épített sorok között, a négyzet alapú terület egyik végéből a másikba eltaláljon. A feladat időhöz kötött, a győztes szerkezet az, melyik a leggyorsabban kijut a labirintusból. A prezentált robotokról bemutatott fejlesztési kihívások és a probléma megoldására kifejlesztett módszerek kreativitása lehangoló volt számunkra. A rendezvényt kis baráti társaságban tekintettük meg, távozáskor döntöttük el, hogy a következő alkalommal saját csapatot formálva indulunk el rajta. Így alakult meg a DINOCO. Felkészülés, irodalomkutatás és egyeztetések után érdembeli munkát a rendezvényt követő hónapban kezdtük el és azóta is folyamatos a fejlesztés, hangolás alatt áll a robot.

Az alábbi dolgozatban a 2024-es Mobilrobotok az útvesztőben című labirintus versenyre szánt munkánknak részletes fejlesztését mutatjuk be, amelyben bemutatjuk a következők:

- Tervezés első lépései; kritériumok; versenyszabályzat elemzése;
- Konstrukciós tervek; elméleti tervezés;
- Robot mechanikája;
- Szenzoros érzékelés;
- Saját elektronika fejlesztése;
- Egyedi váz és kerekek gyártása;
- Prototípus életre keltése;
- Mikrovezérlő választás; ESP32;
- Szoftveres programozás; logika.

2. Tervezési folyamat

2.1 A kezdet

A tervezési folyamatot azzal kezdte a csapat, hogy végig tanulmányoztuk az előző (2023) évi versenyszabályzatot, ugyanis várhatóan ehhez hasonló lesz a következő esemény is. A dokumentáció részletesen leírja, milyen kritériumoknak kell megfelelnie a robotnak és lehetőségei vannak tájékozódni a labirintusban. Azért, hogy ne teljesen vakon közlekedjen, két segítő lehetőség is rendelkezésére áll a kiút megtalálásához. Minden zsákutcában és ahol irányváltás szükséges a legrövidebb út teljesítéséhez RFID (Radio Frequency IDentification - Rádiófrekvenciás Azonosítás) címkék vannak elhelyezve, amelyek beolvasásával megtudjuk a helyes irányt. Ezen felül minden címkén megtalálhatók piktogramok, amelyek szintén jelzik a cél felé vezető utat, ezek felismeréséhez digitális képfeldolgozás alkalmazása szükséges. A szabályzat továbbá tartalmazza az egyetem és a támogatók által biztosított alkatrészek listáját, amiken felül 25000 forint értékben vásárolhatunk saját költségén. Előre adott volt a pálya mérete (480 cm x 480 cm, 16x16 db blokk van), valamint az akkori elrendezés és az ideális útvonal a rajttól a célig, ez természetesen azóta frissült a 2024-es változatban. A verseny két részből áll, az első fordulóban minden irányváltásnál meg van adva az ideális tovább haladási irány, a második fordulóban pedig komplex utasítások alapján kell megtalálni a kiutat, ez annyit jelent, hogy egy RFID címke több egymás után következő haladási iránynak az információit tartalmazza. Minden fordulóban két-két lehetőség van a robotnak teljesíteni a pályát és megfejtetni az útvesztőt.

2.2 Eddigi versenytapasztalatok

A feladat adott volt, tudtuk, hogy milyen határokon belül mozoghatunk, ezután kezdtük el feldolgozni az előző versenyről szerzett tapasztalatokat és más csapatok gyengepontjait. Voltak olyanok, akik nem fektettek elég nagy hangsúlyt a robot méretére, ebből adódóan nehezen tudott elfordulni a kanyarokban. Rengeteg probléma akadt az ultrahangos távolságérzékelőkkel, ezért ezt is mindenképpen el szerettük volna kerülni, továbbá mindegyik csapatnál modulokat láttunk breadboard-dal és vezetékekkel összekötve. Mi ezt a megoldást nagyon bizonytalannak ítéltük. Tapasztalatok alapján a tájékozódásban nagy hátrányt jelentett a kerekek megcsúszása

irányváltásnál és a sebesség változáskor, ugyanis ilyenkor hajlamosak voltak korábban bekanyarodni, ami a falnak ütközéshez vezethet, ezt is mindenféle képpen ki kell küszöbölni. Továbbiakban mivel a jármű viszonylag nagy áramfogyasztással rendelkezik (főleg a motorok terén), kompenzálnunk kell az akkumulátorok merülésének hatását, ugyanis, ha csökken a tápfeszültség, akkor csökken a robot sebessége is, amit szintén korrigálni kell, ha nem akarunk elveszni az útvesztőben.

2.3 Specifikációk pontos meghatározása

Most, hogy lefektettük mit nem szabad csinálnia a robotnak, ideje írunk arról is, hogy nekünk mik az elvárásaink a miénkkel szemben. Az egyik legfontosabb paraméter és mindennek az alapja a teljes váz mérete, ugyanis ezen múlik, hogy korrekció nélkül a legideálisabb útvonalon célba tudjon érni. Azonban a váz nem lehet túl kicsi sem mivel, akkor kevés hely marad az elektronikai komponenseknek, ahhoz, hogy ezt elkerüljük, ki kell használnunk a rendelkezésre álló legtöbb helyet. Ennek meghatározásához építettünk egy modellt a maximum dimenziókról (280 mm x 280 mm x 150 mm) amibe bele kell férnie a robotnak és elkezdtünk kísérletezni, mi lenne az ideális méret. Úgy határoztunk, hogy amíg a jármű alapterülete 140 mm x 140 mm környékén mozog nem lesz gondunk a navigálással, ebbe a területbe bele kell sűríteni a hajtást és az elektronikát is.

A második legfontosabb szempont az egy olyan ideális hajtáslánc tervezése, amellyel maximalizálni tudjuk a gyorsulást, a sebességet és a kanyarodást. Mindenféleképpen 4 kerekes felépítésben gondolkodtunk az elejétől kezdve, azokat oldalanként meghajtva egy-egy motorral, ezzel a megoldással tudja elérni nyomaték hatására a maximális tapadást, és így lesz képes a leggyorsabban bevenni a kanyarokat. Ami pedig a sebességet illeti, mivel várhatóan minden csapatban a támogatók által biztosított motorokat fogják alkalmazni, ugyanis kellően erősnek és gyorsnak bizonyultak. Minket viszont hajt a versenyszellem, ezért a hajtásba gyorsító áttételt implementálunk fogaskerekek segítségével, ezzel is biztosítva azt, hogy nagyobb végsebességet érjünk el, mint a többi csapat. Viszont hiába van egy gyors és erős hajtáslánca a robotnak, ha nincs meg hozzá a kellő tapadási felület, amely véleményünk szerint sokaknál probléma volt. A megoldás négy egyedileg tervezett kerék, amelyen olyan gumi felület található, amely nagyobb tapadási súrlódási erő kifejtésére alkalmas, mint amiket versenytársaink használtak.

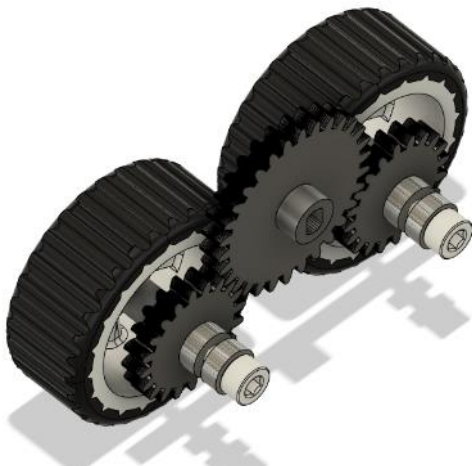
A következő szempont a pontos tájékozódás volt, milyen szenzorok segítségével tudja biztonságosan bejárni a legideálisabb útvonalat a legrövidebb idő alatt. A kiút megtalálásához mindenféleképpen az RFID olvasó mellett döntöttünk, mivel ezt gondoljuk a tájékozódás leggyorsabb és legegyszerűbb módjának, ugyanis nem dolgoztunk még digitális képfeldolgozással. Ahhoz, hogy a jármű épségben célba érjen tudnia kell navigálni a szűk falak között is, ahogy azt a 2.2 fejezetben is leírtunk. Az előző versenyen a csapatok nagyrésze ultrahangos szenzort használt, amellyel adódtak problémák. A kihívás győztese infravörös fényen alapuló távolságérzékelőket használtak, amelyeket jóval stabilabbnak bizonyultak, ezért mi is ezen a technológián alapuló szenzorokat fogunk használni, összesen három darabot, előre és a két oldalra. Ezekkel az eszközökkel most már tud közlekedni önállóan a labirintusban, viszont nem tudjuk, hogy mekkora távolságot tett meg, például egy hosszabb egyenes szakaszban. Itt nem érdemes eltelt időn alapuló távolság mérést alkalmazni, mivel ez csak állandó sebességnél lehetséges, ez pedig nem garantált az akkumulátor feszültségének folyamatos csökkenése miatt. A megoldás a problémára a kerekek vagy a motorok elfordulásának mérése, ezt rés-opto kapukkal és enkóder tárcsákkal fogjuk kivitelezni. A fenti megoldásokkal viszont még mindig nem sikerült elérnünk a jármű hiba nélküli szabályozását, ugyanis a kerekek akaratlanul, de még mindig megcsúszhatnak gyorsulásnál vagy lassulásnál, ami kedvezőtlen tényező. Ha egy gyorsulás szenzort is implementálunk a robotba, akkor tudjuk detektálni mikor is történik tényleges gyorsulás. A mért adatokat össze tudjuk hasonlítani a motor vagy a kerék forgásával, és ha nincs valójában sebességváltozás csak a fordulatszám változik, akkor arra következtethetünk, hogy az a kerék megcsúszott.

Ezek után már csak egy kihívás maradt hátra, az egész kapcsolást megvalósítani úgy, hogy elférjen a 140 mm x 140 mm alapterületű roboton. Már az elejétől kezdve tudtuk, hogy minimalizálni szeretnénk az előre elkészített modulok számát és kerülni fogjuk a breadboard-os és jumper-es megoldásokat is, mivel mozgás közben a kontaktusok nagy eséllyel meg tudnak szakadni, ezért egyből nyomtatott áramkör alkalmazásában gondolkodtunk. Azt is tudtuk, hogy a rengeteg alkatrészt kell majd elhelyeznünk kis helyen ezért kétrétegű NYÁK-ra lesz szükség. Volt olyan ötletünk, hogy a robot fizikai vázát egy vagy két NYÁK lemez alkossa, és ehhez legyen rögzítve a teljes hajtást is, azonban ezt az ötletet gyorsan elvetettük. A végső terv az lett, hogy a 3D nyomtatott vázhoz egymás fölé két darab kétrétegű áramköri lemezt helyezünk ezzel is maximalizálva a kihasználható felületet az alkatrészeknek. Ezt részletezzük a 3. és 4. fejezetben.

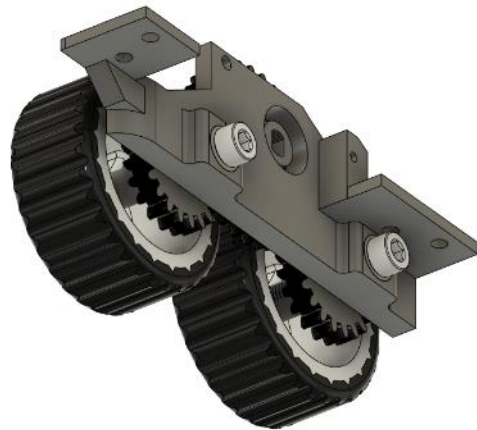
Végül, de nem utolsó sorban, abban is meg kellett egyeznünk, hogy milyen mikrovezérlőt használjunk. Itt három szempont között kellett megtalálni az egyensúlyt, hogy melyiket tudjuk a legmagabiztosabban használni, melyiknek van elegendő GPIO portja (General Purpose Input and Output - Általános felhasználású ki- és bemenet) az általunk használt szenzoroknak és hogy melyiknek van meg a kellő hardware-es tudása. A választásunk a NodeMCU ESP-32-re esett, a 4. fejezetben részletezzük, hogy pontosan miért.

3. 3D terv elkészítése [1][2][3][4][5]

Ahogy azt a specifikációban is részleteztük, az ideális alapterület mérete 140 mm x 140 mm, ez volt a kiindulási pont. Elsőként a kerekek és a hajtás ideális helyét kellett megtalálni a vázon belül. Figyelembe vettük, mechanikai szempontból, hogyan lehet a legkompaktabb, mégis leghatékonyabb rendszert megalkotni, a helytakarékosság érdekében. Mivel mind a négy kereket meg szeretnénk hajtani, ezért alaphoz négy motorra lenne szükségünk, viszont ezt el kellett kerülnünk a motorok méretei miatt¹. A négy kerék meghajtását tehát két motorral kellett megoldanunk, erre az alábbi [3.1./3.2.], a nyomaték szétosztására készült fogaskerék rendszert alkottuk meg.



3.2. ábra, Fogaskerekek elrendezése



3.1. ábra, Hajtáslánc rögzítése a vázhoz

Bár a kép már egy továbbfejlesztett változatát mutatja, az alapkoncepció ugyanaz maradt. A középső, nagyobb méretű fogaskereket hajtja meg a motor a téglalapos furaton keresztül. A kisebbek húsz, a hajtott pedig harminc darab foggal rendelkezik, ez által egy három a kettőhöz arányú gyorsító áttételt kaptunk. A fogaskerekeket az STLGears.com oldal segítségével generáltattuk, a bemeneti paraméterek az alábbiak voltak:

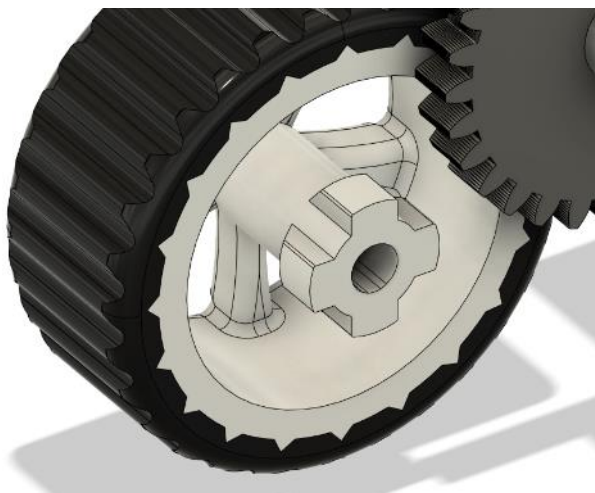
¹ Azt, hogy pontosan milyen motorokat használunk és, hogy miért azt bővebben a 4.2-es fejezetben részletezzük.

	Modul	Fogak száma	Terhelési szög	Fogaskerék szélessége
Kisebb	1.25	20	20	5
Nagyobb	1.25	30	20	5

Az elején aggódtunk, hogy a motornak nem lesz elég nyomatéka a kellő gyorsuláshoz, azonban erre a későbbi tesztelések során rácáfoltunk. A [3.1, 3.2] képen megfigyelhető, hogy a kerekeknek a vázhoz való rögzítéséhez dupla csapágyazást használtunk, ezzel biztosítva a tengelyek stabilitását. A hajtást 5 milliméter átmérőjű metrikus csavarokkal rögzítettük. A nyomatékot a fogaskerékről a [3.3, 3.4] képen látható módon vittük át a kerékre így biztosan nem tud külön elfordulni a két alkatrész.



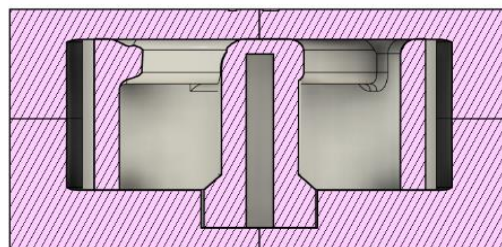
3.4. ábra, A fogaskerék illeszkedési pontja



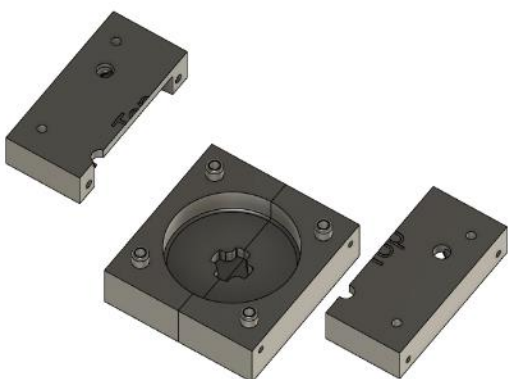
3.3. ábra, A kerék illeszkedési pontja



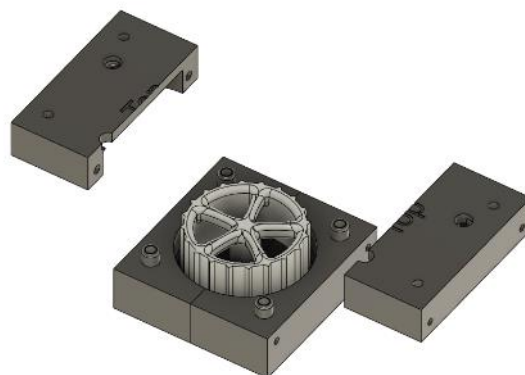
3.5. ábra, Az öntőforma



3.6. ábra, Az öntőforma keresztmetszete



3.7. ábra, Öntőforma elemeire szedve



3.8. ábra, A felni illeszkedése az öntőformához

A felni modelljéből két változat is készült mivel voltak kétségeink az egyik módszer működéséről. Ahogy a specifikációban is említettük nagy hangsúlyt kell fordítani a kerekek tapadására, ezért is volt az első ötletünk az, hogy a gumiabroncsot mi készítjük el, szilikon öntési technológiával. A szilikon egy olyan puha, jól tapadó nagy felületet tud biztosítani, amellyel határozottan tud közlekedni a robot az útvesztőben, nem kell aggódnunk a megcsúszás miatt. Az abroncs készítéséhez 3D nyomtatott öntőformát [3.5-3.8] használunk.

A felni külsőjén egy bordázott felületet hoztunk létre, ezzel megnövelve a kerék felületét, amihez így jobban hozzátapad az öntvény és a talajra levitt forgatónyomaték tépő hatása kisebb eséllyel szakítja el a műanyag felnitől. Az FDM 3D nyomtatási technológia egyik mellékhatása, hogy a felület az egymásra épülő rétegek találkozásainál rücskös/érdes lesz. Esetünkben ez kifejezetten

előny, hiszen van ahová tapadnia a szilikonnak. Az öntőforma tetején három nyílás található, ezek az öntőanyag bevitelére alkalmasak, ezen felül a nem kívánt levegő ezeken keresztül tud távozni.



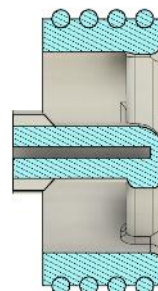
3.6. ábra, Alternatív kerék



3.5. ábra, Alternatív kerék



3.8. ábra, O gyűrűzött felni

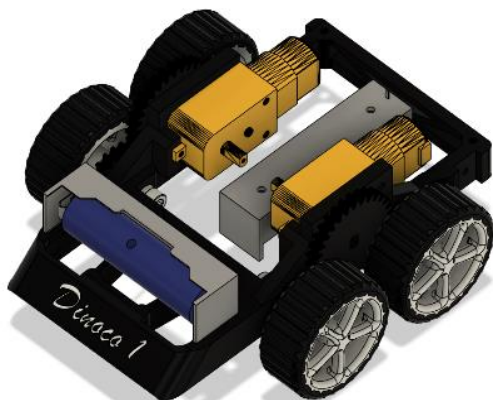


3.7. ábra, Keresztmetszet a kerékről

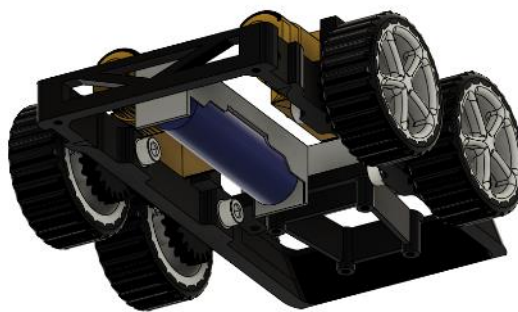
Amennyiben ez a módszer kudarcba fullad, készültünk egy egyszerűbben kivitelezhető változattal is. Itt a tapadó felületet szigetelő gumigyűrűkkel (O gyűrű) hozzuk létre, amik rá feszülnek a felnire. Az O gyűrűk optimális felnire való feszülése érdekében itt a gyűrű átmérőhöz képest 5%-kal túlméreteztük az O gyűrű fészket a felnin. Mivel ebben az esetben kisebb a tapadási felület és keményebbek a gumik is ezért az első változat elkészítésével kezdtük.

Most, hogy tudtuk, mekkora hely szükséges a hajtásláncnak nekiálltunk a váz megtervezésének. Már pontosan tudtuk milyen alkatrészeknek kell helyet biztosítani a roboton belül, viszont nagy dilemma volt hogyan helyezzük el azokat. Mivel elég közel voltak a kerekek egymáshoz a jó kanyarodás érdekében, aggódtunk, hogy borulékony lesz, ezért igyekeztünk a súlypontot minél inkább a váz közepéhez és a földhöz közel elhelyezni. Az ilyen szempontból a kritikus elemek, az akkumulátorok és a motorok, a nagy tömegük miatt.

Ennek a megoldására az alábbiképpen helyeztük el a kritikus alkatrészeket:



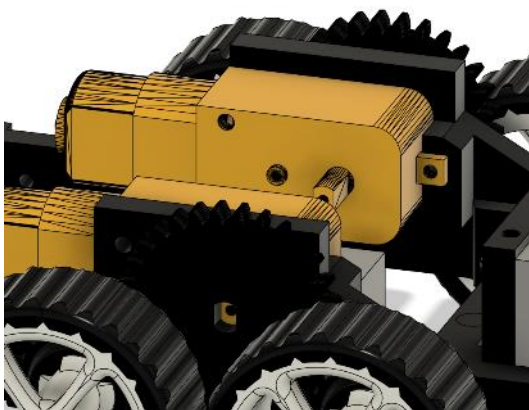
3.10. ábra, Váz felülnézet



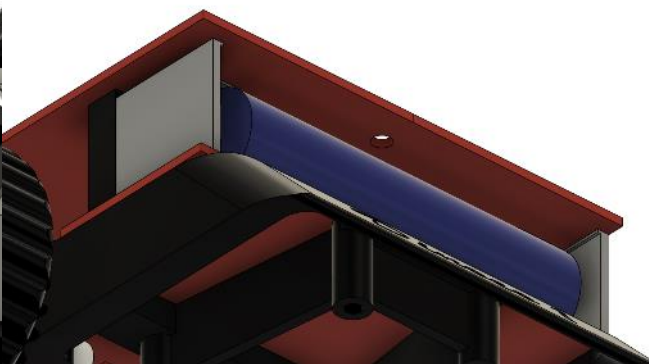
3.9. ábra, Váz alulnézet

Utólagos tapasztalatok alapján hatásosnak bizonyult ez az elrendezés.

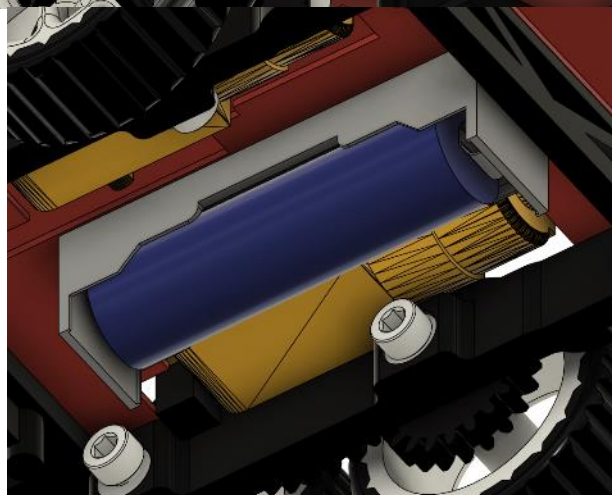
Miután ezzel megvoltunk, rögzíteni kellett ezeket az elemeket a vázhoz. Igyekeztünk mindenhol 3 milliméteres metrikus csavarokat használni, hogy egységes legyen. Abból a szempontból is döntöttünk így, mivel ez nagyban megkönnyíti az összeszerelést és a továbbiakban a szervizelést, ha szükséges. A motorokat a kaszni két oldalához rögzítettük süllyesztett fejű csavarokkal, hogy ne ütközzön a fogaskerekekkel. A jármű orrában elhelyezkedő akkumulátor



3.12. ábra, Motorok rögzítése



3.11. ábra, 1. akkumulátor helye

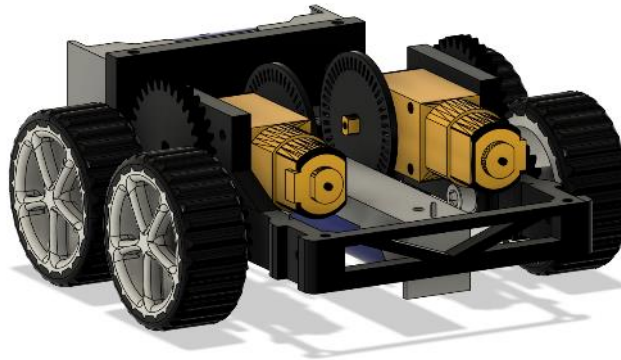


3.13. ábra, 2. akkumulátor helye

foglalatát egy olyan 3D nyomtatott alkatrészhez fogatjuk hozzá, amely összefogja a felső és alsó nyomtatott áramköröket. A hátsó akkumulátor tartót pedig az alsó NYÁK alsó felére kerül felerősítésre.

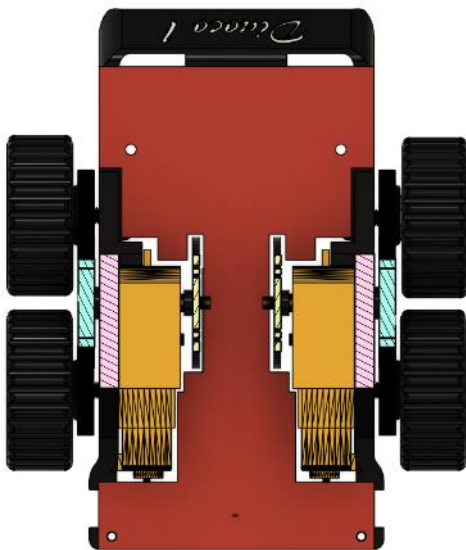
A motorok elfordulását detektáló optikai berendezéshez szükséges érzékelő tárcsa tervezése következett, mivel nagy mérete miatt ez is befolyásoló tényező volt a végső NYÁK illesztéséhez. A tárcsa szélén 46 darab furat van egymástól egyenlő távolságra elhelyezve, ez bekerül egy rés-opto szenzor elé, amely akkor küld jelet, ha a benne lévő fototranzisztor és

fotodióda “látja” egymást. Ennek alapján, amíg a motor megtesz egy teljes fordulatot, az érzékelő 46 impulzust küld a mikrovezérlőnek.

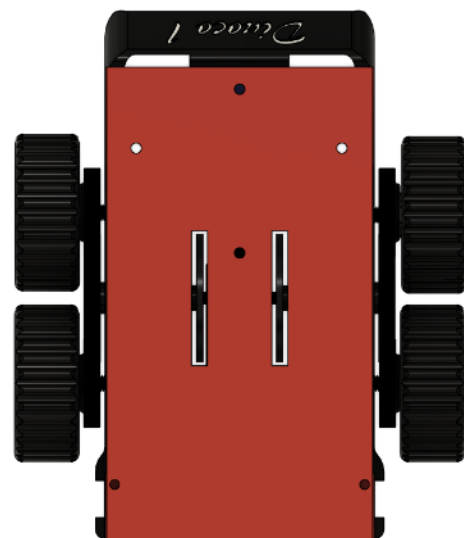


3.14. ábra, Opto tárcsák rögzítése

A következő folyamat az volt, amikor a teljes áramköri felületet be tudtuk már rajzolni a 3D tervünk be. A hely szűkossége miatt, minden rendelkezésre álló felületet ki kellett használnunk, ezért is lett egy nem éppen átlagos formája az alsó NYÁK-nak, viszont ez elengedhetetlen volt.

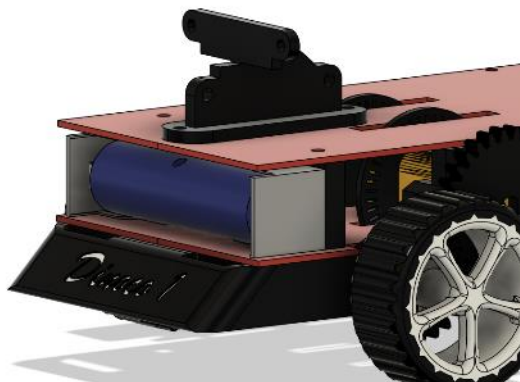


3.16. ábra, Alsó NYÁK lemez

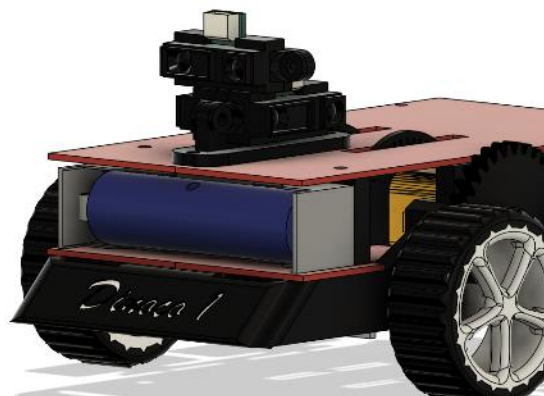


3.15. ábra, Felső NYÁK lemez

A felső áramköri lapra helyeztük el az infra távolságérzékelőket. Mivel találtunk a Grabcad.com nevű oldalon pontos modellt az érzékelőkről, így könnyen tudtunk tervezni hozzájuk egy tartó konzolt, amin elfér mind a három. [1]

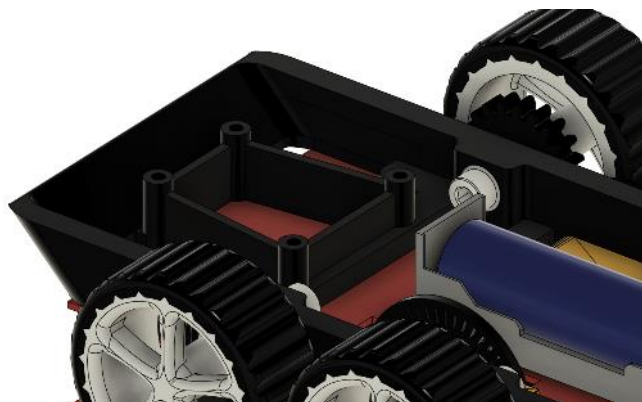


3.18. ábra, Szenzor tatró elem

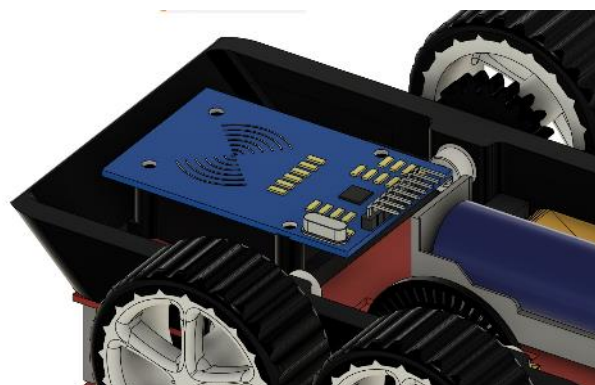


3.17. ábra, Szenzorok helye

Az RFID olvasót egyértelműen a robot alvázán kellett elhelyezzük, mivel a beolvasandó címkék a földön helyezkednek el. Az érzékelőt a lehető legelőrébb tettük, így hamarabb kapunk információt a következő útvonalról. A modulon négy furat található ezt használtuk a rögzítéshez, itt is 3 milliméteres metrikus csavarokat alkalmaztunk.



3.19. ábra, RFID olvasó tartóeleme

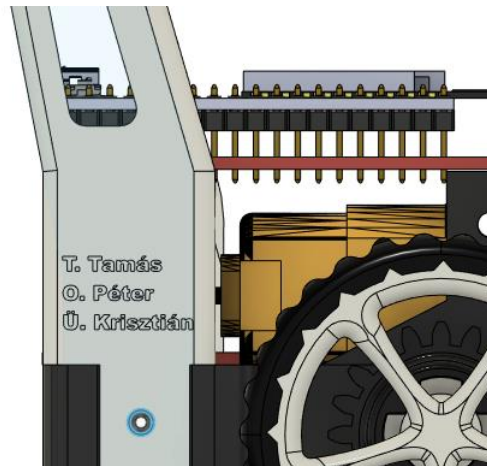


3.20. ábra, RFID olvasó helye

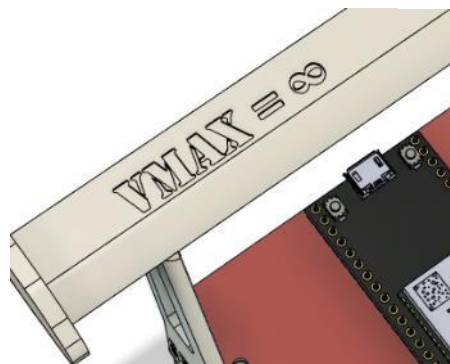
Kiemelkedő hangsúlyt fektettünk a jármű megjelenésére is, ezért is terveztünk a robot elejére és hátuljára is légterelőket. Bár mérhető lezorító erő generálására nem alkalmas, de tökéletesen leolvashatjuk róla az autó nevét, a tervezőket és a maximális sebességet.



3.22. ábra, A robot végleges terve



3.23. ábra, Az alkotók



3.21. ábra, Maximális sebessége

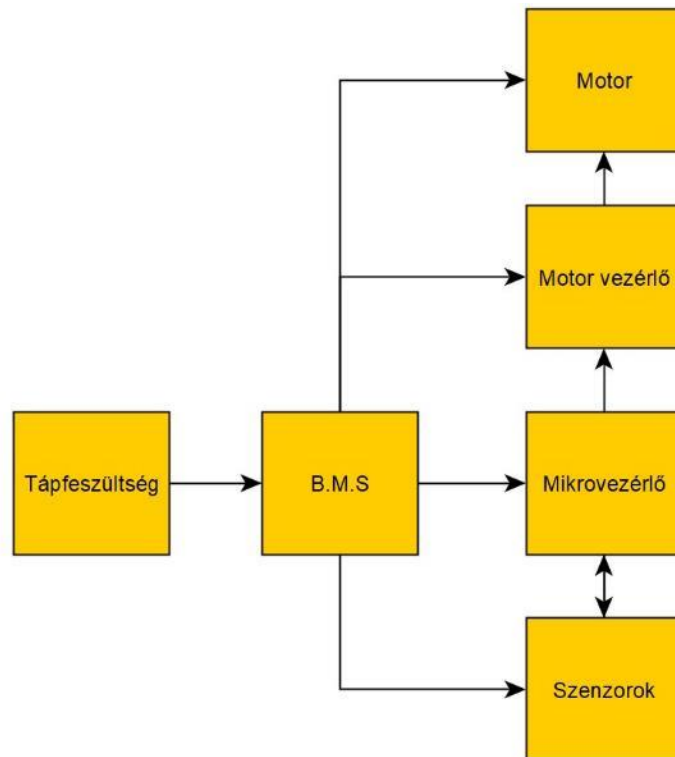
A 3D tervezéshez az Autodesk által készített Fusion (régebbi nevén Fusion 360) nevezetű programot használtuk, ennek is a tanulók által ingyenesen elérhető verzióját. Azért esett erre a választásunk, mivel már előzetesen többször is dolgoztunk vele, ezen felül, olyan eszközkészlettel rendelkezik, amely jóval meggyorsította a tervezési folyamatot.

4. Elektronika

A labirintus robot megfelelő működéséhez elengedhetetlen a hibátlan, biztos elektronikai háttér, amely a robot specifikációjában meghatározott igényeket kielégíti, és ami a verseny szabályzatának megfelel. A munka kezdetén az alap specifikációk meghatározásával konkrét elvárásokat hoztunk létre a kapcsolás és a nyomtatott áramkörrel szemben, ami a szerkezeti elemekkel összhangban készült. Ezeknek az okait már a tervezési folyamatban kifejtettünk.

4.1 Logikai rendszerterv

A legelső specifikáció megírása után figyelembe vettük a robot működtetéséhez nélkülözhetetlen egységeket és ezeket csoportosítva létrejött a következő rendszerterv:

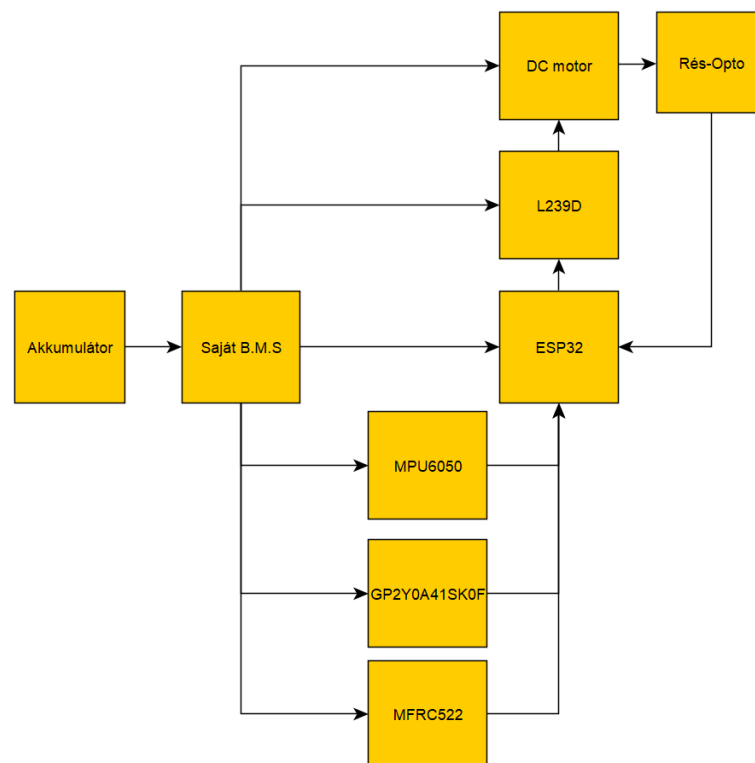


4.1.1 ábra, Logikai rendszerterv

A robot méretei és feszültség, áram igényei miatt kisebb akkumulátor alkalmazása mellett döntöttünk. A következő nagyobb elem a B.M.S (Battery Management System), akkumulátor vezérlő egység. A többi elem védelme és az akkumulátor érdekében kell alkalmazni. A szenzorok a robot közlekedéséhez, annak térbeli tájékozódáshoz szükségesek, amely az általa mért fizikai mennyiségeket villamos jellé alakítja át. A mikrovezérlő szerepe az egyik legfontosabb, hiszen az érzékelők által leadott információt az dolgozza fel és a motorvezérlőn keresztül mozgatja a robotot.

4.2 Fizikai rendszerterv

Az előző alfejezetben kifejtett logikai rendszerterve alapozva a következő rendszerterv készült el.



4.2.1. ábra, Fizikai rendszerterv

A blokkdiagramon a négyzetek belsejébe az egységeket megalkotó főelemek kerültek behelyettesítve végső konkrétumokkal. A verseny szabályzatban leírt költségkeret a robot fejlesztési alapjaira komoly befolyással volt. A szenzorok minőségén nem spóroltunk, modulok és

előkészített fejlesztőpanelek elhanyagolása miatt, saját elektronika létrehozásának anyagköltségét magunk fedeztük.

A roboton alkalmazott feszültségforrásnak, két darab újratölthető 18650-es lítium-ion akkumulátor. Erre a típusra, fizikai méretei és névleges feszültségének értéke miatt esett a választás. Az akkumulátorok túlmerítési feszültség elérését és védelmi funkciókat betöltő szerepet saját fejlesztésű áramkör teszi ki a B.M.S (Battery Management System) egységben. Az egységeket összehangoló vezérlő szerepében NodeMCU-32S ESP32 választottuk. Rengeteg analóg bemenete, kompakt, ugyanakkor nagy számítási kapacitással rendelkezik. Programozás szempontjából fontos volt, a C-ben nyelv kompatibilitása és tesztelési szempontok miatt a Bluetooth képessége is. Motorvezérléshez L293D cél integrált áramkört választottunk egyszerű PWM vezérelhetősége, DIP-es tokozása, emellett a motoroknak kellő áramot képes leadni. A verseny dokumentáció alkatrész listájában megtalálható DC motorokat választottuk. Saját mérések és tapasztalatok alapján kellően erősnek bizonyult, nagy nyomatékkal rendelkezik, hogy a robotot “nagy” sebességgel hajtsa. Tervezési cél volt, hogy a lehető legjobban csökkentsük a külső kész “pajzsok” alkalmazását, egyedüli kivételt a szenzorok terén tettünk. Azoknak tervezési bonyolultsága és a mérések pontossága érdekében alkalmaztuk őket.

Az alkatrészek és a modulok kiválasztásánál számításba kellett vennünk a korlátozott költség határt, ezt igyekeztünk maximálisan kihasználni. Ahogy a csatolt költségvetési listában is látható, keret nagy részét az infra szenzorokra költöttük, ezen nem akartunk spórolni.

4.2.1 18650 Lithium-Ion akkumulátor

A robot tápellátásának megválasztása nagy fejtörést vont maga után, hiszen nem csak dimenzió beli elhelyezése, hanem az elektronikát meghajtó áram és feszültség igények kielégítésének is eleget kell tennie.



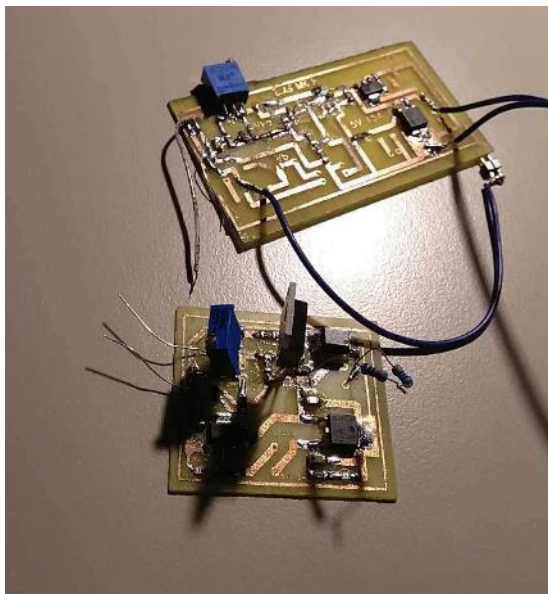
4.2.1. ábra, 18650 akkumulátor

A végső választás az általános hobbi projektek és hordozható eszközök erejére esett, a 18650-es akkumulátorra. 18,4 mm átmérőjével és 65,2 mm szélességű dimenziójú, széles választékban kapható tartó típusokkal könnyedén a robot szerkezeti váz tartó elemévé tudtuk integrálni. 3,7 V névleges feszültsége és 1,3 A áram terhelhetőségéből adódóan kiválóan alkalmazható járművünkbe. Rendszeres terhelésre és feltöltésre szánt. Egyetlen gyenge pontja a túlmerülési feszültség elérése, hiszen 3 V érték esetén, az akkumulátor sérül és már nem lesz újratölthető.

A robot elektronikájának megfelelő optimális kiszolgálásának érdekében, figyelni kell a maximális áramkorlát túllépésének elkerülésére és az állandó merülési feszültségre.

4.2.2. Akkumulátor vezérlő egység

Minden alkalmazott eszköznél, amely használ valamilyen akkumulátoros ellátást, igényli maga az elektronika és önmaga védelmét. Erre a célra a piacon számtalan kész, előre gyártott céláramkör vásárolható, egyes akkumulátorok már tartalmazzák ezt a védelmet. A robot elektronikai tervezés során figyelembe vettük ezeknek a kész produktumoknak a lehetőségét, de más utat választva saját fejlesztésű áramkörrel oldottuk meg a problémát.



4.2.2. ábra, BMS teszt áramkör

A kapcsolat oka, hogy védelemmel ellátott akkumulátor vásárlása a keretet nagyon kitolva volna, emellett tartalmaz olyan funkciókat, amelyek elhanyagolhatók a robot működéséhez. Szerepe, hogy a bemeneti feszültség figyelésével a többi elektronikai részegységet lekapcsolja, abban az esetben, ha az akkumulátorok elérik a merülési feszültségi értéket emellett rövidzár és fordított polaritási védelmet biztosítson. A használt kapcsolat módosított alapja első félévben készült projekten alapul, amely a “Kreatív megoldások a villamosmérnöki szakmában” tárgy év végi projektre készült. Annak a módosított verziója került bele a robot védelmi körébe.

4.2.3. NodeMCU-32S ESP32

A robot vezérléséhez az elektronika és a szenzorok által gyűjtött adatok feldolgozásához nélkülözhetetlen mikrokontroller alkalmazása. Erre a feladatra az interneten népszerű könyvtárral rendelkező ESP32 beágyazott rendszert választottuk.



4.2.3. ábra, NodeMCU-32S ESP32

A modul ESP32-D0WDQ6 chip-el felszerelt, kétmagos processzorral rendelkező IoT alkalmazásra tervezett vezérlő. Külső perifériákkal könnyen integrálható, Bluetooth és Wi-Fi kommunikációra képes, magas számítási feladatok elvégzésére alkalmas, alacsony áramfogyasztással.

A tervezés során felmerült, hogy a NYÁK-on nem modul formájában alkalmazzuk, hanem a chip-et körül építjük a kapcsolásban. Erről az általunk készített NYÁK határai miatt kellett lemondanunk.

A versenyszabályzatában a vezérlőkről külön fejezet található, melyben megfogalmazzák, a választási lehetőségek korlátjait. Hardveres szempontok mellett figyelembe vettük a probléma programozási oldalát, és erősségeinknek megfelelően döntöttünk emelelt. Mivel a robot navigációja és hajtásláncának vezérlése mind időkritikus feladatok, ezért döntöttünk a freeRTOS alkalmazása mellett, aminek működését az 5.1-es fejezetben részletezzük. Ezzel a megoldással optimálisan ki tudjuk használni az ESP32 hardver számítási teljesítményét.

4.2.4. Motorvezérlő



4.2.4. ábra, L293D

A robot hatékonyságát befolyásoló egyik legnagyobb tényezője a mikroprocesszor és annak felprogramozott logikája, mégis a térbeli közlekedéshez teljesen nélkülözhetetlen a motorokat vezérlő elektronika. A motorok hajtásához kész integrált áramkört alkalmaztunk, melynek feladata az ESP-től fogadott jelek hatására a két motor együttes vezérlése.

A választott áramkör a L293D, ST által gyártott DIP-es tokozású, négyszeres félhíd vezérlő. Induktív terhelések és egyenfeszültségű, léptetőmotorok vezérlésére képes széleskörű kapcsolási módokban. Széles tápfeszültség tartományban működtethető, TTL bemeneteinek köszönhetően vezérelhető magas hatásfokkal. És képes PWM-es sebesség vezérlés is. A versenyen választható “pajzsok” listájában ennek a céláramkörnek erősebb típusa volt feltüntetve (L298N), amely sokkal nagyobb munkaponti áramon és magasabb tápfeszültségen képes üzemelni. A végleges kapcsolásban azért döntöttünk a kisebb széria mellett, mert a robot forrásfeszültsége és elektronikája nem lenne képes kihasználni a motorvezérlő teljesességét, emellett az ára és foglalati típusai is csak további gondot jelentettek volna. Motornak az egyetem által biztosított csomagban szereplőket használjuk, ezeknek mérések alapján képes elegendő áramot szolgáltatni ez a vezérlő típus.

4.2.5. GP2Y0A41SK0F infrás távolságérzékelő [24]

A versenyen szereplő robot egyik kulcstényezője a szenzorok pontossága, ezért erre fektetünk az egyik legnagyobb hangsúlyt választásakor. SHARP által gyártott, az iparban széles körben alkalmazott infrás távolságmérő, ami megtalálható a legtöbb robotporszívó, robotfűnyíró és szappanadagolóban. Széles modellválaszték létezik több mérési határértékkel.



4.2.5. ábra, GP2Y0A41SK0F infra szenzor

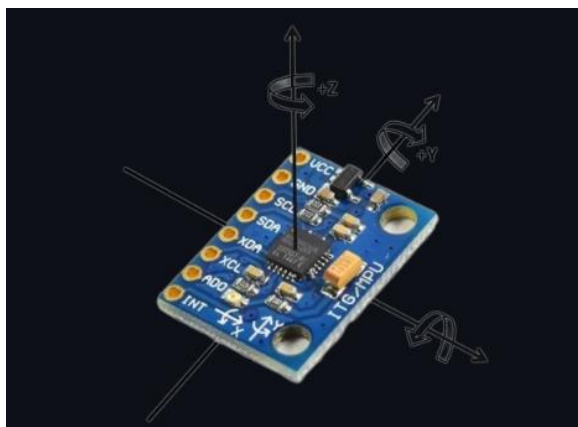
Ennek a szenzornak a legnagyobb előnye, hogy a belső felépítésének köszönhetően az általa mért aktuális értéket a kimeneten egy analóg feszültség jelként biztosítja. Széles tápfeszültség tartományban üzemelhető, belső feszültség szabályozás miatt. Fényre és hőmérsékletre minimális korrekció igényel rendelkezik a piaci versenytársaihoz képest.

A robot dimenzióihoz, és a labirintusút szélességéhez megfelelően választottuk ezt a szenzor típust karakterisztika alapján. Előzetes kalkulációkból és karakterisztikából leolvasva a munkaponti feszültségek 1.5 V - 2.7 V között fognak ingadozni a mért távnak megfelelően. {4}

4.2.6. MPU6050 gyorsulás érzékelő szenzor [15][16][25]

A robot helymeghatározásához szükséges számítások elvégzéséhez külön vizsgálni kell a gyorsulást, ennek feladatára MPU6050 IC-t alkalmazzuk. Az egyszerűség és az ár érték arány megőrzésének érdekében shield-es formájában használjuk.

Az IMU (Inertial Measurement Unit - Belső Mérési Egység) szenzor használatát azért éreztük szükségesnek, mivel a robotnak, nem igazán lenne más módja megmérni, hogy a valóságban mekkora mértékű volt az elfordulása. A legegyszerűbb megoldás időalapú utasításokban elvégezni az irányváltás. Ennek a megoldásnak a hátránya, hogy nincs figyelembe véve a kerekek kipörgése és a valóságban kevesebb lesz a megtett fordulás, ami navigációs gondokhoz vezethet. Ugyanakkor az IMU szenzornak is vannak hibái, például a gyro drift hatás. A gyorsulás érzékelő hosszabb üzemidő alatt vagy külső tényezők hatására könnyen elkalibrálódhat. Mechanikai rezonancia, hőmérsékletváltozás vagy környezeti elektromágneses

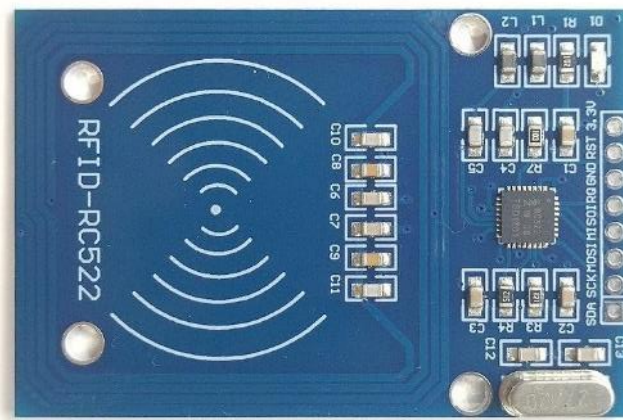


4.2.6. ábra, MPU6050

interferencia hatására fokozatosan jelentkeznek a drift, így hamis értékeket szolgáltat az érzékelő. Ezt rendszeres kalibrálással és szűrők alkalmazásával korrigálható. Ilyen például a Kalman filter, ami matematikai alapon kombinálja a gyorsulásmérő adatait a giroszkóp adataival, ezzel javítva mindkét szenzor hibáit. Ugyanakkor, ha netalán ütközne a robot valamivel, az itt keletkezett hirtelen lassulást mérve megállapítható, hogy baj van, és korrigálni lehet azt. Elgondolkodtunk azon is, hogy lehetséges lenne e kizárólag a gyorsulásokból és eltelt időből helymeghatározást létrehozni, mint ahogy azt a katonai és civil repülőgépeken is csinálták a GPS korszak előtt. Ehhez viszont a szenzor pontossága nem elég nagy és ahogy távolodunk a kiindulási ponttól, az idő függvényében az érzékelő pontossága romlik.

4.2.7. MFRC522 RFID [14][23]

A versenyen a robotok a labirintusban nemcsak szenzorok által mért adatokból, hanem a leragasztott RFID matricák beolvasásával is tájékozódhatnak. A KRESZ által használt közlekedési táblák vannak társítva HEX kódokhoz. Döntésük alapján a szenzorok által mért adatok mellé a szoftver logikájában alkalmazzuk a táblák közlekedési utasításait. Az RFID olvasó modul SPI protokollon keresztül kommunikál a mikrokontrollerrel. A szenzor kezelés tanulmányozása során

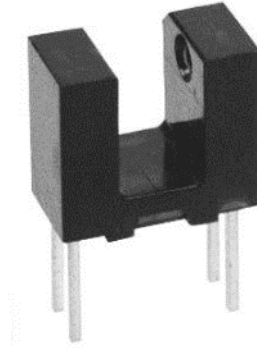


4.2.7. ábra, MFRC522 RFID

rájöttünk, hogy a matricák által olvasható HEX kódokban mindig csak 2 byte változik, így a memória spórolás érdekében a teljes hosszúságú kódokat nem kell a mikrokontrollerben tárolni. A beolvasott teljes hex kódot egy stringben tárolva csak a tényleges információ hordozó részét fogjuk összevetni az “adatbázisokkal”, hogy kinyerjük az útbaigazításhoz szükséges információkat. A modul 5 V-al működik, robot alján vízszintesen lesz rögzítve.

4.2.8. IR- rés-opto [21]

A fizikai rendszertervben megjelenített új blokk elem egy egyszerű rés-opto, melynek szerepe a motor fordulatanak vizsgálata infravörös fény segítségével.



4.2.8. ábra, Rés-opto

A mérés megvalósításához a hajtásokra külön-külön szereltünk egy tárcsát, melynek szélein megegyező távolságban elhelyezkedő lyukak találhatóak. A rés-optók a NYÁK-on úgy helyezkednek el, hogy a tárcsa széle a lyukak mentén az érzékelő középpontosan a photodióda és a phototranzisztor között tudjon forogni. A motor és a tárcsa mozgásával az opto által keltett négyszögjel vizsgálatával a vezérlőn keresztül tudjuk kiszámolni a motor aktuális sebességét és a megtett utat. Az opto phototranzisztor oldala Open-Collectoros mód almozásával lesz a vezérlőre kötve. Az adatlapon található karakterisztikák alapján, az ideális munkaponthoz egy-egy ellenállással beállítható az áram.

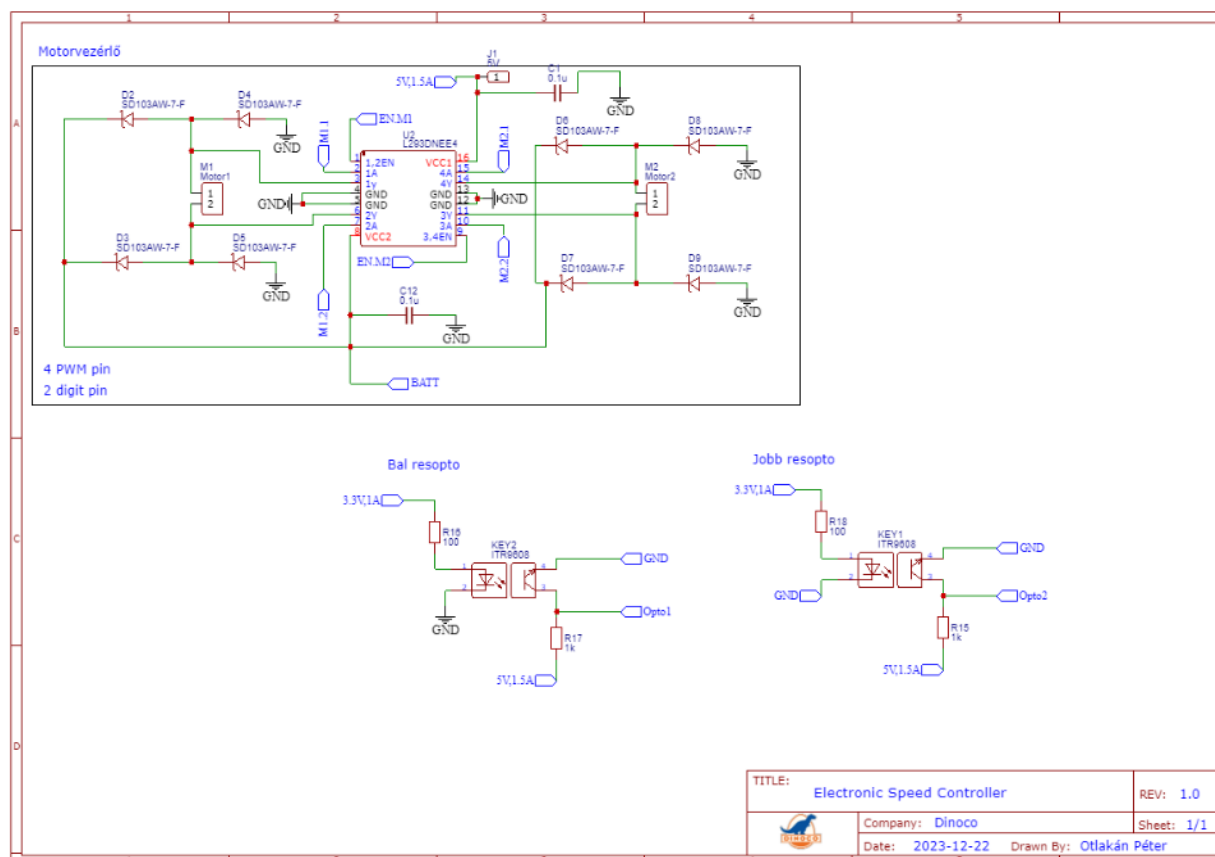
4.3 Teszt áramkörök

A közös egyeztetések és irányadó alapelvek lefektetésével elkezdtük az elméleti tervezést. Ennek első lépése az volt, hogy az összeírt rendszereknek a kiszolgáló elektronika részeit külön-külön elkezdtük fejleszteni. Ez teljesen tesztelésnek szánt, választott lehetőségek valóságos működésének igazolása és annak határainak felderítése volt a célja. Kiindulásnak a fizikai rendszertervben összefoglalt egységeket csoportosítva elkezdtük megtervezni, majd elkészíteni az első teszt NYÁK-okat. A legtöbb alkatrész felületszerelt, általánosságban 0805 méretben, SOT-223 vagy SOT-263 tokozásban.

4.3.1. Motorvezérlő és rés-optók

A fizikai rendszertervben említett L293D motorvezérlő adatlapban megfelelő kétirányú egyenfeszültségű motor kapcsolásában megszületett. Tervezési utasításokra, a motorokra a belső védelem ellenére, az ajánlott gyors válasz idejű védődiódák kerültek be, visszahatások elkerülésének érdekében. A kapcsolás az adatlapban szereplő, kétirányú dupla motoros leírásnak megfelelően készült el.

Az rés-opto Open-Collectoros formájában alkalmazva adatlapi karakterisztikájának megfelelően méretezve dolgozik a kapcsolásban. A photodioda 5 V-on dolgozik, a phototranzisztor 3.3 V-os ágon illesztve a mikrovezérlőhöz. A munkaponti áramokat előtét ellenállással állítottuk be.



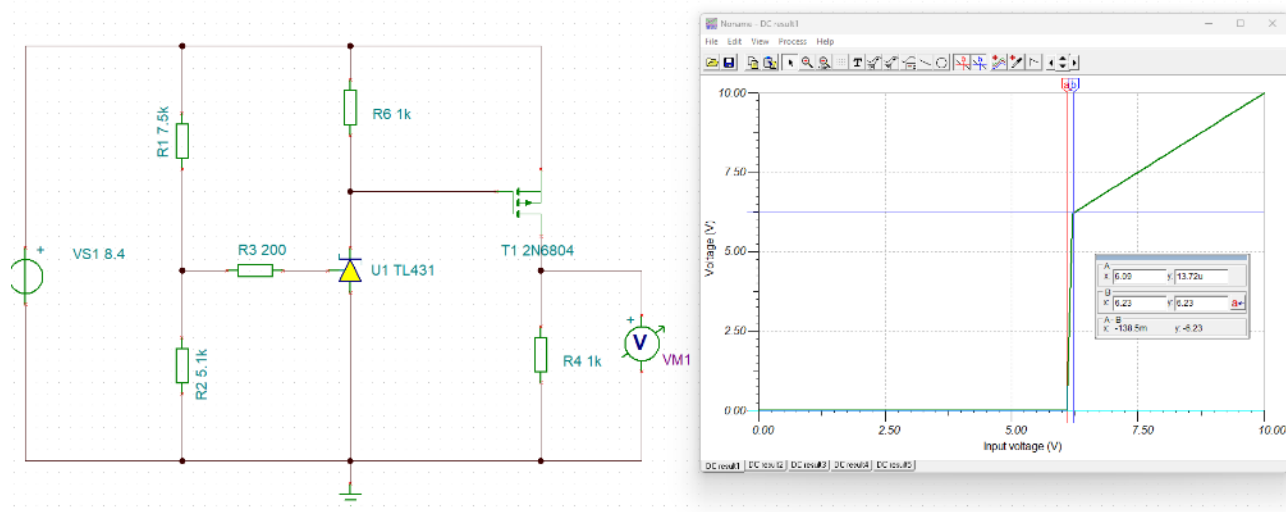
4.3.1. ábra, Motorvezérlő és rés-opto kapcsolásrajza

4.3.2. B.M.S - Akkumulátor vezérlő egység [17][18][19][20]

Az első verzióban a tápfeszültség kapcsolását pozitív oldalról egy P-s csatornájú növekményes MOSFET vezérlésével valósítottuk meg. Kapcsolóüzemű tápoknál ellenőrző körben elterjed programozható Zénert, nevén TL431-es dióda alkalmazásával figyeljük a bemeneti feszültség nagyságát. Működési előnye, hogy nagyon kevés áramot fogyaszt, így nem veszünk el feleslegesen a védelemmel az elektronika egységektől energiát. A működési elve, hogy a programozható Zéner nyitásával vagy zárásával vezéreljük a tápvonalon beiktatott MOSFET-et. A referencia lábára eső feszültség által, vezérelhető dióda Anód-Katód feszültsége, amivel a MOSFET-et tudjuk nyitni-zárni. A lehető legpontosabb működés érdekében az osztóköri, potenciométer segítségével a számolt munkapontot be lehet állítani. Az akkumulátorok merülésével a feszültség folyamatosan csökken a referencia lábon, és 2.5V-os feszültség szint beesése alatt a kimeneten a MOSFET bezár és az OPTO-nak a tranzisztor kimenete is. Mivel a LED Open-

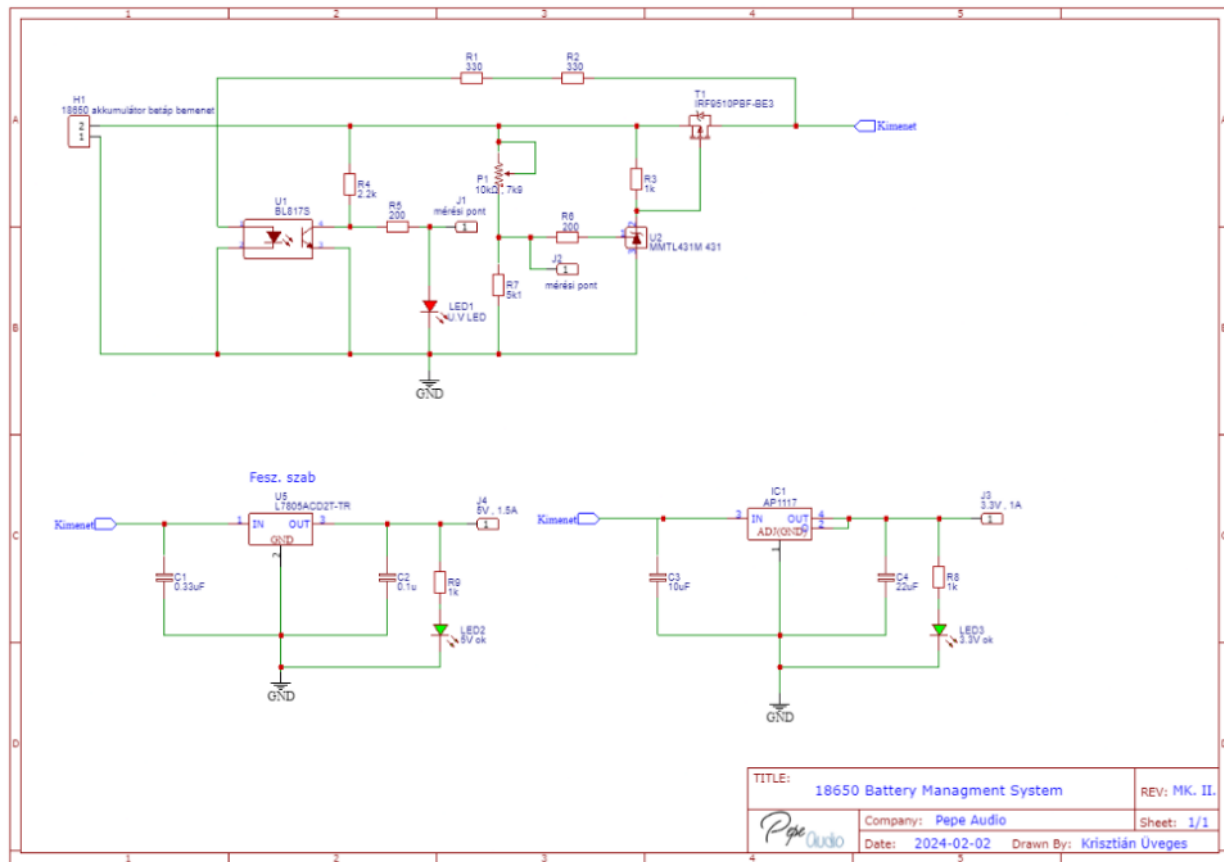
Collectoros bekötésben van alkalmazva, zárt állapotban a dióda megnyit és világít. Jelezve a felhasználónak a tápfeszültség cserélését. A tervezéshez nem volt elegendő az adatlapi információk felhasználása, a dokumentációban feltüntetett tervezési szakirodalom alkalmazásával született meg a működő verzió. A BMS-t több próbálkozás alkalmával sikerült feléleszteni. Ennek oka, hogy első verzióban nem vettük figyelembe az áram érzékenységet a diódának, ezáltal rövid használat után a munkapontok elcsúsztak és rövidzárba lépett.

A valóságban összerakott áramkör előtt, szimulált környezetben Tina 9 számítógépes programban teszteltük a kapcsolást.



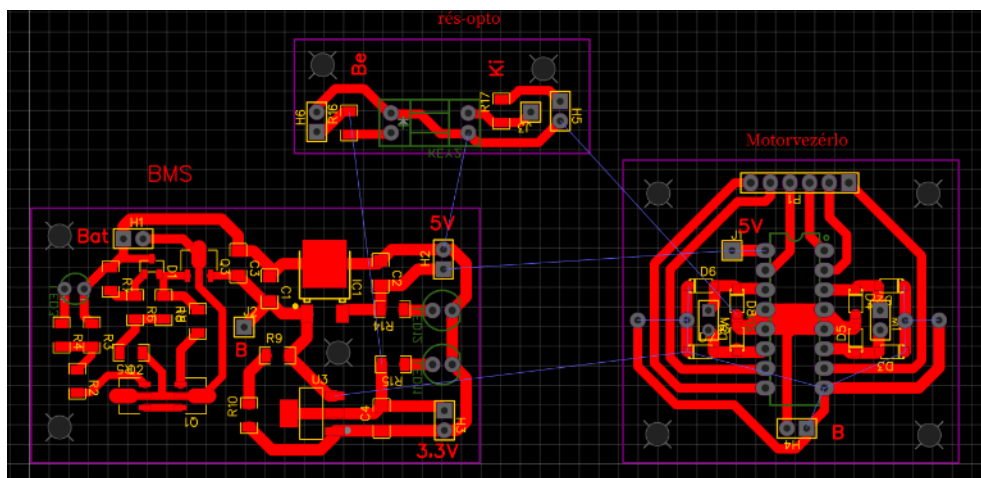
4.3.2. ábra, BMS Tina kapcsolása és a szimuláció eredménye

Annak érdekében, hogy a teljes akkumulátor vezérlőként működhessen a kapcsolásban beiktattunk feszültség regulátor integrált áramköröket. Ennek oka, hogy az elektronika többi részének biztosítsuk az 5 V és 3.3 V tápszintet, emellett ezeknek az egységek integrálásával, kimeneti rövidzár, hőmérséklet védelem funkció is belekerült a kapcsolásban. Későbbi mérések alapján az optos visszacsatolást elvetettük, mert áram szempontjából sokkal több fogyasztott a LED-es visszajelzés, mint maga a vezérlési kör.

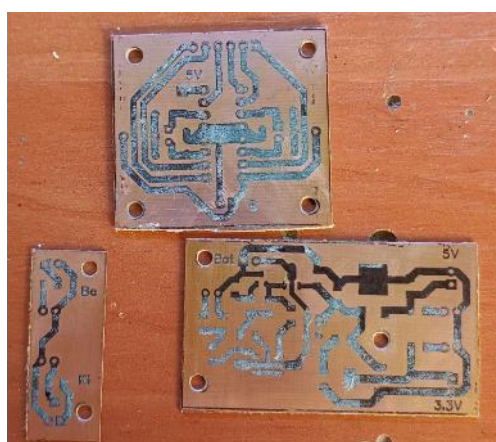


4.3.3. ábra, BMS teszt áramköre

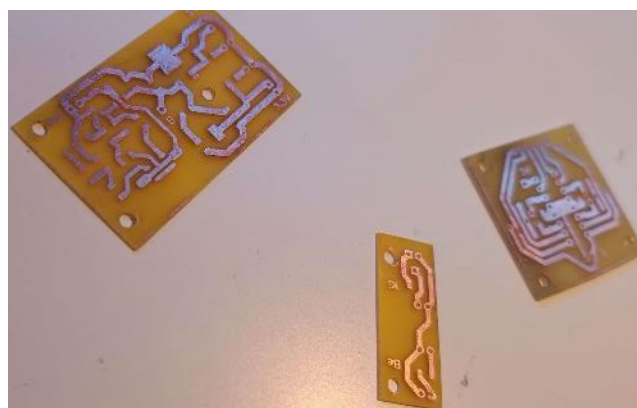
Az alapelemek megtervezése és szimulálását teszt modulok megvalósítása követte. Áramkörtervező programban a NYÁK rajzait megalkottuk. Igyekeztünk az első áramköröket úgy megalkotni, hogy megfeleljenek a robotban leírt elvárásoknak és a modularitásra, hogy a mérések könnyen kivitelezhetőek legyenek. A gyártási technológia a hagyományos maratási módon, otthoni garázs körülmények között megszülettek.



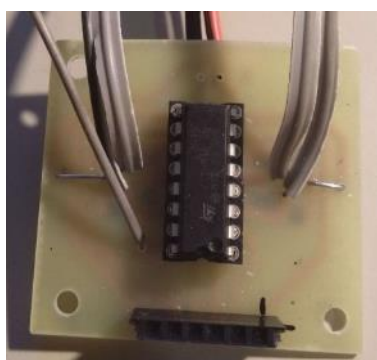
4.3.4. ábra, Teszt NYÁK-ok terve



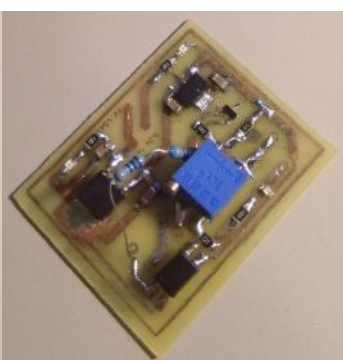
4.3.5. ábra, Vasalt teszt NYÁK-ok



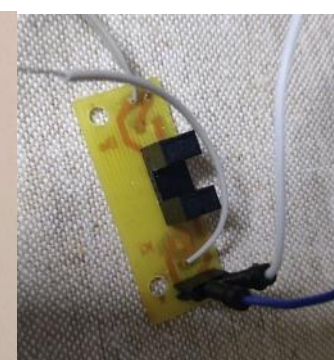
4.3.6 ábra, Maratott teszt NYÁK-ok



4.3.7. ábra, Motorvezérlő



4.3.8.. ábra, BMS



4.3.9. ábra, Opto kapu

A beültetett NYÁK-okat külön-külön teszteltük. Első tervezésre a motorvezérlő és a rés-opto az elvártak szerint hibátlanul működött. Egyedül az akkumulátor vezérlő egység produkált nagyon instabil állapotokat, ami egy teljes újra számolást és kivitelezést igényelt miután a hiba forrását megtaláltuk. A kiépített otthoni műszerpark, digitális mérések rögzítésére nem alkalmas, ezért nem tudtunk dokumentálható értékeket felvenni a projekt kezdetén.

A motorvezérlő teszteléséhez, egy Raspberry pico-t alkalmaztunk. Az első mérés célja az volt, hogy a valóságban mért adattal rendelkezünk a motorok által felvett teljesítményről és annak a sebességéről. Az integrált áramkör adatlapján található igazságtáblázatnak megfelelően a lábait magas vagy alacsony TTL jel szintre helyezve, elindultak a motorok. Maximális tápfeszültség esetén üresjárásban 200 mA, terhelt esetben hatással 450 mA mértünk.

A rés-opto mérés során az elmülethez hűen működött. Analóg oszcilloszkópon vizsgáltuk az állapot váltási idő függvényét, de olyan gyorsaságban történt az átmenet, hogy konkrét időt összevetni az adatlapon feltüntetett értékekkel nem tudtunk. A B.M.S működő verziója a szimulált karakterisztikájának egyenértékű adatokat prezentált. A kimeneti állapotjelző LED-ek alkalmazásával terhelési teszt során a működési áram 14 mA lett. Az alkalmazott szenzor "pajzsok" -ról komolyabb méréseket nem készítettünk, mert azok rendes dokumentációval rendelkeznek.

A teszt modulok mérésével kapott adatok alapján tudtunk tovább haladni az összetettebb fejlesztési folyamatokhoz.

4.4 Végző kapcsolás [22]

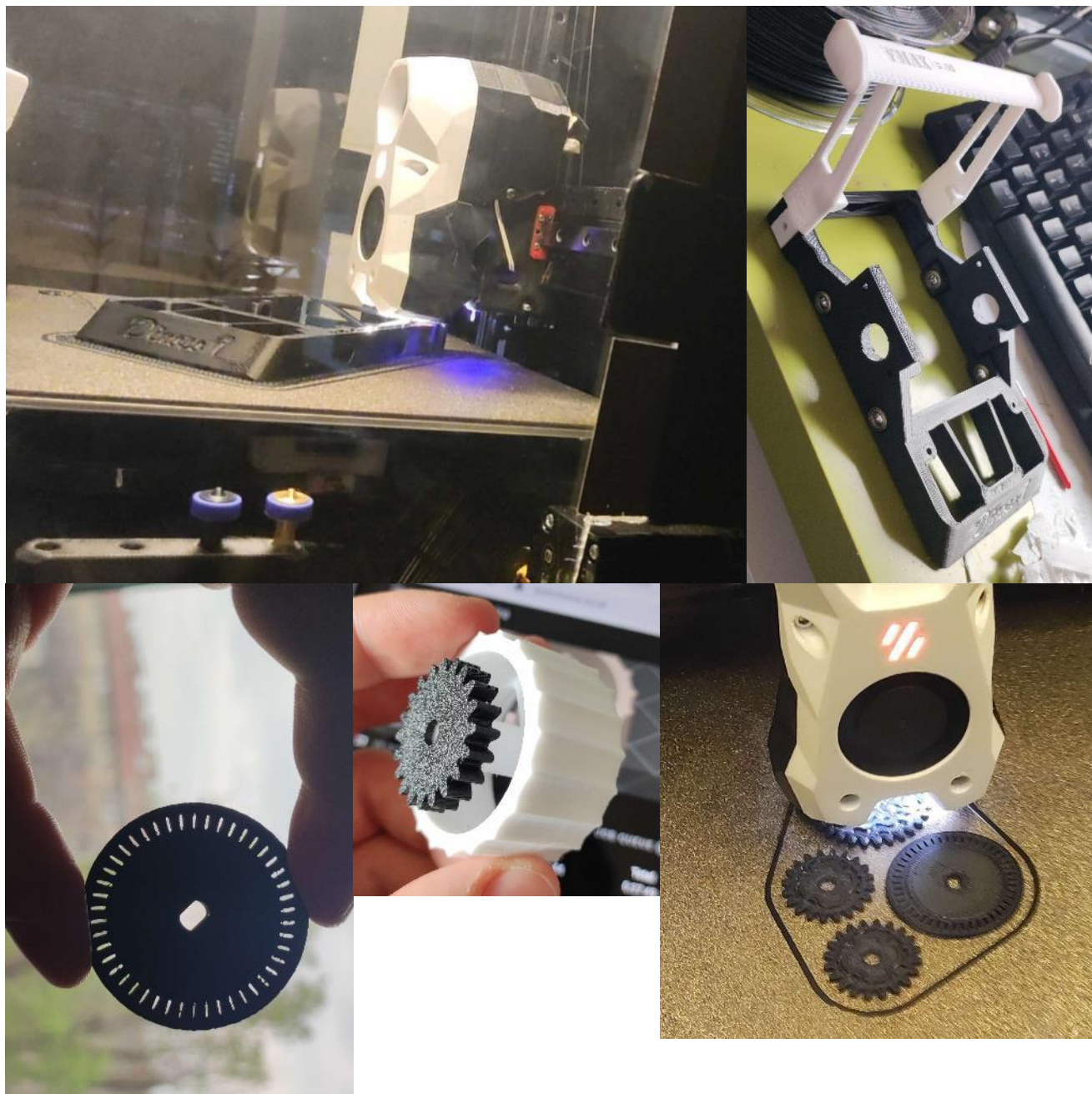
A váz és a mechanika létrejöttével, a végleges elektronika megalkotása követte. A robot igényei és minden elektronikai egységnek figyelembevételével a {1}, {2}, {3} -as csatlományokban szereplő kapcsolás született.

Végleges áramkör a teszt modulok és a mikrovezérlő összességéből áll. A szenzorok és az integrált áramkörök dokumentációjában feltüntetett tervezési javaslatok apróságaival lett kiegészítve. Tápellátás két darab sorosan összekötött 18650-es akkumulátor teszi ki, ami a bemeneti névleges feszültséget 7,4 V, 1.3 A-t tesz ki. A lemerülési feszültség 6V-ra lett méretezve, a B.M.S karakterisztikának megfelelően 6.1 V-nál már jelentős letörés történik, és a robot lekapcsol. A szenzorok, mikrovezérlők és az integrált áramkörök tápfeszültség beállításához, 5 V és 3.3 V szabályozott feszültséget állítunk elő az akkumulátor névleges értékéből. Ezt több helyen is szűrjük a felvett zavarjelek elkerülése érdekében. Az ESP32 GPIO kiosztásához igazodva a szenzorok kommunikációs adat vagy analóg jel szerint bekötöttük. A mikrovezérlőhöz csatlakoztattunk két darab nyomógombot. Funkciója, hogy vezérlési állapotokat tudjunk hozzáadni a robothoz különböző kombinációk lenyomásával. Integráltunk több, egyenként címezhető RGB LED-et, dizájn és a program logikájának látványosabb kiegészítése képpen. A WS2812 RGB LED, 5050-es chipen alapuló, speciális integrált áramkört tartalmaz, amelynek köszönhetően sorba kötve az ESP-re vezérelhetőek egy adatbuszon keresztül. A roboton két különálló NYÁK foglalja össze a teljes elektronikát. Figyelembe véve, hogy a robot szétszedhető legyen a kettő közötti kapcsolatot szalagkábelrel kivitelezük. A tápfeszültségek szállításából adódóan a felvett zavar jelek szűrésére szükségesnél több helyen is beiktattunk kondenzátorokat.

Az áramkört, a robot vázának tulajdonságaihoz igazodva és helytakarékosági szempontból két részre kellett bontani, egy alsó és egy felső NYÁK-ra. FR4-es 2 rétegű, 1.5 mm-es lemezekon valósítottuk meg a nyomtatott áramköröket. Gyártás technológia szempontból a két réteg összekötése nagyon nagy pontosságot igényelt elkészítése során. Ennek szükségessége a vezető felületek kiterjesztése és a felületszerelt alkatrészek elhelyezése készítetett minket. A feladat legnagyobb kihívását az alsó panel jelentette, mert ott volt a legkevesebb felületünk. A mikrovezérlő és az érzékelők csatlakoztatásához tűkesor aljzatokat alkalmaztunk, a motorvezérlőt DIP-es kiserelése miatt foglalatban.

5. Fizikai megvalósítás

5.1 3D nyomtatás



5.1.1-5. ábra, Alkatrészek 3D nyomtatása

A [5.1.1-5.] képen látható, hogy a mechanikai alkatrészek elkészítésére FDM (Fused Deposition Modeling - Olvasztott rétegképzés) technológiával működő 3D nyomtatót használtunk. A robot váza, a fogaskerek és az opto tárcsák ASA-ból (Akrilnitril-Sztirol-Akrilát) készült, mivel nagy merevséggel rendelkezik, jól ellenáll a hőnek és a vegyi anyagoknak. A hátsó szárny és a felhők PETG-ből (Polyethylene Terephthalate Glycol), a jármű orrában található akkumulátor tartó konzol pedig PLA-ból (Polylactic Acid) lett nyomtatva.

Ezzel elkészült minden mechanikai alkatrész, egyet kivéve, a végső gumival ellátott kereket. Ahogy a 3. fejezetben is írtuk, elsőként a szilikon gumi öntésével kezdtük a kísérletezést, de így csak egy kereket tudtunk sikeresen kiönteni, az összes többi próbálkozásunk kudarcba fulladt. Minden alkalommal műszerolajat használtunk, hogy elkerüljük az öntvény beleragadását a formába, viszont ez csak az első alkalommal működött. Akkor egy több éve kibontott tubus maradékát használtuk el, ezért is tudott könnyen megkötni. A többihez új adag kellett az viszont nem tudott megkötni az öntőformán belül és hozzátapadt a formához, így az alternatív megoldáshoz kellett folyamodnunk.

5.2 Nyomtatott áramkörök

A kapcsolási rajz végső ellenőrzése után megbizonyosodtunk arról, hogy elkezdhetjük megvalósítani a végleges áramkört. Az 1,5 milliméter vastagságú NYÁK lemezt először karos lemezvágóval vágtuk méretre, majd vasalásos technikával felvittünk a kapcsolást a nyers rézzel bevont lapra, majd nátrium-perszulfát oldattal lemarattuk a felesleges vezető felületet. A következő folyamat a végleges körvonalak elkészítése volt, hogy pontosan illeszkedjen a vázhoz,

ehhez dekopírfűrész és reszelőt alkalmaztunk. Az alkatrészlábak és az átvezetések furatait 0,8 milliméteres fúrófejjel készítettük el, multicsiszoló segítségével.

5.3 Áramkör tesztelése

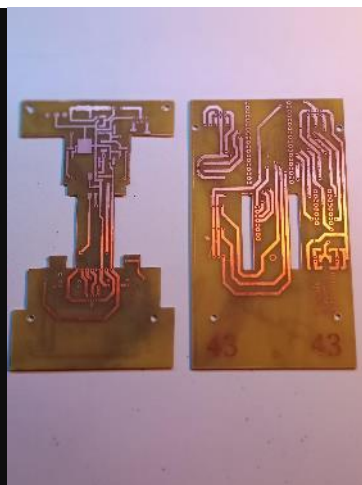
Maratás után, következett a minőség ellenőrzés és a vezető felületek összekötésének az ellenőrzése. Az első körben, mindkét rétegen és nyomtatott áramkörön vizsgálva a vezető vonalak



5.3.1. ábra, Maratási folyamat



5.3.2. ábra, NYÁK lap formára vágása



5.3.3. ábra, Kész NYÁK-ok

szélességét és a felületek épségét kellett felülvizsgálni. Ennek oka, hogy maratási technikának egyik hátránya, hogy a nagyobb panelek esetén a lemezről a számunkra haszontalan réz réteg nem egyenletesen oldódik le. Ez a gyakorlatba azt jelenti, hogy bizonyos felületek több időn keresztül van a folyadékban, ami a számunkra hasznos felületről képes lemarni a festékkal védett réteget. Inspekció után bizonyos vezető felületeket korrigálni kellett, mert a félig lemart réz réteg már nem is vezetőként, hanem kiló ohmikus nagyságrendben ellenállásként szerepelt volna a kapcsolatban. Vékony ón réteg felhúzásával vagy előőnozott 0.3 mm vezetőknek belső szálaival ráforrasztva lettek a panelek korrigálva. Második körben a rétegek közötti kapcsolat és a két nyák összekötésének pontosságát ellenőriztük. Az alsó és felső rétegeket összekötött furatokkal oldottuk meg, amiben alkatrész lábat forrasztottunk, evvel létrehozva a kapcsolatot a két réteg között. A kézi pontatlanság határain belül a furatok és vezetők elhelyezkedése egymásra nagyon pontosan fedik. Az alsó és felső rétegeken egymással párhuzamos alkatrész furatokat tettünk. Ezekben passzív alkatrész lábak lettek összeforrasztva, ami létrehozza vezetést a két oldalt között. Az ellenőrzést a felületszerelt alkatrészek beültetése követte. Ólom tartalmú ónt és folyasztószert

alkalmaztunk a pontos szép munkához, a nyákot vegyszeresen kezeltük utána, védő lakkal láttuk el.

Az áramkörök működésének vizsgálata rész egységekre bontva történt. Az akkumulátor vezérlő egység volt az első vizsgált elem. A tesztmodulban mért áramkör a roboton sem viselkedett másképpen. Az elméleti tervezésnél megalkotott karakterisztikának megfelelően viselkedett labortápról meghajtva. A pontos mérést otthoni környezetben nem tudtunk megvalósítani, mert az általunk otthon rendelkező műszerekkel nem lehet olyan felbontásban értékeket leolvasni, amivel kiértékelhető lenne a karakterisztika.

A rés-optók beültetésével a tárcsa mozgásával vizsgálható volt a phototranzisztor Collector-án létrejött jelalak. Első alkalommal analóg oszcilloszkópon mértük egyszerű objektum középre helyezésével a feszültségi szint változását. Az első tervezési hiba ekkor derült ki, mert a mért jelalakból leolvasható volt, hogy a rés-optonak a tranzisztor és dióda oldalának a táp feszültségét megcseréltük.



5.3.4. ábra, BMS tesztelése



5.3.5. ábra, BMS tesztelése

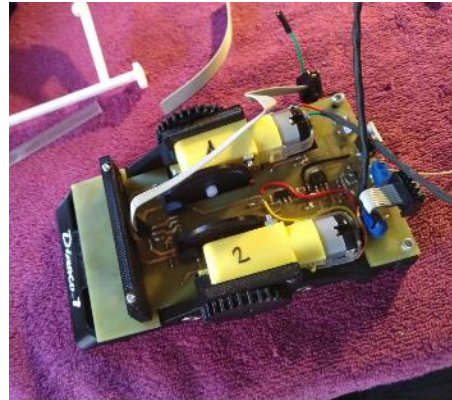
A probléma, hogy a mikrovezérlő GPIO-ja (General Purpose Input/Output - Általános Célú Bemenet/Kimenetnek) maximum bemeneti feszültsége 3.3 V táp feszültségi szintje lehet. A fordított táplálással ez az érték 4.86 V az ESP32 mentes áramkörben. Szerencsére kisebb igényes átforrasztással a tervezési hiba kiküszöbölhető. Feszültségmentes motor kézzel nem mozgatható ezért a végleges mérés és az esetleges javítási szükségesség, a mikrovezérlő teszthez felprogramozott kiegészítéssel lesz ténylegesen vizsgálható. A szenzorok, integrált áramkörök (motorvezérlő) és az RFID “pajzs” feszültségre kötésénél, vezérlő nélküli állapotban 100 mA vett fel.

5.4 Végő összeszerelés

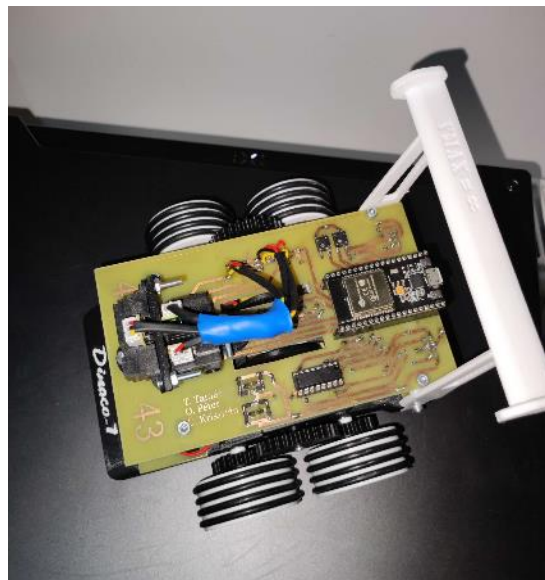
Miután megbizonyosodtunk arról, hogy az áramkör az elvártnak megfelelően működik, megkezdtünk a robot végő összeállítását. Elsőként a hajtásláncot szereltük be, ez után következett az alsó NYÁK, itt egy kicsit reszelni kellett a vázból a felületszerelt alkatrészek végett. Amint az le lett csavarozva érkezett rá a távtartó és aztán a következő áramkör. Bekötöttük a két csatlakozót, majd ellenőriztük, hogy a kommunikáció megfelelően működik a két egység között. Rögzítésre kerültek a távolságérzékelő szenzorok és a hátsó szárny is, ezzel fizikailag elkészült a DINOCO csapat mobilrobotja.



5.4.1. ábra, Váz formára igazítása



5.4.2. ábra, Alsó NYÁK és motorok beszerelve



5.4.3. ábra, A teljes robot összeállítva

6. Szoftver

6.1 RTOS ismertetése [6][7][8]

Az azonos időben való párhuzamos feladatok végzéséhez a hardveres erőforrások végeessége miatt szükség van egy operációs rendszerre. Valószínűleg meg lehetne oldani időzítésekkel is, viszont tapasztalatok alapján az ESP32-n ez a fajta megoldás nem igazán bizonyult stabilnak. A robotnak egy olyan operációs rendszer kell, ami stabil, determinisztikus, azaz mindig ugyan úgy reagál adott impulzusra, és gyorsan képes kezelni a feladatokat. A Windows itt szóba sem jöhet, mivel az az operációs rendszer a felhasználói interakcióra lett kitalálva.

Pontosan ilyen felhasználásra lettek kitalálva a valós idejű operációs rendszerek. A történelemben az első ilyen operációs rendszert az IBM mérnökei fejlesztették ki a hidegháború űrverseny korszakában a NASA Apollo programjához, és ez a szoftver vezérelte az Hold kompot, ami végül 1969. július 20-án a Hold felszínére juttatta az első embereket. Ha űrhajó vezérléséhez bizonyítottan megfelelt, valószínűleg a mi robotunk vezérléséhez is elegendő lesz.

Mivel feltalálni újból a kereket nem annyira célszerű, jelen esetben a FreeRTOS projektet használtuk fel. Ez nagyban meggyorsítja a fejlesztés menetét, mivel az operációs rendszer már adott, csupán a feladatokat kell integrálni bele és megmondani, hogy melyik feladat mennyire fontos.

Ennek az operációs rendszer lelke az ütemező. Ez az algoritmus szabja meg, hogy melyik feladat mikor és mennyi processzor időt kap. Ezt úgy kell elképzelni, mint egy forgó bakelit lemezt. A lemez szeletekre van osztva, ezek a szeletek jelképezik az idő ablakokat, ahová az ütemező be tudja szűrni a feladatokat. Implementációtól és prioritástól függően egy feladat több egymás után következő idő ablakot is kaphat. A feladatok beosztására kétfajta algoritmus áll rendelkezésre:

- Prioritáson alapuló: Mindegyik feladatnál definiáljuk a fontossági szintjét. Ha egy feladat éppen végrehajtás alatt áll, viszont felbukkan egy másik feladat, aminek magasabb a fontossági szintje, az éppen futó feladat felfüggesztésre kerül és a fontosabb feladat hajtódik végre először.
- Együttműködő: itt mindegyik feladat egyforma fontossággal bír, az új feladat csak akkor következik soron, mikor az éppen futó feladat befejeződött.

6.2 Hajtáslánc vezérlés és visszacsatolás szenzor adatok alapján

A motorvezérlőt 6 bemenet segítségével lehet irányítani. Ezek közül 2db enable(engedélyező) pin biztonsági célt szolgál, amíg ezek logikai 0-ban vannak, addig nem reagál semmi féle irányítási parancsra. Motoronként kettő PWM bemenet áll rendelkezésre, haladási irányonként 1-1 PWM bemenet.

A motorok gyártási szórásából adódóan nem biztos, hogy teljesen egyforma forgási sebességet szolgáltatnak egy adott feszültségen. Ezért is terveztünk visszacsatolást a hajtásláncba. A motorok tengelyein enkóder tárcsát helyeztünk el, amin 46 db lyuk található. Ezt egy opto kapuval olvassuk. Hogyha sebesség eltérés van a 2 motor sebessége között, az enkóder adatok alapján ez korrigálni lehet az adott motorra küldött PWM jelet. Ezt úgy lehet kivitelezni, hogy számolunk motoronként tengely forgási sebességet az adott idő alatt megszámlolt opto kapu impulzusból, összehasonlítjuk őket és az eltérés százalékát PWM korrekcióként hozzáadjuk a lassabb motor jeléhez, vagy ha maximum kivezérlés van érvényben, akkor levonjuk a gyorsabb motor PWM jeléből.

Ugyanakkor a tengelyre szerelt enkóder távolságmérő szenzor szerepét is betölti, mivel ismert a tengely forgási sebessége, ismert a végáttétel arány a fogaskerekek között és szintén ismert a kerék átmérő is, ezekből az adatokból meg viszonylag pontosan lehet megtett távolságot számolni, például a fordulópontok/ RFID-k között. Ezek az adatok majd az útvonal optimalizálásnál fognak felhasználásra kerülni.

6.3 Útvonal feltérképezés és tájékozódás [8][11][12][13]

A versenyszabályzat alapján a leggyorsabb kiút a labirintusból az, hogyha az RFID matricák által tartalmazott utasításokat követve halad a robotunk. Ha meg nincs RFID matrica, akkor egyenesen kell haladni tovább. A labirintus pálya 28x28 cm-es négyzetekből áll. A matricák a pálya közepére lesznek majd elhelyezve, így célszerű a középvonálhoz minél jobban igazítani a robotot. A közvetlen környezet érzékelését a robot orrán elhelyezett infra távolságmérő szenzor “árbóc” teszi lehetővé. A szenzorok maximális hatótávolsága 30 cm. Ez azt jelenti, hogy 1 kockával lát maga elé a robot. Egy zsákutcát úgy tudunk felismerni, hogyha 20cm-re maga előtt falat érzékel és oldalt is fal van. Ilyenkor 180 fokos fordulót célszerű eszközölni. A lehetséges “utacákat” úgy lehet felismerni, hogyha valamelyik vagy akár mindkét irányba az oldalt lévő távolságmérők hirtelen maximális értékre ugranak.

A robotot a navigáció során oldalra néző szenzor távolság adatai alapján kétféle algoritmus segítségével tudjuk középen tartani.

Az egyszerűbb alkalmazott megoldás az arányos szabályzás (proportional control), itt a két szenzor mért értékének különbsége, ez lesz a hiba értékünk, hogyha negatív, akkor balra, ha pozitív akkor jobbra kell korrigálni. A fordulás mértékét egy általunk definiált konstanssal tudjuk befolyásolni.

A bonyolultabb megoldás egy PID kontrollert implementálása lenne, így a robot sokkal rezponzívabb a gyors útvonal változásokra, például, ha a pálya tartalmaz szerpentinzerű részt. Mivel viszont a pálya egymásra merőleges egyenes szekciókból áll, nem lenne kifizetődő ebben az esetben a PID kontrollert használata.

A robot elfordulását az IMU szenzor adataival tudjuk mérni. Itt a fő gondot a gyro drift jelenti, ami a függőleges tengely körüli középpont elcsúszását jelenti. Ez a szenzor sajátos hibája. Javítani viszont lehet rajta a Kalman szűrő használatával. Ez a szűrő működése meglehetősen bonyolult, szerencsére léteznek kész implementációk könyvtárak formájában, amit használhatunk. Röviden leírva a következőképpen működik: Az algoritmusnak két szakasza van, az első a becslési szakasz. Itt ki kell számolni az aktuális állapotváltozókat és a hozzájuk tartozó bizonytalanságokat is. A második szakaszban a mérési adatokat súlyozott átlagolással kell majd figyelembe venni. A folyamatos valós idejű mérések során az átlagolás miatt egyre jobb eredményeket kapunk, mivel a zajok és más zavaró tényezők kiesnek. Ez egy rekurzív jellegű algoritmus, azaz aktuális számolt állapotot és az aktuális mért adatokat veszi figyelembe. A Kalman szűrő alapfeltevése az, hogy a

vizsgálandó rendszer az egy lineáris dinamikus rendszer és az ebben fellelhető hibafüggvényeknek normális eloszlásuk van.

Ugyancsak versenyszabályzat szerint fordulóként 2x mehet végig a pályán a robot. Ebből kiindulva két módot programoztunk a robotba. Bekapcsolásnál az első, feltérképező módban indul el a robot, ilyenkor lassabban halad, hogy véletlenül se hagyjon ki egyetlen RFID matricát se.

A beolvasott RFID tagek szerinti utasításokat egy sor típusú adat struktúrába menti el, a matricák közötti megközelítőleg mért távolságok meg szintén mentésre kerülnek. Ha a robot eléri a végső STOP matricát, akkor a mikrokontrolleren lévő LED gyors villogással jelzi, hogy megerősítésre vár. Ha úgy gondoljuk, hogy jó útvonalon ment végig, a felső nyákon található 2 gomb egyszerre történő legalább 1 mp-ig tartó nyomtatásával tudjuk menteni az útvonalat (ez csak addig lesz tárolva, amíg áram alatt van a robot). Ezzel egyben a második üzemmódba is kapcsoljuk a robotot.

6.4 Feltérképezett útvonal optimalizálás, végső gyors kör

Végül, a feltérképező út alatt gyűjtött távolság adatokból számítható szükséges féktáv. Ez azért fontos, mert így a két fordulópont közötti utazási idő minimalizálni tudjuk az elérhető legnagyobb sebességgel való haladással, a várható kanyar előtt meg simán lassítjuk biztonságos kanyarodási sebességre. A kerék szenzornak köszönhetően nem csak távolságot, hanem sebességet is tudunk mérni a roboton. A robotnak szükséges féktávot, maximális kanyarodási sebességet és maximális gyorsulást csak tesztelesek során tudjuk kideríteni. Viszont az ezek ismeretében a második üzemmódban a lehető leggyorsabb időt fogja futni a robot remélhetőleg. Itt mivel már ismert az útvonal, az sem baj, ha nem mindegyik RFID matrica kerül beolvasásra.

7. Fejlesztési lehetőségek

Az eddigi munkákat összefoglalva még a robot tesztelési fázisában vagyunk, ennek ellenére különböző területeken észrevettünk a jövőben még fejlesztést vagy finomhangolást igénylő területeket.

Az első prototípus megvalósítási munkafolyamatai alatt tapasztaltuk, hogy a szerkezeti váz és az arra készített nyomtatott áramkör, habár teljesen ráillő, az összeszerelési folyamatokat nem vettük figyelembe. Gyors szervizelés igénye esetén, nagyon nehezen szétszedhető, a motorok belső tárcsájának felhelyezése miatt, játszani kell annak helyére illesztéséhez. A jövőben erre figyelve az alsó NYÁK lemez motor tárcsája mentén nagyobb kivágással ez kiküszöbölhető.

Nagy minőségi szint ugrást jelenete, ha gyárban készült nyomtatott áramkört alkalmazhatnánk. Több réteg használatával az átkötések száma jelentősen csökkenthető lenne, kisebb raszter távolságokban lehetne elhelyezni az alkatrészeket. Emellett kivitelezhető lenne csak felületszerelt alkatrészek alkalmazása. Jelenleg az akkumulátor vezérlő egységben a kapcsoló MOSFET által leadott hőt hűtőbordán keresztül vezetjük el. Felületszerelt tokozás esetén a lemez felületén tudnánk ezt leadni. Ebben minket a két rétegen rendelkezésre álló felület hiányában nem állt rendelkezésünkre.

A másik nagy fejlesztési lehetőség a mikrovezérlő bővítésével történhet. Képfeldolgozási rendszerek alkalmazásával, számottevően csökkenthető a roboton található elektronika és a rendszer összetettsége.

8. Összefoglalás

A dolgozatunkban hat hónapnyi közös munkánkat olvashatták, szerencsére a csapatban már a projekt kezdetén megtaláltuk a közös hangot, így tudtunk hatékonyan dolgozni. Ennek köszönhetően az általunk kitűzött határidőket mindig sikerült betartanunk. Tóth Tamás felelt a szoftverért, Üveges Krisztián Tibor és Otlakán Péter fejlesztette, építette és tesztelte az elektronikai elemeket. Minden kritikus tervezési folyamatot közösen végeztünk, átbeszéltük kinek milyen igényei vannak és figyelmeztettük a másikat milyen szempontokra figyeljen oda. Így tudtuk elkerülni azokat a hibákat, amelyek jelentősen meghosszabbította volna a projekt kivitelezését, ezért is tudtunk ilyen rövid idő alatt ennyi mindennel elkészülni.

A dolgozatunkban egy labirintus megoldására specializálódott mobilrobot tervezési és megvalósítási folyamatait írjuk le. Részletesen jellemeztük, hogyan oldottuk meg a velünk szembe érkező problémákat.

A projekt itt nem fejeződik be, a jövőben megrendezett versenyeken is fogunk indulni, arra pedig meg fogjuk valósítani a fentiekben leírt továbbfejlesztési lehetőségek nagy részét.

9. Források:

9.1 Ábrák:

[4.2.1.] 18650

https://www.18650batterystore.com/cdn/shop/collections/new_114325435344.jpg?v=1608670986 (2024. 04. 16.)

[4.2.3.] ESP32

https://mikroelektronik.hu/2024317-large_default/szin-esp-wroom-32e-nodemcu-esp-32s-esp-wroom-32e-wifi-fejlesztokartya-soros-wifi-bluetooth-modul.jpg (2024. 04. 16.)

[4.2.4.] L293D

<https://www.doitsmart.hu/wp-content/uploads/2021/01/L293D.jpg> (2024. 04. 16.)

[4.2.5.] GP2Y0A41SK0F

https://hu.mouser.com/images/sharpmicroelectronics/lrg/GP2Y0A41SK0F_DSL.jpg (2024. 04. 16.)

[4.2.6.] MPU 6050

<https://github.com/ElectronicCats/mpu6050> (2024. 04. 16.)

[4.2.7.] RC522

<https://hu.szks-kuongshun.com/Content/upload/201884835/201811121640493497320.jpg> (2024. 04. 16.)

[4.2.8.] ITR9608

<https://www.rutronik24.com/pim/direct/everlight/medium/everlight-itr9608-f.jpg> (2024. 04. 16.)

9.2 3D modellek:

[1] IR szenzor

<https://grabcad.com/library/gp2y0a41sk0f-2> (2024. 03. 21.)

[2] MPU 6050

<https://grabcad.com/library/mpu6050-8>

(2024. 03. 21.)

[3] 18650 foglalat

<https://grabcad.com/library/one-cell-battery-box-battery-holder-1>

(2024. 03. 21.)

[4] RFID olvasó

<https://grabcad.com/library/mfrc522-simple-model-1>

(2024. 03. 21.)

[5] Motor

<https://grabcad.com/library/geared-motor-12>

(2024. 03. 21.)

9.3 Szoftver:

[6] Free RTOS

<https://github.com/FreeRTOS>

(2024. 03. 27.)

[7] Valós idejű operációs rendszerek (RTOS-ek) és alkalmazásuk

<https://www.digikey.hu/hu/articles/real-time-operating-systems-and-their-applications>

(2021. 02. 25.) *Lim Jia Zhi, Senior Embedded Software Engineer*

[8] Real-time Operating Systems (RTOS) for Embedded Systems

[https://www.researchgate.net/publication/377955410_Real-](https://www.researchgate.net/publication/377955410_Real-time_Operating_Systems_RTOS_for_Embedded_Systems)

[time_Operating_Systems_RTOS_for_Embedded_Systems](https://www.researchgate.net/publication/377955410_Real-time_Operating_Systems_RTOS_for_Embedded_Systems)

(2023. 01.) *Ashif Mohammad; Rimi Das; Cp Md Aminul Islam Rana; Farhana Mahjabeen*

[9] Attitude Estimation with Accelerometers and Gyros Using Fuzzy Tuned Kalman Filter

[https://www.researchgate.net/publication/242746278_Attitude_Estimation_with_Accelerometers](https://www.researchgate.net/publication/242746278_Attitude_Estimation_with_Accelerometers_and_Gyros_Using_Fuzzy_Tuned_Kalman_Filter)

[_and_Gyros_Using_Fuzzy_Tuned_Kalman_Filter](https://www.researchgate.net/publication/242746278_Attitude_Estimation_with_Accelerometers_and_Gyros_Using_Fuzzy_Tuned_Kalman_Filter)

(2015. 03.) *Chul Woo Kang*

- [10] Creating a Simulation Environment for the Micromouse
https://www.researchgate.net/publication/358945942_Creating_a_Simulation_Environment_for_the_Micromouse
(2022. 02.) Martin Komák, Elena Pivarciova
- [11] A differential steering control with proportional controller for an autonomous mobile robot
<https://ieeexplore.ieee.org/document/5759849>
(2011. 05. 02.) Mohd Saifizi Saidonr; Hazry Desa; Md Noor Rudzuan
- [12] Getting Started With Proportional Control On A Micromelon Rover
<https://micromelon.com.au/resources/getting-started-with-proportional-controls>
Luc Nederhof
(2024. 03. 27.)
- [13] PID Control
<https://www.autonomousrobotslab.com/pid-control.html>
(2024. 03. 27.)
- [14] Serial Peripheral Interface (SPI)
<https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi/all>
(2024. 03. 27.)
- [15] I2C kommunikáció
<https://learn.sparkfun.com/tutorials/i2c/all>
(2024. 03. 27.)
- [16] The MPU6050 Explained
<https://mjwhite8119.github.io/Robots/mpu6050>
(2019. 07. 26.)

9.4 Adatlapok:

- [17] TL431
https://www.ti.com/lit/an/slva987a/slva987a.pdf?ts=1713334816324&ref_url=https%3A%2F%2Fwww.google.com%2F
(2018. 06.)

[18] IRF9510

<https://www.vishay.com/docs/91072/91072.pdf>

(2021. 07. 16.)

[19] L7805CDT-TR

https://www.hestore.hu/prod_getfile.php?id=10766

(2016. 11.)

[20] LM1117IMP-3.3

https://www.hestore.hu/prod_getfile.php?id=5351

(2006. 04.)

[21] ITR9608-F

https://www.hestore.hu/prod_getfile.php?id=10583

(2016. 11. 15.)

[22] WS2812B

https://www.hestore.hu/prod_getfile.php?id=9800

(2017. 03.)

[23] RC522-MFRC

https://www.hestore.hu/prod_getfile.php?id=8199

(2014. 08. 17.)

[24] GP2Y0A41SK0F

<https://www.alldatasheet.com/datasheet-pdf/view/506283/SHARP/GP2Y0A41SK0F.html>

(2007. 03.05.)

[25] MPU6050

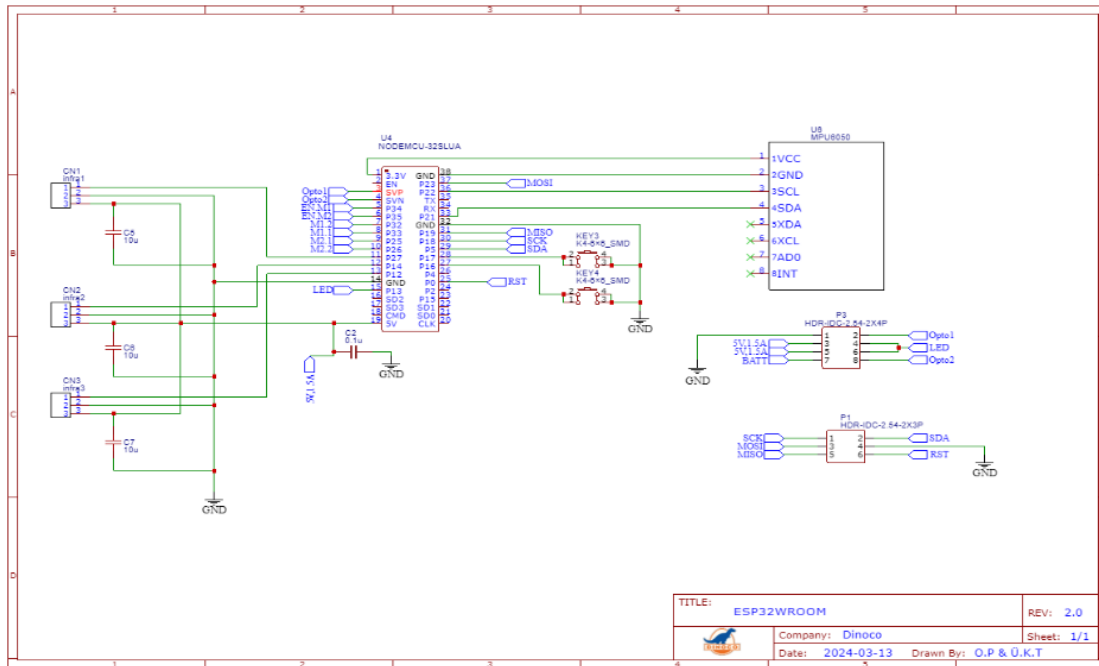
https://www.hestore.hu/prod_getfile.php?id=8161

(2013. 08. 19.)

10. Csatolmányok:

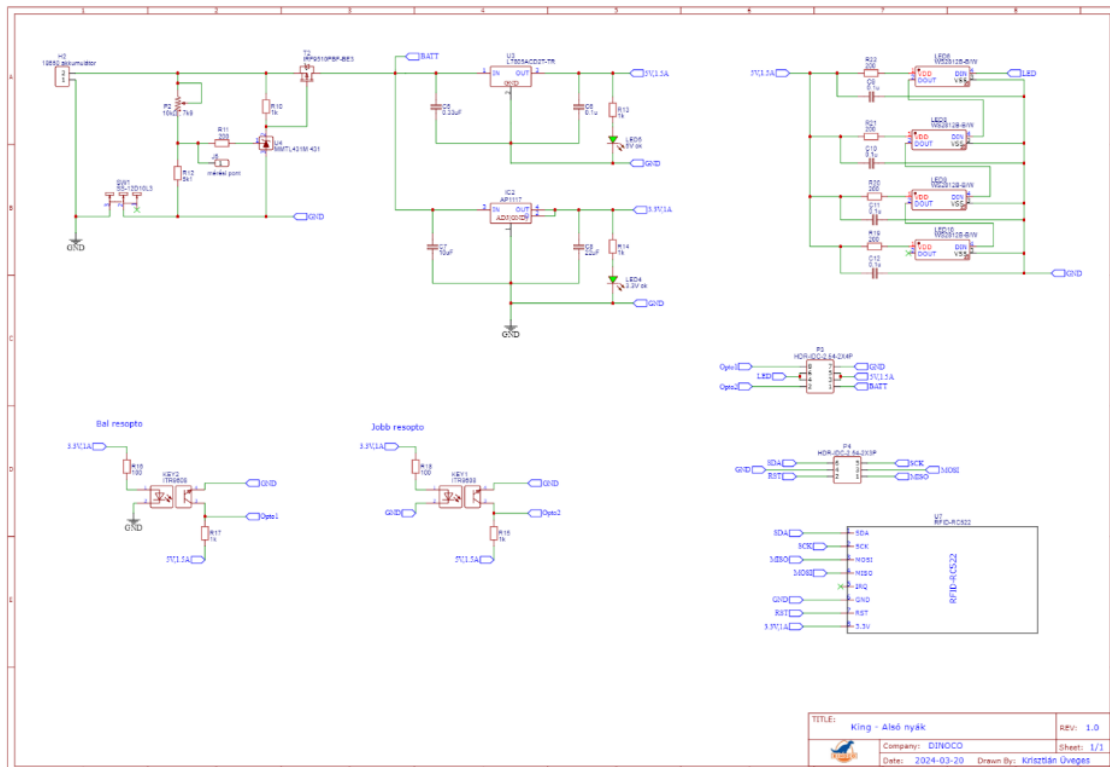
{1}

10.1 ábra, Végleges kapcsolása az ESP32-nek



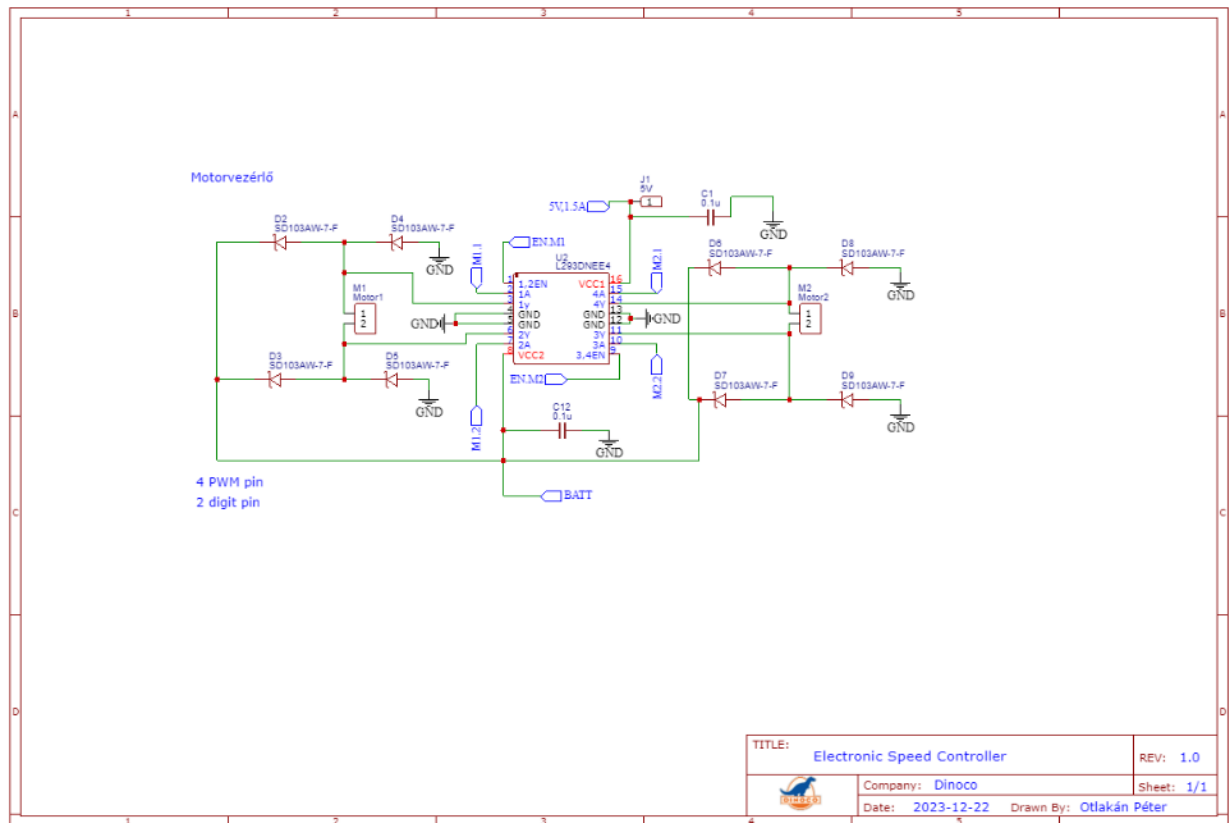
{2}

10.2. ábra, Végleges kapcsolása az alsó NYÁK-nak

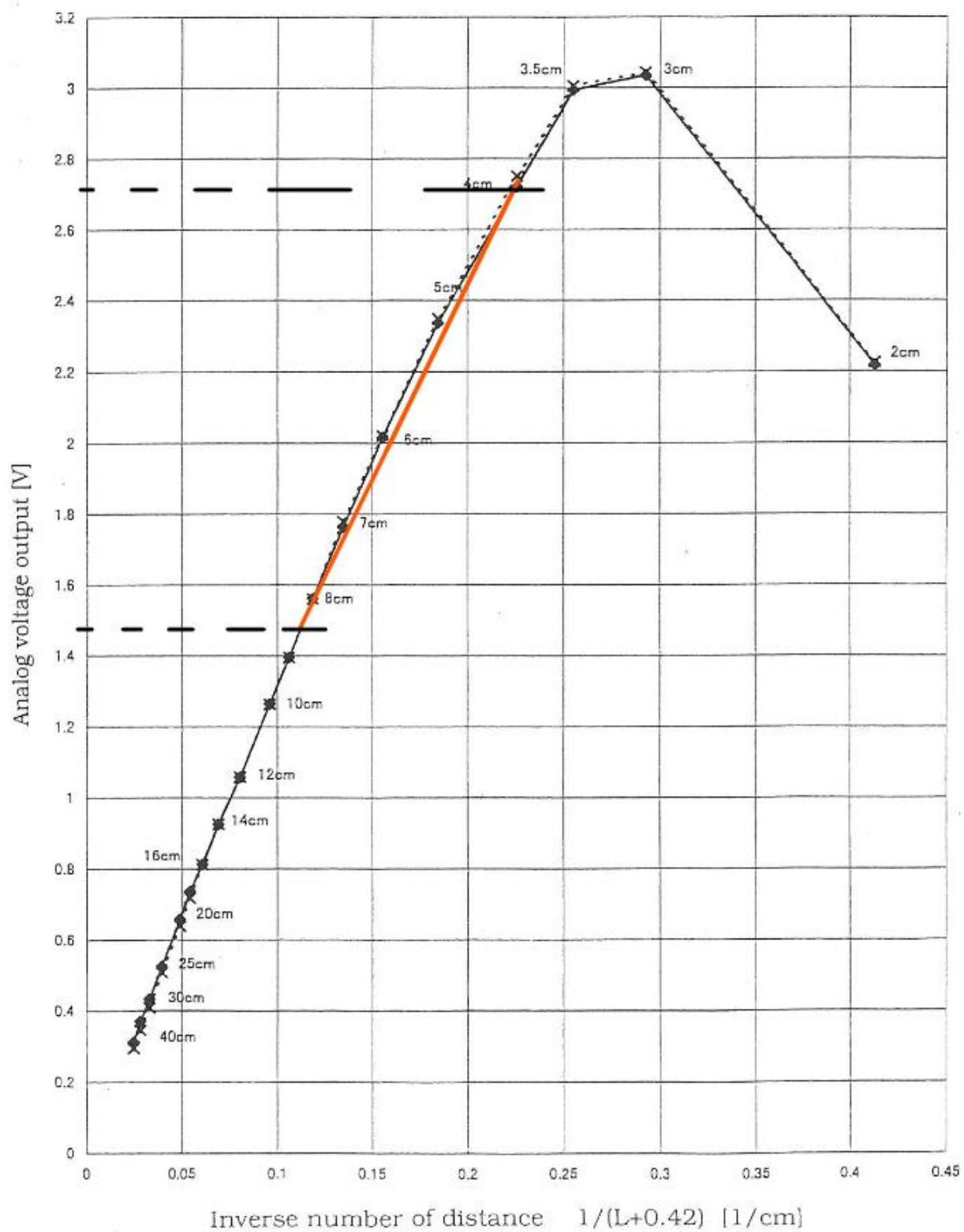


{3}

10.3. ábra, Végleges kapcsolása a motorvezérlőnek



{4}



10.4. ábra (forrás:[24]), Infra távolságérzékelő szenzor feszültség – fordított távolság grafikonja az általunk használt szakasszal (narancssárga egyenes)