



36. ORSZÁGOS TUDOMÁNYOS DIÁKKÖRI KONFERENCIA 2023

TUDOMÁNYOS DIÁKKÖRI DOLGOZAT

ROBOT TENGERALATTJÁRÓ ÉPÍTÉSE - AQUALAB

Szerző:

Kertész Domokos

Balassi Bálint Nyolcévfolyamos
Gimnázium,
11. osztály

Konzulensek:

Dr. Komáromi Annamária
gimnáziumi tanár

Dr. habil. Wühl Tibor
egyetemi docens



Tartalomjegyzék

Bevezetés.....	4
A robot felépítése	4
Tervek.....	5
Fejlesztés szakaszai	5
Első prototípus.....	6
Második prototípus.....	7
Test felépítése.....	8
Ballasztrendszer	9
Fedélzeti vezérlés	11
Áramellátás.....	13
Szoftverek.....	14
Fedélzeti szoftverek.....	14
A vezérlő program.....	16
Megrendelők számára készített felület	19
Kommunikáció	20
Hasonló projektek elemzése.....	21
További tervek.....	22
Összegzés	22
Köszönetnyilvánítás	23
Irodalomjegyzék.....	24

Absztrakt

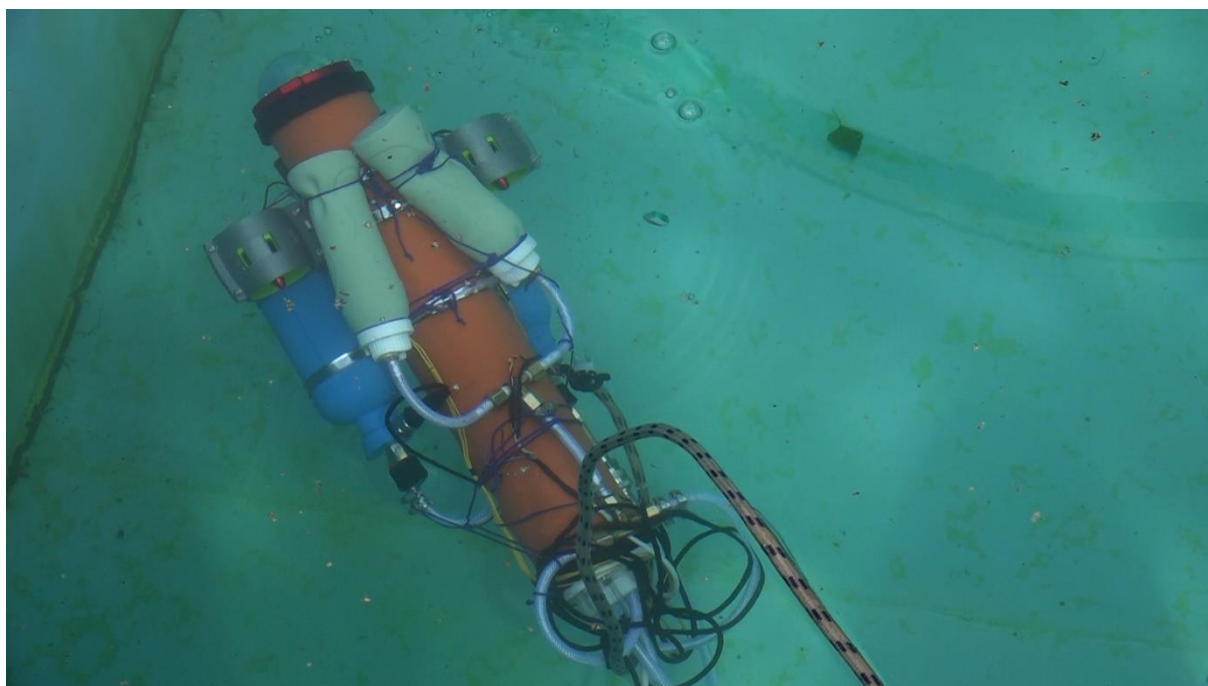
Projektemmel egy robot tengeralattjáró fejlesztését tűztem ki célként, amely képes biológiai kutatást végezni a vízfelszín alatt. Jelenleg a természetes vizek minőségének vizsgálata helyszíni mintavétellel és laboratóriumi elemzéssel történik. A minta szinte kizárólag a felső rétegekből kerül levételre, ráadásul gyakran csak a partközeli régiókban. A projektemmel – a kutatók pozitív visszajelzéseire reagálva – ezen szeretnék változtatni. Tehát a vizet vizsgáló biológusok számára az általam fejlesztett AquaLab robot tengeralattjáró egy automatizálási lehetőség, ami a mintavételi és elemzési nehézségekben nyújt segítséget, mert hiszek abban, hogy a víz állapotának felügyelete az első lépés természetes vizeink minőségének megőrzése felé.

A robot egy körülbelül 15 kilogrammos víz alatti szerkezet, amelynek alapját egy PVC cső adja, a vezérlése Arduino mega és Raspberry pi segítségével valósítható meg, míg az irányítás élőben, a partról történik. A fedélzetén található biológiai szenzorok adatait szintén a felszínen élőben lehet feldolgozni, az általam fejlesztett informatikai rendszeren keresztül, melyet C++ programnyelvben írtam. A merülés és az emelkedés vezérlésére egy ballasztrendszert fejlesztettem, melyet hasonló méretű robotokon még nem alkalmaztak. Ez két nagynyomású tartályból, és két felfújható tartályból áll. A nagynyomású tárolókból mágnesszelepek segítségével ereszthető át a levegő, mely a test külső részén elhelyezett változó térfogatú tartályokba áramlik. Feltöltéskor a térfogat megnő, ezzel emelkedésre bírható a test, míg a levegő kieresztésének hatására a test süllyedni kezd. A rendszert teljes egészében én fejlesztettem, az alapoktól indulva.

Dolgozatomban a teljes rendszer működését, építésének folyamatát mutatom be.

Bevezetés

Természetes vizeink vizsgálatára nagy szükség van minél jobb állapotban való megőrzésük érdekében. A jelenlegi vizsgálati módszerek véleményem szerint hatékonyabbá tehetők, gondolva itt az automatizálási lehetőségekre. Természetesen megjelentek már kézi műszerek, melyekkel bizonyos tulajdonságok mérhetők, a piackutatásom után azonban kiderült számomra, hogy a biológusok hasznos információkat nyerhetnek ki abból, hogy a minőséget mutató paraméterek a mélységgel és a távolsággal hogyan változnak. Erre alkalmas az általam fejlesztett robot tengeralattjáró, mely víz alatti laboratóriumként szolgál, és egy automatizálási lehetőség a biológusok számára, ami mintavételezési és elemzési feladatokhoz nyújt segítséget.



1. ábra - A merülő robot

A robot felépítése

A rendszer két fő részből áll, magából a tengeralattjáróból, illetve az azzal kommunikáló és az azt vezérlő parti egységből. Az irányítás folyamatosan a felszínről történik, a fedélzeten a kiadott parancsokat nem módosítja egyik program sem. A robot főbb eszközei közé tartoznak a meghajtást végző motorok, az emelkedést és süllyedést szabályzó ballasztrendszer, illetve a biológiai szenzorok. Ezek segítségével a robot képes a vízben föl-le, előre-hátra, illetve kanyarodva haladni, és mindeközben biológiai méréseket végezni. A vezérlés és a kapcsolat a felszínnel folyamatos.

A fedélzeten egy Raspberry pi számítógép és egy Arduino mega végzi az adatok begyűjtését és a vezérlést. A mikrokontrollerre azért volt szükség, mivel az analóg jelek olvasására a Raspberry pi csak bővítőkártyával képes, illetve a motorok vezérléséhez szükséges négyzögjeleket sem tudja megfelelő pontossággal előállítani.

A vezérlőegység akármilyen Windows vagy Linux operációs rendszerrel futó számítógép lehet. Egy saját fejlesztésű grafikus programmal lehet végezni a vezérlést. Ezt támogatja a hardveres gázkar és joystick. A kommunikáció UTP kábel segítségével történik, az internetes protokollok használatával.

Tervek

A projekt kezdetén egy olyan robotot szerettem volna építeni, amely a Balaton fenekén képes kutatásokat végezni különböző történelmi emlékek után. A projekt fejlesztése közben azonban felmerült egy sokkal kézenfekvőbb probléma, ez pedig a víz minőségének vizsgálata. Ezután kutatókkal konzultáltam, ennek alapján kiderült, hogy milyen igényeknek kell megfelelni. Így a biológusok konkrét elképzelései alapján folytattam a fejlesztést. A biológiai vizsgálatokkal kapcsolatban támasztott elvárások között konkrét mérendő tulajdonságok (pH, zavarosság, stb.) is helyet kaptak. Emellett szerettem volna a mintavételezést legalább másodpercenként elvégezni, illetve a mért adatok feldolgozására kifejleszteni egy biológusok által használható rendszert, amivel mérülés közben tudják az adatokat felügyelni.

Cél volt, hogy praktikus és költséghatékony megoldást találjak, amihez a műszaki megvalósítás terén sikerült újdonságot kidolgoznom. A hasonló méretű robotokon az emelkedést és süllyedést motorokkal valósították meg, amivel én nem voltam elégedett, ezért elkezdtem fejleszteni egy emelkedést és süllyedést Arkhimédész törvénye szerint szabályzó rendszert. Fontos célkitűzéskén szerepel a folyamatos irányíthatóság, és az alapfeladatok pontos és megbízható elvégzése. Emellett, a tanulás, és személyes fejlődésem egy fontos célja ennek a projektnek, ennek érdekében a rendszer minél több részét szerettem volna én magam fejleszteni.

Fejlesztés szakaszai

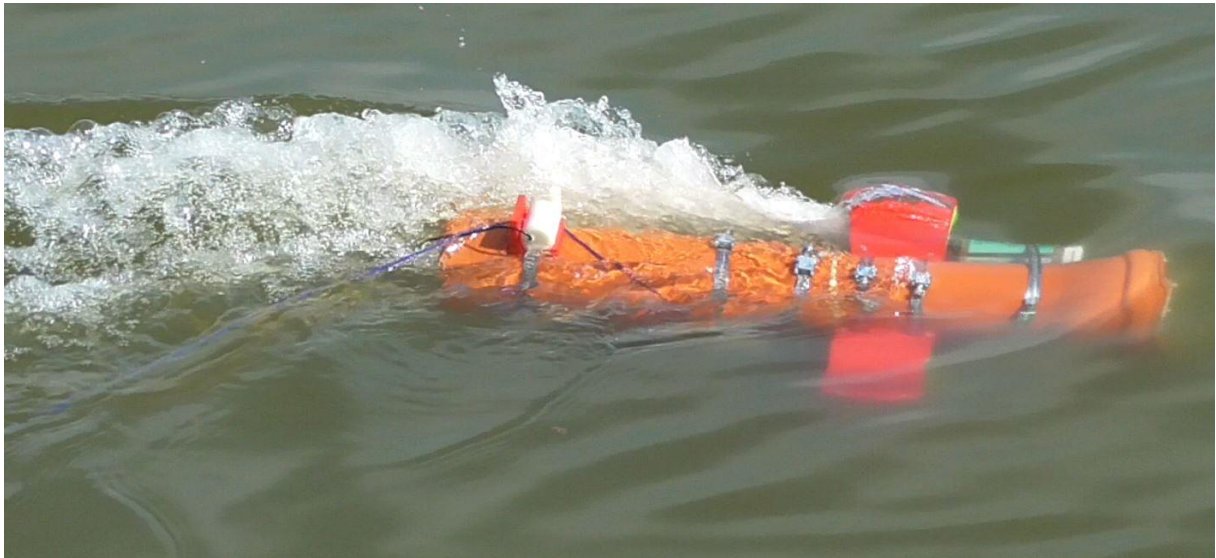
A munka közben három prototípust készítettem, ezek segítségével tagolható az építés különböző részekre.

Első prototípus



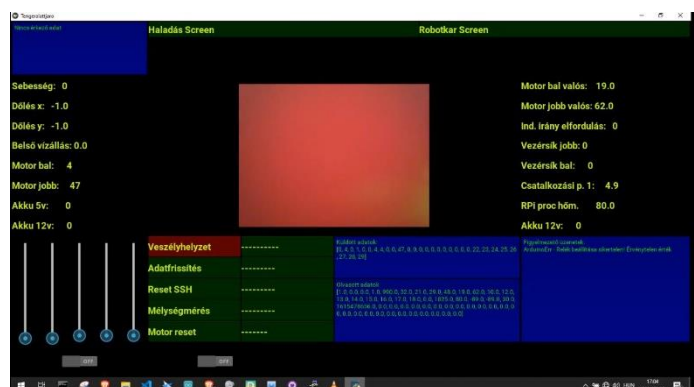
2. ábra - Az első prototípus kameraképe

változtatta meg a test dőlését, ezáltal a víz alá buktatva vagy éppen fölé emelve a testet. A robotot egy lakóhelyemhez közeli élővízben ki is próbáltam, a feladatát megfelelően elvégezte, a merülésből rengeteg tapasztalatot szerezhettem.

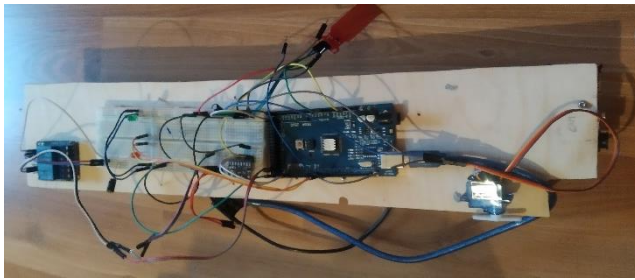


3. ábra - Az első prototípus merülés közben

A vezérlést a felszínről egy saját fejlesztésű Python programmal lehetett elvégezni, amely a vezérlő számítógépen futott. Ez csatlakozott egy Raspberry pi számítógép által létrehozott vezeték nélküli, lokális hálózathoz, amelyhez a tengeralattjáróból koaxiális kábelben keresztül kivezetett antennán keresztül lehetett kapcsolódni. Az adattovábbítás azért történt ilyen módon, mivel a kezdeti terveim között szerepelt, hogy a felszínen egy pontonhajó segíti a rendszert, mintegy átjátszóállomásként működve a part és a tengeralattjáró



4. ábra - Az első prototípus vezérlőrendszere



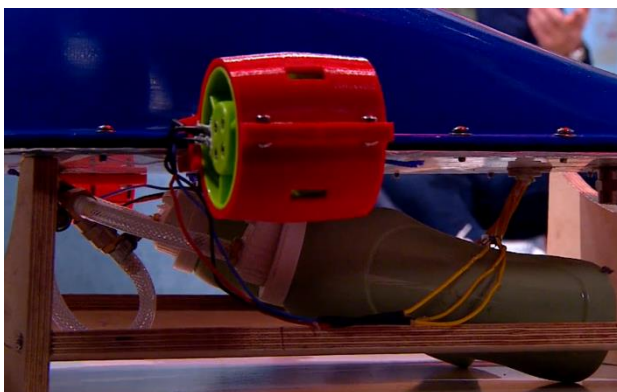
5. ábra - Az első prototípus vezérlő elektronikája

semmilyen kutatás elvégzésére, de az alapfeladatokat megfelelően el tudta látni, így a további fejlesztésekhez jó irányt adott.

Második prototípus

Az első próbák sikere után szerettem volna egy saját testformát használni a fejlesztésnél, mivel úgy ítélt meg, hogy nagymértékben

korlátozza a lehetőségeimet a PVC cső. Ezért egy hajóépítő segítségével egy új test építésébe kezdtem üvegszálas műanyagból, amit közel egy hónapos munka után el is készítettem. A test

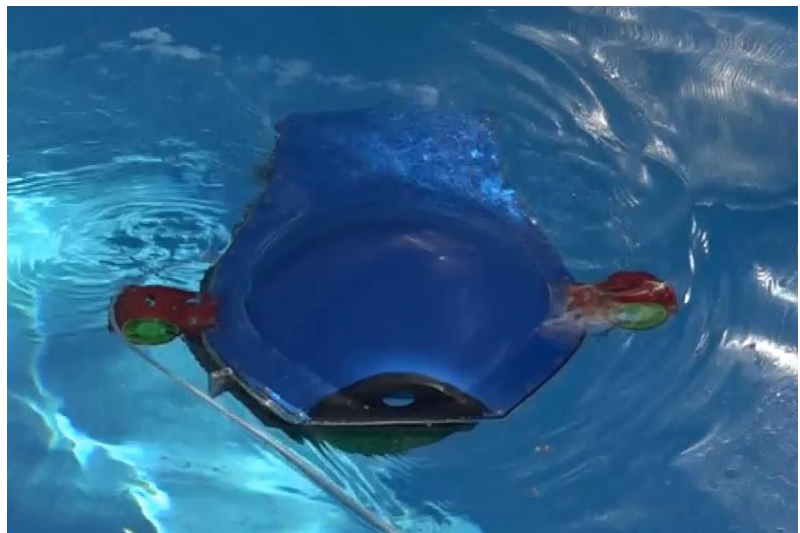


7. ábra - A második prototípus motorjai és a ballasztrendszer

miatt nem volt megfelelően kiegyensúlyozható, előre-hátra, és oldalra a legkisebb külső behatástól is megdőlt, ráadásul a rengeteg holtter miatt a testnek nagyjából 25kg-nak kellett volna lennie, ami a motorok rendkívüli ereje ellenére sem egy könnyen mozgatható szerkezet.

között. Ezt a megoldást később elvettem, mivel a víz alatti kommunikációt biztosító kábel akkor sem akadályozta nagymértékben a munkát, ha közvetlenül a partra került kivezetésre.

Ezen robot nem volt alkalmas



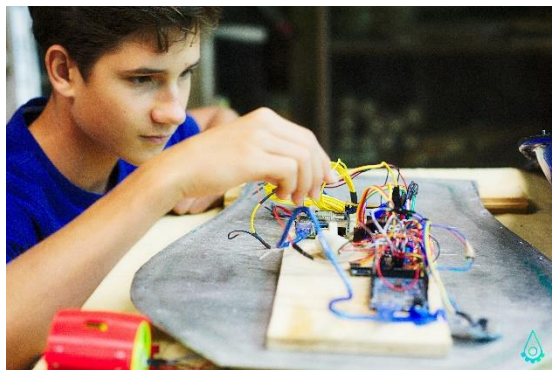
6. ábra - A második prototípus merülés közben

speciális formájából adódóan igen jó menettulajdonságokkal rendelkezett volna.

A rendszer szoftverek tekintetében már közelített a jelenlegi formájához, így az erre vonatkozó fejezetben mutatom be. A lényeges különbség az volt, hogy a roboton volt hely robotkarnak, illetve biológiai szenzoroknak is. A kísérletek végül nem igazolták a tervemet, mivel a test a formája

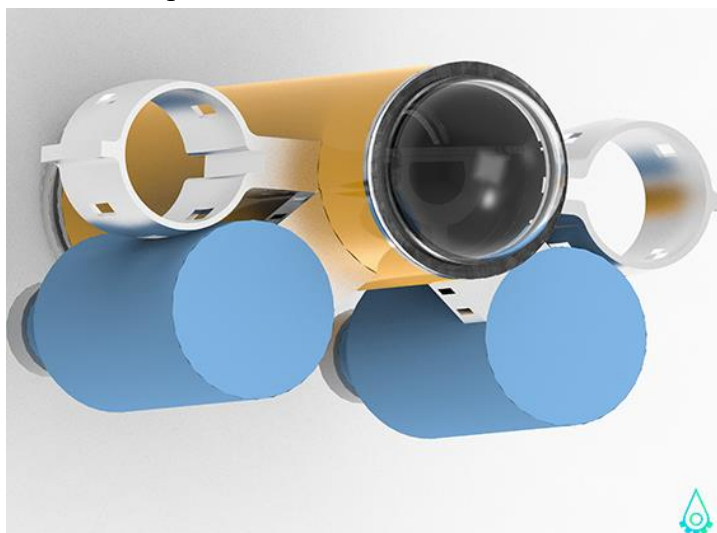
Ez a konstrukció tehát lényegében kudarccal zárult, azonban a szoftveres és rendszerszintű fejlesztések ennek ellenére hasznosan épültek be a rendszerbe, és ebből a hibából rengeteget tanultam.

Test felépítése

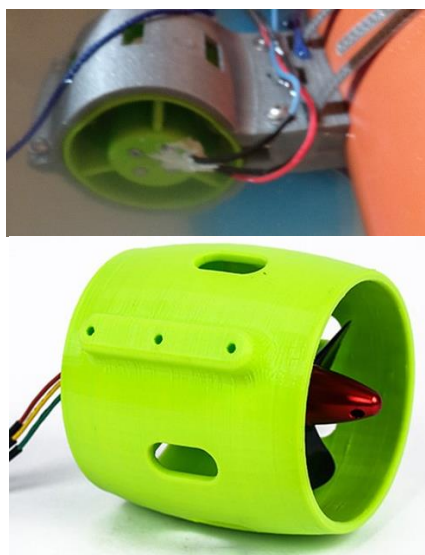


8. ábra - A második prototípus kezdeti elektronikája

A második test sikertelensége után inkább visszatértem a PVC csőhöz, és azon eszközöltem fontosabb módosításokat. Első lépésként a belső elektronikai alkatrészek elhelyezését gondosan megterveztem, a kiegyensúlyozásra külön figyelmet fordítottam, és a kamerának speciális „ablakokat” vásároltam.



9. ábra - A végleges test 3D modellje



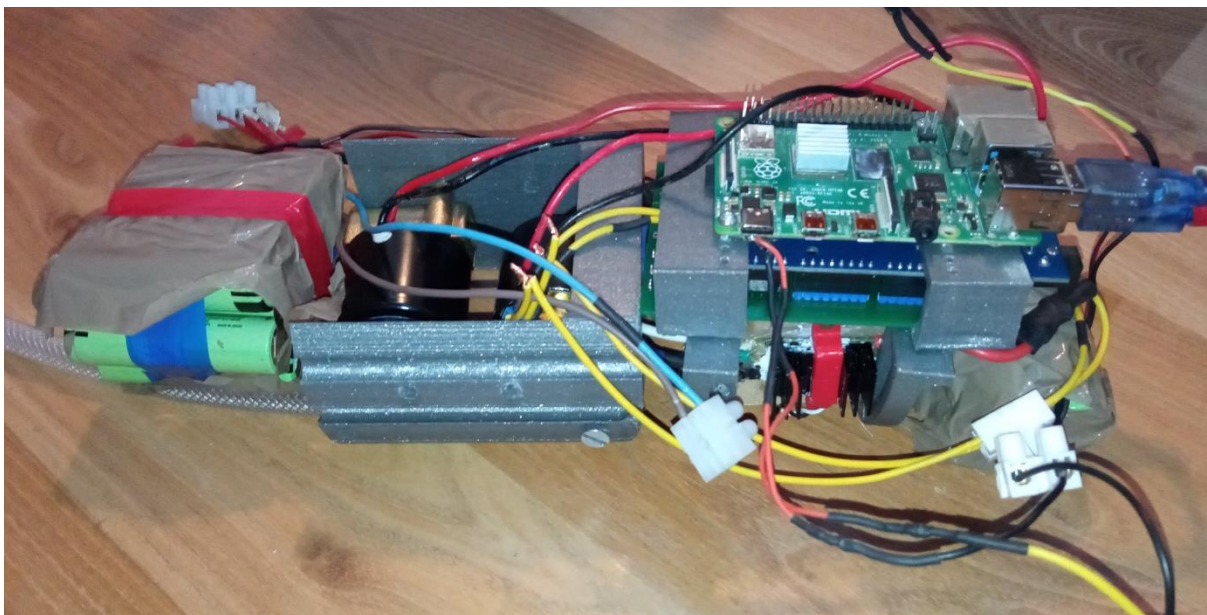
10. ábra - A motorok

A test a motorok elrendezését tekintve nem



11. ábra - A testen csúsztható rögzítőeszközök

változott, csupán a motorok 3D nyomtatott rögzítő elemeit optimalizáltam. A rögzítést továbbra is csőbilincscsel oldom meg, ezen a testen azonban helyet kapnak a hajtóművek mellett a biológiai szenzorok és a ballasztrendszer alkatrészei is. A ballasztrendszer két nagy tartálya mozgatható a test hosszanti tengelye mentén, így a kiegyensúlyozás problémáját sikerült megoldanom, mivel ezek jelentős súlyt képviselnek.

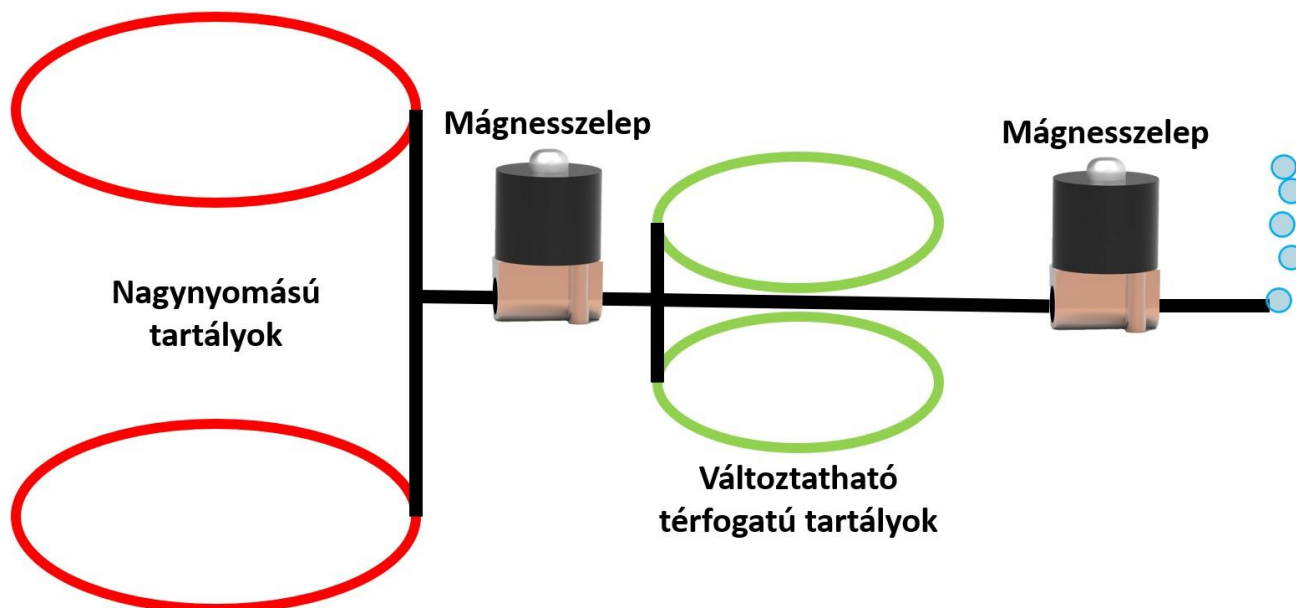


12. ábra - A végleges test elektronikája

A belső elektronika méretét saját tervezésű nyomtatott áramköri lappal jelentősen csökkentettem, ezáltal az első testhez képest nagyjából negyedannyi helyre van szüksége a vezérlésnek. A testen belül jelentős hely marad így az akkumulátoroknak, illetve a ballasztrendszer működéséért felelős mágnesszelepeknek. A test hátsó részén a ballasztrendszerhez tartozó nagynyomású csöveket vezettem ki, a szigetelést tömszelencékkel megoldva.

Ballasztrendszer

A merülést és a süllyedést szabályzó rendszer két nagynyomású tartályból és két változtatható térfogatú tartályból épül föl. A változó térfogatú ballaszttartályokba levegőt lehet



13. ábra - A ballasztrendszer működése



14. ábra - A rendszer szemből

be-, illetve kiengedni a nagynyomású tartályokból, amik ezáltal vagy fölfújódnak, vagy összezsugorodnak. Így a test által kiszorított víz mennyiségével Archimédész törvénye alapján nő, vagy csökken a testre ható felhajtóerő, mely így emelkedni vagy süllyedni kezd. A levegő a nagynyomású tartályokból egy mágnesszelepen keresztül jut a változtatható térfogatú ballaszttartályba, ahonnan a vízbe lehet kiengedni a fölösleges

levegőt. A kieresztés miatt a rendszerből folyamatosan fogy a levegő, azonban a tapasztalataim alapján megfelelően hosszú ideig lehetővé teszi a merülést a két nagynyomású tartályban maximum 10 bar nyomáson lévő 4 liter levegő. A tartályok nyomását folyamatosan felügyeli egy nyomásmérő szenzor, így a kezelő személy és a szoftver is dönthet úgy, hogy már nem engedélyezi a további merülést, mivel fennáll annak a kockázata, hogy a robot nem tud a felszínre emelkedni. A rendszer vezérlése érzékeny, azonban a 15kg-os tengeralattjárónál az általam választott időalapú vezérlés is megfelelő pontosságot eredményez. Ez úgy működik, hogy a mágnesszelepek adott igen rövid időre (20-60ms) nyílnak ki, ami alatt csak kevés levegő jut át rajtuk, így nagyon pontosan lehet beállítani a tartályokban lévő levegő mennyiségét.

A tesztek előtt tartottam tőle, hogy a rendszer a számításaim ellenére sem alkalmas a lebegés állapotának eléréséhez, azonban a kísérletek közben az a kellemes meglepetés ért, hogy mindenféle nehézség nélkül ez mégis megtehető, köszönhetően a test arányaihoz képest kicsi levegőbefecskendezésnek.

A rendszert levegővel olyan módon lehet feltölteni, hogy a teljes test nem kerül szétbontásra, mivel az alkalmazott gyorscsatlakozók segítségével a testen kívül is elvégezhető a feladat, így a merülés rövid megszakítással folytatható a levegő elhasználása után is.

A nagynyomású tartályok beszerzésekor komoly problémák merültek föl azzal kapcsolatban, hogy a 10 bart is elbíró palackok nagyon magas áron voltak csak elérhetőek. Emiatt én egy tűzoltókészülék átalakítása mellett döntöttem, ami azt jelentette, hogy a nagynyomású csatlakozókhoz kialakítottam egy átalakítót a tűzoltókészülék különleges menetéhez. A tartályok azért voltak megfelelőek, mert a nyomást rendkívüli módon tűrik, üzemi nyomásuk 15 bar, a porral való feltöltés előtt pedig 27 baron tesztelik őket. A rendszer leggyengébb pontja így már a tűzoltókészülék és a mágnesszelep közötti rész, ezt azonban természetesen leteszteltem és mindenhol adott nyomásra hitelesített eszközöket használtam. A 10 bar semmilyen körülmények között nem jelenthet problémát a rendszerben.

A változó térfogatú tartályok kategóriája gyakorlatilag nem létezik, így a beszerzés igen nagy nehézséget jelentett. A megoldást egy olyan kulacs jelentette, amelyik szilikonból készült, és nincs merev váza. Erre könnyen lehetett csatlakozót készíteni egy menetfúró segítségével. Mivel itt nincs nagy nyomás, a térfogatnövekedés anélkül is elérhető, ezért a tartályok megfelelően működnek.



15. ábra - A rendszer felülről

A nagynyomású levegő, és az esetleges meghibásodás esetén bekövetkező süllyedés miatt a rendszer egyik kulcsfontosságú része a ballasztrendszer, ahol rendkívül figyelni kell az üzembiztonságra. Ezt sok tesztel és minőségi alkatrészekkel értem el. A felszínen történő esetleges szivárgások, vagy légrobbanások elkerülése érdekében a rendszerbe építettem vészleeresztőt, mely egy mozdulattal megszünteti a túlnyomást a rendszerben.

Fedélzeti vezérlés

A fedélzeti rendszerek ugyan folyamatos kapcsolatban állnak a felszínnel, azonban a vezérlés megszűnése esetén is képesnek kell lennie bizonyos műveletek elvégzésére, így komolyabb feldolgozóegységekre is szükség van a víz alatt.

Az elektronika egy Raspberry pi számítógép köré épül, mivel ezen keresztül zajlik a kommunikáció a felszínnel. Ehhez csatlakozik egy Arduino Mega mikrokontroller, amely a motorok és a ballasztrendszer vezérlését végzi, mivel itt szükség lehet analóg jelek érzékelésére, valamint PWM jelek kibocsátására. Emellett a vészforgatókönyvek szerint az Arduino bírna a legtovább egy esetleges balesetet, így szerencsés esetben még a ballasztrendszert sikeresen

felpumpálhatná levegővel az emelkedéshez. Amennyiben a kommunikáció a fedélzeti számítógéppel folyamatos, az Arduino semmilyen esetben sem bírálja felül az adatokat.

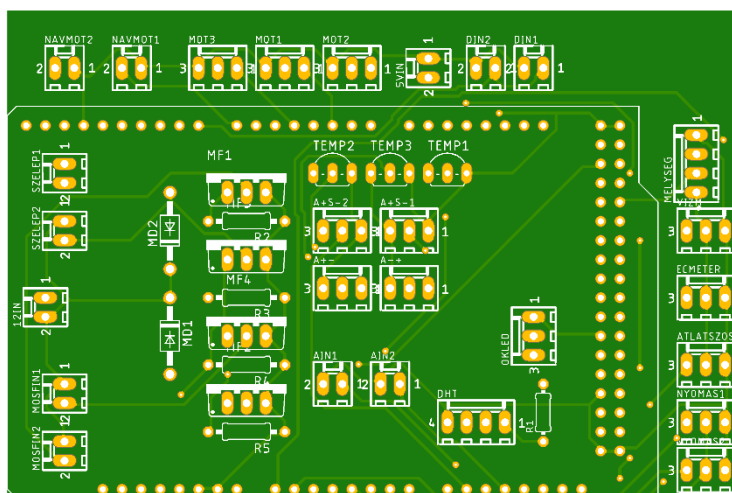
A tengeralattjáró élő kameraképet továbbít a felszínre, melyet egy USB csatlakozóval ellátott webkamera szolgáltat. A választás amiatt esett erre a megoldásra, mivel szoftveresen könnyen kezelhető, illetve jóval költséghatékonyabb, mint az alternatívaként felmerült Pi camera.



16. ábra - A kamera képe

A meghajtást kefe nélküli motorok teszik lehetővé, melyeket eredetileg csak tesztelésre rendeltem, azonban ereje és könnyű vezérelhetősége miatt megmaradtam ennél a megoldásnál. Egyetlen hibája volt, hogy a motor tervezési hiba miatt néhány óra használat után a tengelye mentén termelt minimális salakanyagot, mely akadályozta a forgást. Ez a nehézség egyszeri takarítás és ismételt kenés után a megoldódott. A vezérléséhez ESC-t használok, az első testen egy olcsó, eredetileg modell helikopterekhez való eszközt használtam, azonban hamar felismertem, hogy itt precíz eszközre van szükség, így a Blue Robotics cég RUV-hez vagyis tengeralattjárókhoz fejlesztett modelljére váltottam. Ez már alkalmas volt a tökéletesen pontos vezérlésre.

A fent ismertetett rendszerek egy két rétegű nyomtatott áramkörön keresztül kapcsolódnak egymáshoz, ezzel csökkentve az elektronika méretét. A lapot én terveztem, a gyártást a JLCPCB nevű cég végezte, melynek minősége a kipróbált vasalásos házi technikát jelentősen felülmúlta. Az Arduino Mega a



17. ábra - A nyomtatott áramkör

bővítőkártyákhoz hasonlóan kapcsolódik a rendszerhez, ezáltal az összes kimenet összeköthető a PCB-n keresztül. Ehhez a kapcsoláshoz köthetők sorkapcsokkal, tűkesorral, vagy közvetlen forrasztással a szenzorok, és vezérlőeszközök.

Áramellátás

A rendszernek szüksége van 12 volt feszültségre a haladáshoz és a mágnesszelepek működtetéséhez, azonban a vezérlés 5 voltos feszültséget igényel. A biztonság miatt ezt két különböző akkumulátorral oldottam meg. Az elektronikát egy 9,6 Ah kapacitású, maximum 40 A leadására alkalmas áramforrás táplálja. Ez körülbelül 5 óráig tudja a rendszert elektromossággal ellátni. A 12 voltos akkumulátor 9,6 Ah kapacitású, és maximum 40A leadására képes, mely általános haladási sebességgel a kísérletek alapján megközelítőleg 3 óráig bírja a terhelést.



18. ábra - Az akkumulátorok

Az akkukat ponthegeesztési módszerrel készítettem a Panasonic NCR18650 lítium-ion cellákból. Kiemelten a 12 volt feszültségű pakknál van szükség nagy terhelhetőségre, emiatt felmerül, hogy a lítium-polimer akkuk biztosabb megoldást jelentenének, azonban az általam választott cellák nagy terhelésre vannak kialakítva. Mivel a lítiumos cellák nagyobb kapacitásúak, végül emellett a megoldás mellett döntöttem. A biztonság miatt mind a kettő pakkot elláttam védelmi elektronikával, melyek képesek a rövidzárlat okozta bajt megelőzni, mivel ez esetben lekapcsolnak. A két akkumulátor összesen körülbelül 0,8 kg tömegű.

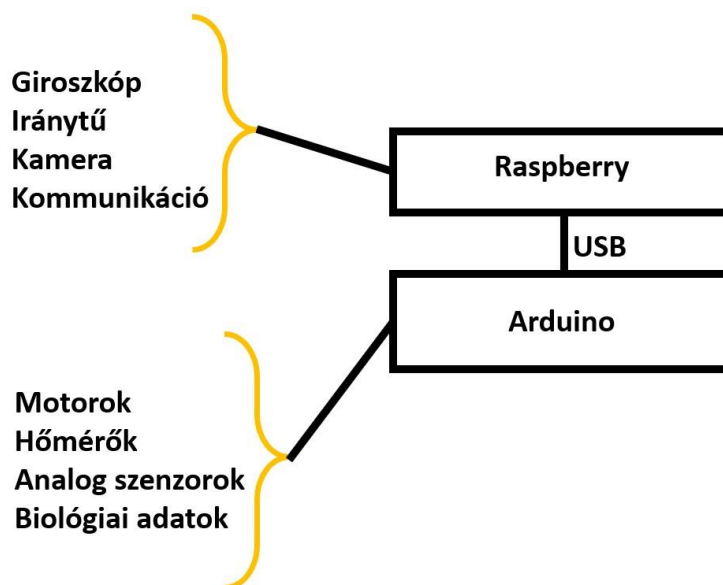
Szoftverek

A szoftvereket három részre lehet felosztani. A fedélzeten futó rendszer a közvetlen vezérlést végzi, valamint kommunikál a parancsok kiadására alkalmas vezérlőegységgel. Ettől a két rendszertől elkülöníthető a harmadik, mely a biológiai adatok elemzésére alkalmas, akár merülés közben is. Ez a felszíni vezérlőprogrammal kommunikálva, vagy kompatibilis fájlokon keresztül kapja meg a szükséges adatokat.

Fedélzeti szoftverek

A fedélzeten a Raspberry pi-n és az Arduino megán egymástól független, de egymással folyamatosan kommunikáló programok futnak.

A fedélzeti számítógépen fut a rendszer egyetlen Python programozási nyelvben írt kódja. Itt azért volt alkalmas a szkript nyelv, mivel a számítási kapacitás elegendő volt a kód futtatásához. A program 6 külön szálon fut, melyek vezérlik



19. ábra - A vezérlés megosztása

- a belső adatok begyűjtését (fedélzeti számítógép hőmérséklet, egyéb szenzorok) és a biztonsági funkciókat,
- a kommunikációt az Arduino megával,
- az élő kamerakép olvasását a kamerából, és küldését a felszínre,
- a felszínre küldendő adatok továbbítását,
- a felszínről érkező adatok olvasását.

Mivel a folyamatok elkülöníthetően futnak, a rendszer folyamatosan friss információkkal rendelkezik az összes fontosabb eseményről. Ez rövidíti a parancs felszíni kiadásától a végrehajtásig eltelt időt. A Python előnye, hogy a folyamatok párhuzamos futására könnyen használható beépített könyvtárakkal rendelkezik, ezért választottam a C++ helyett, ahol a mutex zárolások bonyolultabb megoldásnak bizonyultak, illetve kevesebb tapasztalatom volt a nyelvben a fejlesztések kezdetén. A rendszerben ez az egyetlen program, ami a kezdeti

fejlesztések óta csak minimálisan változott a felépítésében, bár a kód természetesen teljesen átdolgozásra került.

Az Arduino-n egy C programnyelvben írt program található, amely a mikrokontroller felépítéséből adódóan egyetlen szálon fut, azonban a feldolgozó függvények egyszerűsége miatt a lineáris programban a fő végtelen hurok másodpercenként közel 50-szer képes lefutni.

Az Arduino Mega egy nyomtatott áramkört laphoz csatlakozik, melyen egy kontroll LED is helyet kapott. Ez a hibakeresésnél lehet hasznos, a végső verzióban a kommunikáció hiányát, meglétét, és egyéb beállított hibákat jelez különböző szignálokkal.

A program folyamatosan ellenőrzi, hogy rendelkezésre áll-e új beérkező adat. A legfrissebb adatokat pedig folyamatosan feldolgozza, akkor is, ha nem érkezett az előző beállítás óta adat. Ez elősegíti, hogy ne fordulhasson elő az, hogy a rendszer nem a legfrissebb beállított paraméterekkel rendelkezik.

Az Arduino-n futó kód beállítja a ballasztrendszert a megfelelő szintre. Ezt az előző verziókban reléekkel oldottam meg, a legújabb verzióban azonban moszfetekkel működik a rendszer. Erre azért van szükség, mert a mágnesszelepek 12 voltos feszültséggel működnek, így a mikrokontroller nem alkalmas a közvetlen meghajtásra. Egy skálán lehet a felszínen állítani a töltöttséget, a víz alatt pedig ezt állítja be a rendszer a bekapcsolás pillanatát nullának számítva. Így kommunikációs szinten nem plusz és mínusz parancsokkal lehet vezérelni a rendszert, holott ez a felhasználó által látott felületen történhet így is. Ezáltal adatvesztés esetén egy teljes töltés vagy eresztés kimaradhat, hanem egy skálával, ahol ez esetben csupán később kerül beállításra az érték. Ez a szórványosan előforduló csomagvesztés esetén is stabil megoldást jelent.



20. ábra - Mágnesszelep

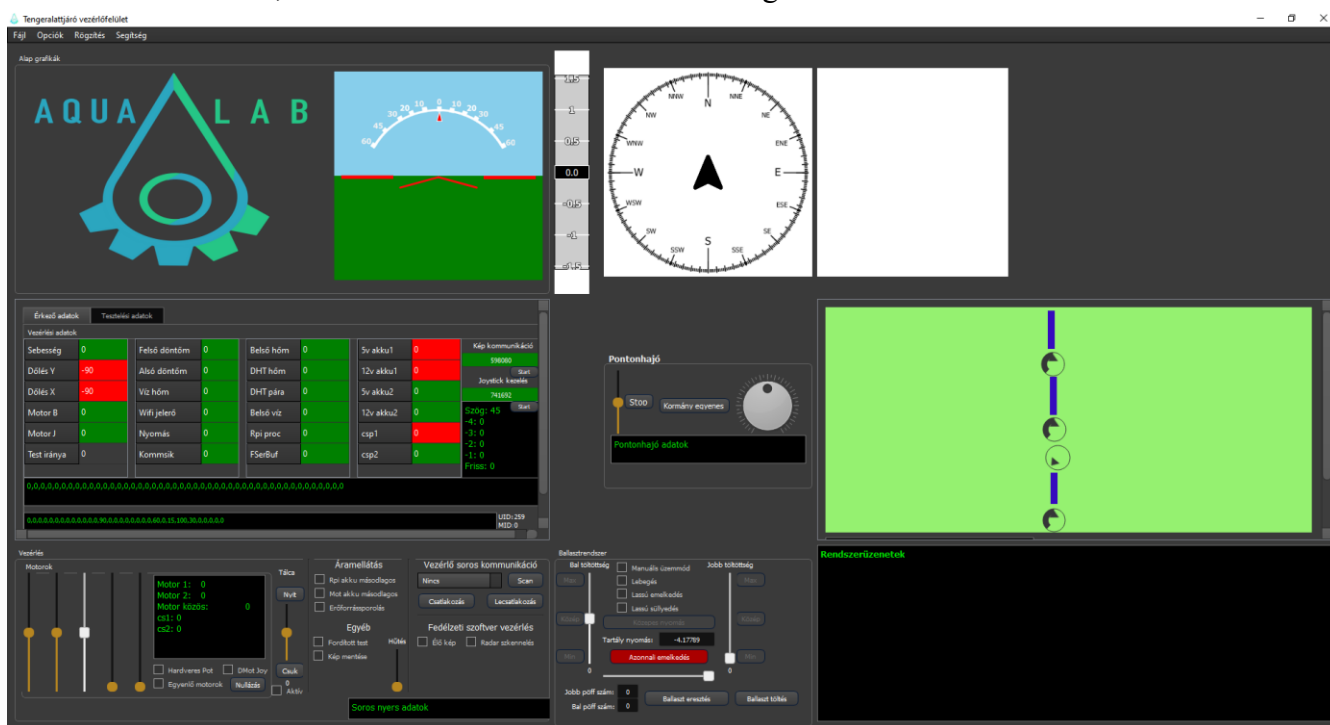
Szintén a mikrokontroller állítja be a motorok forgásának sebességét és irányát is. A PWM kimenetek és a Servo könyvtár segítségével a négyszögjelek impulzus időtartama állítható, így az ESC-k megfelelő értéket kapnak.

A program méri a belső hőmérsékletet és a páratartalmat egy DHT11 modullal, mivel ezáltal időben felfedezhetőek bizonyos hibák, például az akkumulátorok túlhevülése, vagy szivárgás. Emellett a rendszeren belül található egy vízmagasságmérő, ami furcsán hangzik hiszen elméletileg nem kerülhet a test belsejébe víz. Vészhelyzet esetén azonban ezt az adatot is felhasználva időben lehet detektálni a nagyobb mennyiségű vizet a rendszerben.

Fontos feladata ezen felül az Arduino-nak, hogy a biológiai szenzorok, a jelenlegi verzióban az átlátszóság, és pH szenzorok adatait mérje és értelmezze. A felszínre így már a konkrét zavarosság vagy kémhatás érték kerül.

A vezérlő program

A felszínen egy saját fejlesztésű grafikus programon keresztül lehet a rendszerrel kommunikálni. Ez akármilyen Windows operációs rendszerű, x64-es architektúrájú számítógépen futtatható, amelyiken található Ethernet csatlakozó. A fordító kompatibilis Linux rendszerekkel is, azonban arra nem készítettem csomagokat.



21. ábra - A vezérlőfelület

A programot a C++-ra fejlesztett Qt keretrendszerben fejlesztettem, ami megkönnyíti a fejlesztést, és kiegészíti a nyelv néhány, programozói szemmel nehezen kezelhető részét, például a dinamikus memóriakezelést, emellett maga a felület grafikusan fejleszthető. A grafikusan beállított paraméterek azonban kóddá alakulnak, így nem veszít a program a gyorsaságából. Ebből következik, hogy a mögöttes logika viszont programozásra kerül, ezáltal nagy szabadságot adva a fejlesztőnek.

A program egy külön szálon olvassa be és küldi az internetes protollokon keresztül az adatokat, ezáltal folyamatosan naprakész információkkal dolgozik a rendszer. A felületen lévő adatokat a rendszer 20 milliszekundumonként frissíti a küldendő csomag összeállításával együtt.

A felületen helyet kap az elő kamerakép is, ezt egy segédprogram segítségével megadott elérési útvonalhoz menti a rendszer, a grafikus program onnan frissíti folyamatosan, körülbelül

15 képet megjelenítve másodpercenként. Ezt a megoldást azért választottam, mivel a C++ képkezelése bonyolult, ezzel pedig jelentős időt takarítottam meg.

Emellett található egy mesterséges horizont is, amely a fedélzeti giroszkóp adatait használja a helyes dőlés meghatározására. Ez egy saját fejlesztésű widget, amelyet a paraméterek beállításával lehet mozgatni, ezáltal jól láthatjuk, a tengeralattjáró dőlését. Más hasonló widgetek is találhatóak a felületen, melyek a tengeralattjáró merülési mélységét, és azt mutatják, hogy a test orra merre áll.

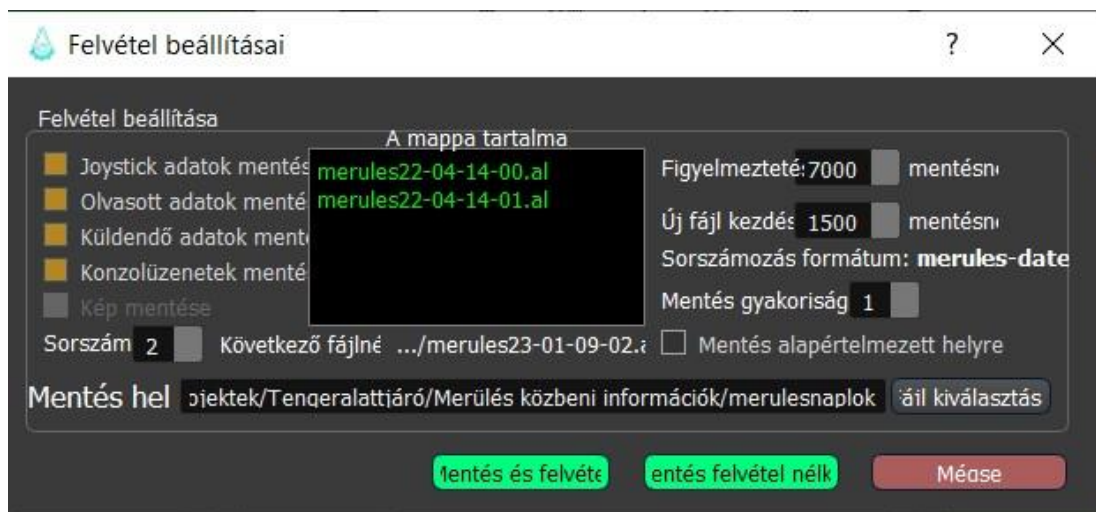
Ezután egy táblázatban az összes fedélzetről érkező adatot nyersen olvashatja el a kezelő, alatta pedig a beérkező és a kimenő adatok listáját feldolgozás nélkül is láthatjuk.

A vezérlési felületen beállíthatunk számos paramétert, a motoroktól az áramellátásig. A ballasztrendszer vezérlésénél számos lehetőséget készítettem, mivel nem tudtam, hogy mivel a legkönnyebb irányítani. Egyetlen be- és kieresztést egy gomb megnyomásával érhetünk el, a csúszkák húzásával viszont adott esetben túl sok ideig lehet nyitva a mágnesszelep, ami problémát okozhat, ezért a potméterek használatát le lehet tiltani. Ennek ellenére a hirtelen műveleteknél szükség lehet a használatukra. A legfontosabb adat itt a „pöff szám”, ami a rendszerbe a valóban eresztett löketek mennyiségét mutatja meg, vagyis az adatot a fedélzeti Arduino szolgáltatja.

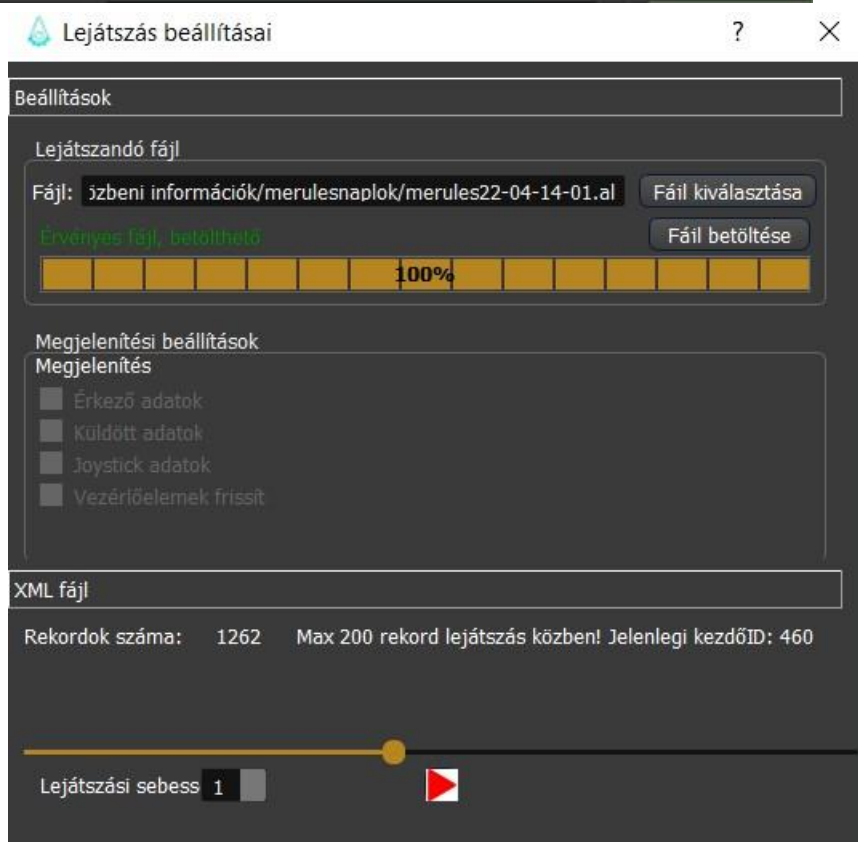
A rendszerüzeneteket tartalmazó szövegdobozban figyelmeztetéseket és hibakeresési üzeneteket lehet megjeleníteni.

A felület többi része nincs használatban. Itt kezdtem el kísérletezni például a robotkar irányítórendszerével, ami vezérlés szinten megfelelően működött a hardveres problémák ellenére. Az egyéb funkciók a menüből érhetőek el. Egyik fontos lehetőség az, hogy kiadhatunk szöveges parancsokat is, ezáltal a program működését lehet befolyásolni, valamint meg nem jelenített adatokat lehet kinyerni. Ez a parancssori funkció már jelenleg is 20 különböző parancsot kezel, amit a jövőben tervezek kibővíteni amennyiben a funkcionalitást tekintve szükség van rá.

A merülések során rendkívül sok adat keletkezik, melyeket az irányító merülés közben nem tud teljeskörűen megérteni. Emiatt készítettem el a mentés lehetőségét a rendszerben, melyet egy külön ablakon keresztül lehet elindítani. Ez egy XML szabványú fájlba menti az érkező és küldött adatokat egyaránt. A rendszer a nyers adatokat menti. Az adatbiztonság miatt, valamint a dinamikusan használt memória végeessége miatt az információkat bizonyos előre beállított időközönként menteni kell fájlba. A gyakoriságot a számítógép hardvertulajdonságai szabják meg, de a felhasználó állíthatja be.

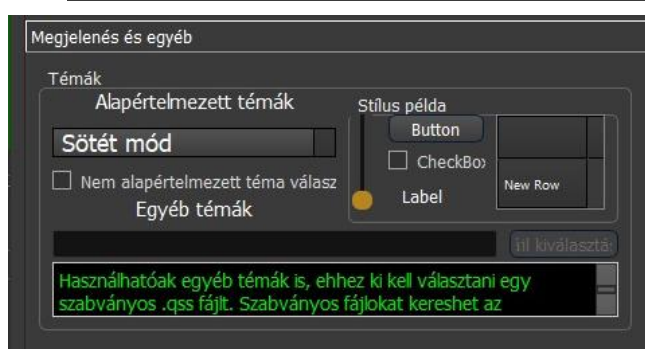
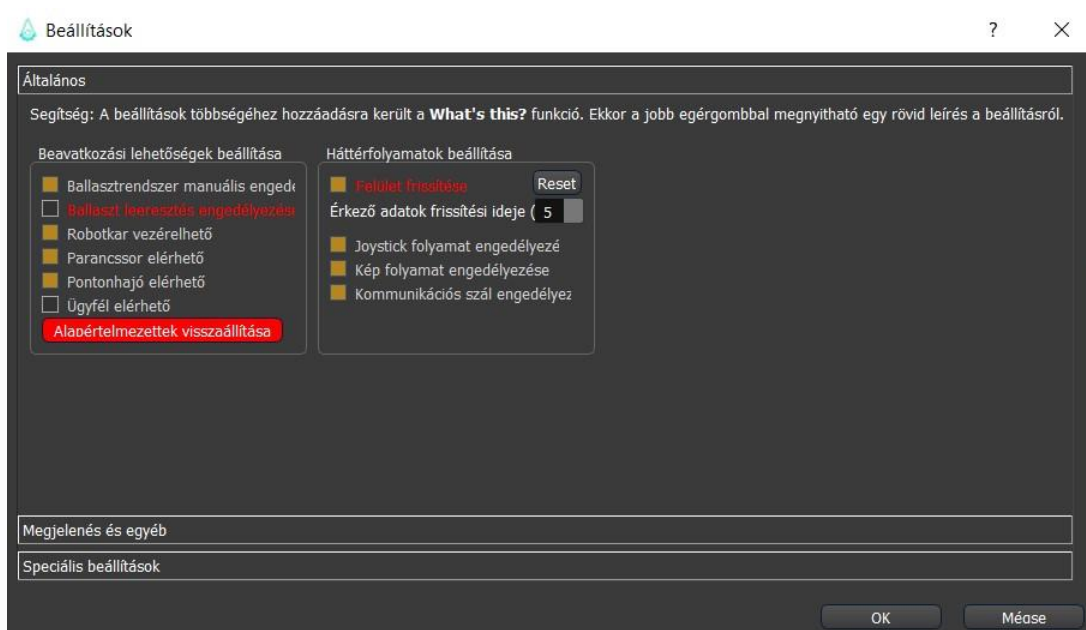


A visszajátszás szintén külön ablakban történik meg, ahol egy lejátszóeszköz érhető el, amiben a valós idejű visszajátszó funkció mellett egy csúszkával is állítható a fájlban belüli navigáció. Emellett a beolvasandó fájlt kell kiválasztani, amelyből a rendszer 1000 mentést alapból betölt. A fájlok rendszerint ennél több adatot tartalmaznak. Amikor egy nagy fájlt kell beolvasni csak egy



22. ábra - A rögzítő és a lejátszó felület

kis részét értelmezi a program, és amikor a lejátszásnál új adatokra van szükség, a beállított idő előtti és utáni adatokat is beolvassa a program. A felhasználó számára ez folyamatos lejátszás. A funkció használatakor a program közel ugyanúgy érzékeli az érkező adatokat, mintha a kommunikáción keresztül érkeznének, így a grafikus elemek mellett minden egyéb funkció is működik a beavatkozási lehetőségeken kívül. A funkció segítségével a biológiai adatokat lehet elemezni, illetve az esetleges hibákat lehet kielemezni.



23. ábra - A beállítások

A felületen elérhető egy beállításokat tartalmazó ablak is, ahol a felületen megjelenő adatok megjelenítésének engedélyezése és letiltása történhet meg, illetve a program kinézetét is megváltoztathatjuk az előre beprogramozott stílusok, vagy saját fájl alapján. A program a beállítható adatokat

a számítógépen tárolja, ezzel is egyszerűsítve a felület kezelését.

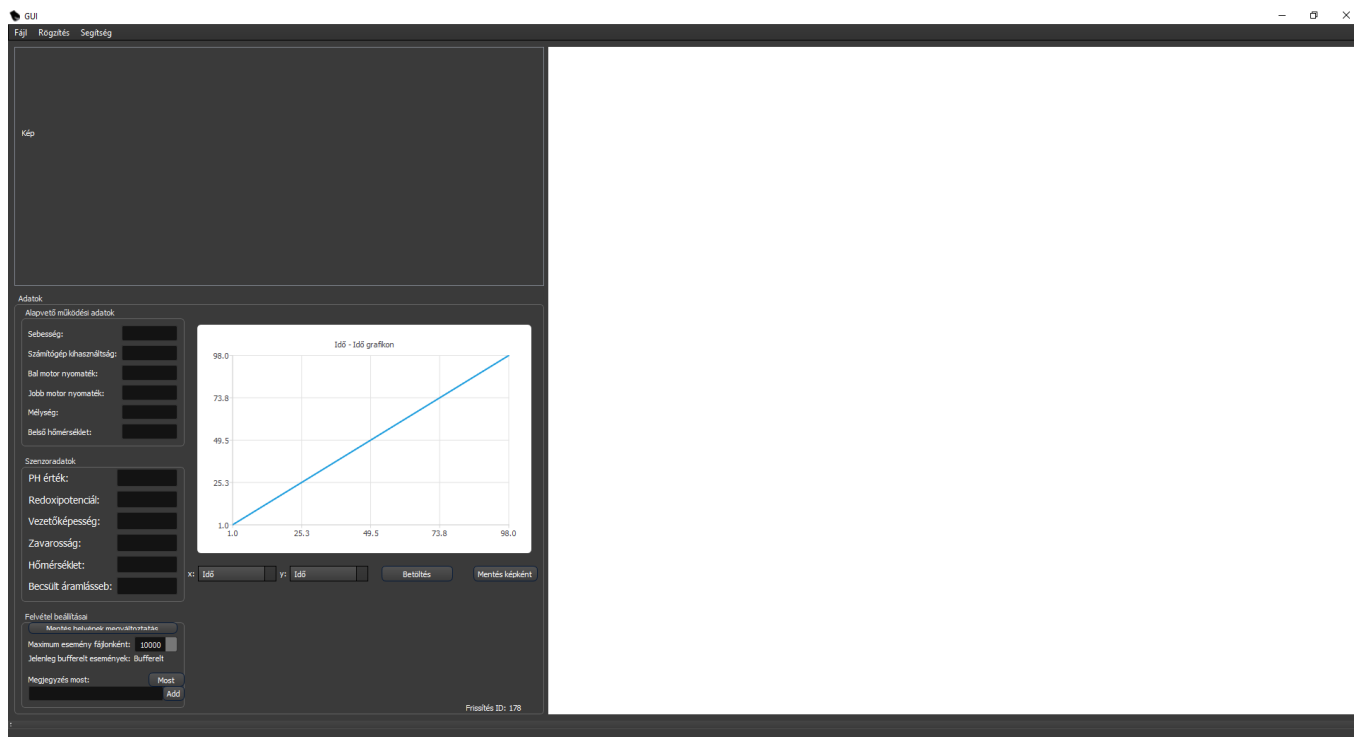
A vezérlés megkönnyítése érdekében egy joysticket, és egy hardveresen használható potmétert fejlesztettem. A gázkarként funkcionáló, Arduino nano-ra épülő mini kiegészítő USB-n keresztül, soros aszinkron kommunikációval adja át az adatait a vezérlő programnak. A Joystick három tengelyének adatai is USB-n keresztül érkeznek a számítógéphez. Erre az egerrel végzett műveletek kiváltása miatt volt szükség.

Megrendelők számára készített felület

A biológiai mérések elemzésére egy külön felületet készíték, mivel a biológusok számára szeretném könnyebben elemezhetővé tenni az adatokat, mind mérülés közben, mind utólag. A rendszer egy adatmegjelenítő részt tartalmaz, ahol az élő adatokat lehet megtekinteni. Ez alatt egy grafikon található, ahol a rendszer megjeleníti a kívánt adatokat, melyeket egymás függvényében is vizsgálhatunk. Ezáltal rengeteg különböző típusú elemzés készíthető. Fejlesztés alatt áll egy interaktív térkép az adatok még hatékonyabb eléréséhez.

A program elérhető angol és magyar nyelven egyaránt. A beágyazott fordítási támogató rendszerrel egyszerűen fordítható a program rövid idő alatt akár más nyelvekre is.

A felület hasonló menüvel rendelkezik, mint a vezérlő program, itt is van lehetőség rögzíteni és lejátszani, valamint beállításokat eszközölni. A két program által készített mentések kompatibilisek egymással megkönnyítve az adatok feldolgozását.



24. ábra - A megrendelői felület fő ablaka

Kommunikáció

A tengeralattjáró és a vezérlőeszköz között folyamatos kapcsolatra van szükség a vezérléshez, ezt a rendszerben használt egyéb kisebb távolságú és bonyolultságú kommunikáció egészíti ki.

A rendszerben néhány szenzor (pl. giroszkóp, mágnesesség mérő) I2C (Inter-Integrated Circuit) kommunikáció segítségével tartja a kapcsolatot az olvasó master eszközzel. Ez a kis hatótávolságú szinkron, többvezérlős kommunikáció kifejezetten a beágyazott áramkörüi kommunikációra használható. Az egyéb szenzorokból kinyert adatok analóg jelek segítségével jutnak el a vezérlőkhöz.

A fő kommunikációs rendszer az UTP kábel köré épül, segítségével a vezérlőegység számítógépe és a fedélzeti Raspberry pi számítógép között létrehozható egy magánhálózat, mely az IP-protokollt használja. Ezáltal a csomagtovábbításra használhatóak a TCP/IP réteg protokolljai is. Én az UDP-t használtam a szállítási rétegben, mivel a TCP-hez képest gyorsabb,

ami kisebb átviteli időt, és gyorsabb reagálást eredményez. A sikeres visszacsatolás hiányát pedig fel lehetett dolgozni a nagy mennyiségű adatcsomagok segítségével. Ennek alapján az adatok úgy kerültek összeállításra, hogy egy-egy kiesett adatcsomag ne okozzon problémát (ld.: ballasztrendszer vezérlése), azonban a folyamatos kimaradás észlelhető lehessen.

Az adatcsomagok szöveges formában kerülnek elküldésre, egy szabvány formátumban, mely vesszővel választja el a kizárólag tört, vagy egész számokat tartalmazó adatokat. Ennek határozott előnye, hogy hálózati átvitel figyelő programok használata esetén emberi szem számára is olvashatóak az adatok. A kommunikáció könnyen bővíthető új adatok hozzáadásával, amelyre a fejlesztés során számos alkalommal sor került. A zárt hálózatnak köszönhetően, melyre nem lehet csatlakozni vezeték nélkül, nincsen szükség titkosításra, ráadásul ez lassítaná az adatátvitelt.

A biológusoknak készített megrendelői felület is képes élőben olvasni az adatokat, ez azonban már egy vezeték nélküli hálózaton keresztül történik, melyet vagy a vezérlő számítógép hoz létre, vagy egy tetszőleges router. Ezen keresztül szintén UDP segítségével továbbítja az adatokat a vezérlőegység. A szeparáltságnak köszönhetően a rendszer továbbra is biztonságos marad, mivel a vezérlési adatok továbbítása továbbra is a zárt hálózaton keresztül történik, ahova kívülről érkező adatokat nem továbbít a rendszer. Ez a továbbítási rendszer még fejlesztés alatt áll, itt szükséges lesz például az adatok titkosítása.

Hasonló projektek elemzése

Természetesen nem én fedeztem fel először a robot tengeralattjárókban lévő potenciált. Mint sok más technológiát, ezt is a hadiipar igényei alakították, miután a civil szférában is felfedeztük benne a lehetőséget. A katonai feladatokra a víz alatti robotok az 1990-es évektől hivatalosan is jelen vannak többek között az Amerikai Egyesült Államok Haditengerészetének flottájában. Fontos kiemelni a Boeing és az US Navy együttműködésében fejlesztett Orca projektet. Ez egy pilóta nélküli tengeralattjáró, mely autonóm módon képes feladatokat elvégezni. Emellett a kínai haditengerészet hajóosztályát a HSU-001-et is meg kell említeni, melyről a szokásosan kevés információ is alig áll rendelkezésre, ezek alapján az amerikai fejlesztésekhez hasonló katonai platformról beszélhetünk.

Még érdekesebbek számomra a kereskedelmi és a kutatási célú fejlesztések, hiszen ezen a területen dolgozom én is. Egy olyan robot tengeralattjáró, mely egy kamerával rendelkezik, mindössze 1 millió forintos ár körül megvásárolható. Az ilyen típusú robotok nem, vagy alig

bővíthetőek, így kutatásra nem alkalmasak, de a mozgásukhoz használt technológiát teszteltem az első prototípusomon.

A kereskedelmi forgalomban elérhető termékek másik véglete a csillagászati árú, nagyon professzionális robotok, melyek óceánkutatásra, mederkutatásra és víz alatti karbantartásra használhatóak. Áruk millió euróban mérhető, így egy biológiai kutatóintézet számára nem finanszírozható kiadás.

Magyarországon a világ többi részéhez hasonlóan zajlik a kutatás a víz alatti robotok témájában: a Miskolci Egyetem által vezetett projekt, a UNEXUP keretein belül egy olyan robotot fejlesztettek a kutatók, mely a víz alatt lévő bányák világát térképezte föl egy nemzetközi projekt keretein belül. A robot specifikus probléma megoldására készült, de a technológia hasonló az általam alkalmazotthoz. Az Óbudai Egyetemen az OUBOT csapat pedig tengeri alkalmazású, autonóm robotot fejleszt.

Az inspirációt a fent említett projektekből is szereztem, az irodalomjegyzékben számos ehhez kapcsolódó forrást találhatnak.

További tervek

A továbbiakban szeretnék egyéb biológiai szenzorokat is a rendszerhez kapcsolni, hogy minél több értékes információ álljon rendelkezésre. Ezt a kutatók visszajelzései alapján tervezem megtenni. A terveim között szerepel egy iszapminta vételező egység megépítése is, mely igen nagy előrelépést jelentene. Ez a korábban tervezett robotkar által inspirált, átalakult terv, melyet a piackutatásom során pozitívan értékeltek.

A technikai rendszereket szeretném minél megbízhatóbbá tenni, ez természetesen folyamatos fejlesztésekkel jár. A test strapabíróságán, kiegyensúlyozási lehetőségein is érdemes lesz a jövőben fejleszteni. Emellett a biológusok számára készült szoftvert tervezem befejezni.

Összegzés

A projektem során rengeteget tanultam, ezt a tudást a későbbiekben is hasznosítani tudom. A projekt során az egyik legfontosabb célom az volt, hogy a tervezés és megvalósítás során minden feladatot én végezzek el, és ezáltal egy projekt összes szegmensében tapasztalatot szerezzek, így a technikai tudás megszerzésén felül a projekt menedzselési területein is hasznos tapasztalatra tettem szert.

A fejlesztést több mint két évig végeztem, ami alatt sokak rendkívül fontos visszajelzésekkel tereltek az úton, melyek sokat segítettek a projektemben.

Köszönetnyilvánítás

Elsőként köszönöm a támogatást a családomnak az egész projekt alatt, és Dr. Gábris Katalinnak nagymamámnak a lektorálásért, és a csomagátvételért!

Köszönöm a mentoraimnak, Dr. Komáromi Annamária tanárnőnek a kezdetek óta tartó támogatást, illetve Dr. habil. Wühl Tibor tanár úrnak az értékes iránymutatást!

Köszönöm a segítséget a ballasztrendszerrel, és a hajótesttel kapcsolatos fejlesztésben osztálytársamnak, Tasnádi Bencének.

Köszönöm a második prototípus fejlesztésében nyújtott óriási segítségét Német-Tibor István hajóépítőnek!

Köszönöm a Budapest Innovation Group tagjainak támogatását!

A projekt a Mol Alapítvány, és a Nemzeti Tehetség Program pénzügyi támogatásával valósult meg, ezúton is köszönöm a támogatást!



Irodalomjegyzék

Kutatáshoz használt irodalom:

- http://www.hisutton.com/Chinese_LDUUV.html
- <https://www.navy.mil/Press-Office/Press-Releases/display-pressreleases/Article/2538616/navy-marine-corps-release-unmanned-campaign-plan/>
- <https://www.forbes.com/sites/lorenthompson/2022/06/02/navys-robotic-orca-submarine-could-be-a-gamechanger-in-the-pacific/>
- <https://thediplomat.com/2019/11/china-enters-the-uuv-fray/>
- <https://www.emag.hu/chasing-gladius-mini-dron-viz-alatti-4k-uhd-kamera-maximalismelyseg-100-m-hatotav-2-ora-hatizsak-gmini/pd/DJ6ZMPMBM/?cmpid=87202>
- <https://unexup.eu/>
- <https://adgeo.copernicus.org/articles/54/109/2020/>
- <http://oubot.ekik.uni-obuda.hu/>

A fejlesztéshez használt források:

- <https://www.qt.io/>
- <https://www.arduino.cc/>
- <https://www.python.org/>
- <https://en.cppreference.com/w/>
- <https://infoc.eet.bme.hu/>
- <https://stackoverflow.com/>
- <https://bluerobotics.com/>
- <https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml>
- <https://datatracker.ietf.org/doc/html/rfc8200#page-27-28>