

SZAKDOLGOZAT

Agárdi Róbert Flórián
Gépészmérnök BSc.

2024



Magyar Agrár- és Élettudományi Egyetem

Kaposvári Campus

Műszaki Intézet

Gépészmérnök alapképzési szak

Gépjármű-adatgyűjtő tervezése mikrokontrollerrel

Belső konzulens: Dr. Gergely Zoltán Albert
egyetemi adjunktus

**Belső konzulens
intézete/tanszéke:** Műszaki Intézet
Járműtechnika Tanszék

Külső konzulens: Vizeli István András
egyéni vállalkozó

Készítette: Agárdi Róbert Flórián
P65O9Z
Nappali tagozat

Kaposvári Campus

2024

MŰSZAKI INTÉZET GÉPÉSZMÉRNÖK ALAPSZAK
Gépjárműtechnika specializáció

SZAKDOLGOZAT

feladatlap

Agárdi Róbert Flórián (P6509Z)

részére

A szakdolgozat címe:

Gépjármű-adatgyűjtő tervezése mikrokontrollerrel

Feladatkiírás:

Gyűjtse össze és értékelje a témával kapcsolatos szakirodalmat! Méretezzen az adott járműhöz érzékelőket és tervezze meg a mérés programját! Végezzen több mérést és értékelje ki a mérési eredményeket! Szükség esetén tegyen javaslatot a konstrukció javítására, fejlesztésére!

Közreműködő tanszék: Járműtechnika Tanszék

Külső konzulens: *Vizeli István András, gépészmérnök, egyéni vállalkozó*

Belső konzulens: *Dr. Gergely Zoltán Albert, egyetemi adjunktus, MATE, Műszaki Intézet, Járműtechnika Tanszék*

Beadási határidő: 2024. november 5. (kedd) 12.00 óra

Gödöllő, 2024. június 15.

Jóváhagyom


Prof. Dr. Kiss Péter
tanszékvezető


Prof. Dr. Szabó István
szakfelelős

Átvettem


Agárdi Róbert Flórián
hallgató

A dolgozat készítőjének külső konzulense nyilatkozom arról, hogy a hallgató az előre egyeztetett konzultációkon megjelent.

Gödöllő, 2024. 11. hó 04. nap


(külső konzulens)

Tartalomjegyzék

1. Bevezetés, célkitűzés	3
2. Adatgyűjtés a gépjárművekben	5
2.1. Adatgyűjtés célja a járműiparban	5
2.1.1. Az adat definíciója	5
2.1.2. Járművön belüli adatgyűjtés	5
2.1.3. A jármű diagnosztikáról	6
2.1.4. Jármű diagnosztika napjainkban	6
2.2. Alkalmazott műszaki megoldások	6
2.2.1. Az Off-board diagnosztika	6
2.2.2. Az On-board diagnosztika	7
2.2.3. A fedélzeti diagnosztika felépítése	8
2.2.4. Buszrendszerek	9
2.3. Mikrovezérlők és az Arduino	10
2.3.1. Mikrovezérlők fejlődés története	10
2.3.2. Az Arduino megjelenése	12
2.3.3. Az Arduino fejlesztőkörnyezete	12
2.4. Érzékelők	14
2.4.1. Szenzorok definíciója	14
2.4.2. Csoportosításuk	14
2.4.3. Releváns érzékelők bemutatása	15
2.4.4. Hőmérséklet szenzorok	15
2.4.5. Fordulatszám szenzorok	16
2.4.6. Feszültség és áramerősség mérők	16
2.4.7. Rezgés, kopogásérzékelők	17
2.5. Méréstechnika	17
2.5.1. Mi a mérés?	17
2.5.2. Mérési folyamat	18
2.5.3. A mérések bizonytalansága	18
2.5.4. A mérés pontosítása	19
2.5.5. Mérési sorozatok	20
2.5.6. Mérési bizonytalanság kifejezése	20
3. Konstrukció kialakítása	22
3.1. Vizsgált jármű bemutatása	22
3.2. Tervezés, méretezés, programozás	23
3.2.1. A hardverek kiválasztása	24

3.2.2.	Szenzorok és a kiegészítő elemek méretezése.....	25
3.2.3.	Az elektronika tokozása.....	37
3.2.4.	A mérés és kiértékelés programterve.....	39
3.3.	Fizikai megvalósítás.....	42
3.3.1.	Az elektronika összeépítése.....	43
4.	Eredmények, konklúzió.....	Hiba! A könyvjelző nem létezik.
5.	Összefoglalás	55
6.	Summary	56
7.	Irodalomjegyzék	58
8.	Nyilatkozatok	60
9.	Mellékletek	62

1. Bevezetés, célkitűzés

Az információ gyűjtés korunkban igencsak meghatározó tevékenység. Az élet minden területén adatokat gyűjtünk, elemezzük azokat és optimalizáljuk a hozzájuk kapcsolódó folyamatokat. Az információ valós idejű gyűjtése az iparban, gyártásban és a járműtechnikában talán mondható, hogy kifejezetten fontos mivel bizonyos folyamatokra azonnal kell reagálnunk személyesen vagy valamilyen erre a célra fejlesztett eszközzel, mint például a mai gépjárművekben használt központi vezérlőegységek.

Ha az előbbi példát vesszük alapul, akkor könnyen belátható, hogy a bizonyos esetekben a motorvezérlő egységnek korrigálnia, irányítania és vezérelni kell a jármű egyéb egységeit annak érdekében, hogy a motor a megfelelő üzemben működjön, a károsanyag kibocsátás ne legyen túl magas, elégséges legyen az égés minősége. Ahhoz, hogy a vezérlőegység tudja hova kell beavatkoznia érzékelőkre van szükség, ahonnan az információt kapja például légmennyiség mérő, fordulatszám jeladó, helyzetérzékelők, és így tovább. A beérkezett információ-áradat alapján a vezérlő beavatkozik a megfelelő helyen és probléma vagy szélsőséges helyzetekben, vagy mikor egy hibás érzékelő okozza a szélsőséges helyzetet, a gépjármű vezetője felé is jelez. Egyrészt a műszerfalán felvillan a megfelelő hibajelző piktogram, amely a probléma jellegére utal, de vannak kifejezetten hibakeresésre alkalmas készülékek, amik képesek a vezérlőegységgel valamilyen módon kommunikálni és pontosabban behatárolni a hiba okát.

Tehát a gépjárműtechnikában nagy szerepe van az információ szerzésnek a jármű zavartalan működéséhez, ahhoz hogy tudjuk milyen folyamatok játszódnak le a különböző egységekben, be tudjunk avatkozni közvetlenül, és hogy az esetleges hibákat könnyebben feltárhassuk. A diagnosztikus rendszerekben különösen nagy szerepe van az adatgyűjtésnek, hiszen az alapján mit érzékelnek a szenzorok és milyen jelet bocsátanak ki magukból egy külső vagy belső diagnosztikai rendszer megkönnyítheti és felgyorsíthatja a hibakeresést, mellyel végeredményben időt spórolhatunk meg, amivel pedig a kiadásokat lehet csökkenteni.

2023-ban Magyarországon 449 686 mezőgazdasághoz, erdőgazdálkodáshoz vagy halászathoz kapcsolódó vállalat, ebből nagyjából 13 000 társas vállalkozás, valamint közel 19 000 egyéni vállalkozás működött (*http 1, 2024.03.23.*). Az egyéni vagy családi gazdaságokban a mai napig előszeretettel használnak a 2000-es évek előtt készült gépeket,

járműveket, vontatmányokat, melyek viszonylag könnyen kezelhetőek, könnyen javíthatóak és alapvetően egyszerű gépek. Viszont pont az egyszerűségük miatt ma már inkább elavultnak mondhatóak a mechanikus érzékelők és vezérlők miatt, amik csak és kizárólag a legalapvetőbb dolgokat látják el és igazán pontos vezérlést nem tesznek lehetővé. A személygépjárműveket tekintve is az a saját tapasztalatom, hogy sokak használnak olyan járműveket amelyek nincsenek ellátva korszerű, komplex érzékelő és vezérlő rendszerrel.

Ennek okán jött az ötlet, hogy egy egyszerű, arányaiban kis beruházás keretében tervezzek, méretezzek és állítsak össze egy olyan rendszert, amely képes az elavultnak mondható járművek vizsgálatára adatgyűjtéssel, továbbá az általam beépített szenzorokkal és egy mikrokontroller technológián alapuló eszköz segítségével méréseket végzek egy járművön melyhez családilag van hozzáférésem, majd ezt valamilyen egyéb eszközön megjelenítem és kiértékelem. Az általam kitalált konstrukció elsősorban inkább a diagnosztikus megközelítést testesíti meg, tehát főként hibakeresésre, illetve a jármű megfelelő üzemének vizsgálatára szolgál érzékelések, mérések segítségével. Érdeemes megjegyezni, hogy erre is van igény és kereslet, hiszen hiába „egyszerűek” a régebbi járművek, sok esetben így is komplikáltabbak, pont a mechanikus vezérlés, érzékelés okán, mint sok mai gépjármű, sőt a mezőgazdasági, erdészeti és építőiparban használt céljárművekre a mai napig nem született egységes, szabványosított diagnosztikai egység. Így ha egy ilyen rendszerrel szereljük fel az idősebb gépeket, további időt spórolhatunk meg a hibakeresésen és a szerelésen, mellyel akár a gép állásával járó bevételkiesést is csökkenthetjük.

2.1.3. A jármű diagnosztikáról

Az alapvető üzemelés és vezérlés mellett, a diagnosztika és üzemfenntartás kulcsfontosságú szereplője lett az információ szerzés. Minél többet tudunk egy adott jármű egységeinek állapotáról, annál könnyebben és gyorsabban behatárolható egy esetleges hiba forrása, mellyel pénzt és időt spórolhatunk meg. A diagnosztika lényege, hogy egy gép vagy jármű állapotáról úgy szerezzünk információt, hogy azt ne kelljen komolyabb bontás alá vetni, tehát, olyan mérési eljárás, mely során adatot gyűjtünk a járműről, szétszerelés mentesen. (Staudt, 1988; Mészáros és Sitkei, 1965; Ember et al, 1983) Használat következtében minden gép vagy berendezés folyamatosan amortizálódik, kopások jelentkehetnek az üzemelés közben, a kenőfolyadékok elfáradhatnak, bekoszolódhatnak, a tömítések elgyengülnek és így tovább. Ezen kismértékű hibák folyamatos teljesítmény romláshoz vezethetnek, így súlyosbítva a már meglévő problémákat vagy újakat idézhetnek elő. Ennek megelőzése érdekében fontos szinte minden helyzetben, minden időpillanatban ismernünk, hogy milyen körülmények között működik a járművünk. A diagnosztikai rendszerek a hibafeltárást hivatottak elősegíteni. Segítségükkel úgy következtethetünk a jármű állapotára, hogy azon jóformán egy csavart sem kell meglazítanunk.

2.1.4. Jármű diagnosztika napjainkban

Manapság a járműtechnikában olyan diagnosztikai rendszereket alkalmaznak, melyek a vizsgálathoz szükséges méréseket közvetlenül végzik valamilyen villamos jellemzőre vetítve. Ezeket a méréseket aztán számítógépes rendszerek dolgozzák fel, így a mérés automatizálható, az egymást befolyásoló jellemzők egyszerre figyelhetőek, a mért adatokat tudjuk tárolni, illetve azokat összehasonlítani, valamint az esetleges hibákról azonnal értesülhetünk, sőt bizonyos esetekben beavatkozási lehetőséget is kínál a rendszer, de legalább a hiba jellegét meghatározza. (Lakatos és Nagyszokolyai, 2006).

2.2. Alkalmazott műszaki megoldások

A gépjárműdiagnosztikában alapvetően két csoportot különböztet meg a szakma:

- *Off-board diagnosztika*: alkalmi, külső mérőrendszeres állapotvizsgálat,
- *On-board diagnosztika*: folyamatos, vagy részben folyamatos fedélzeti rendszerek.

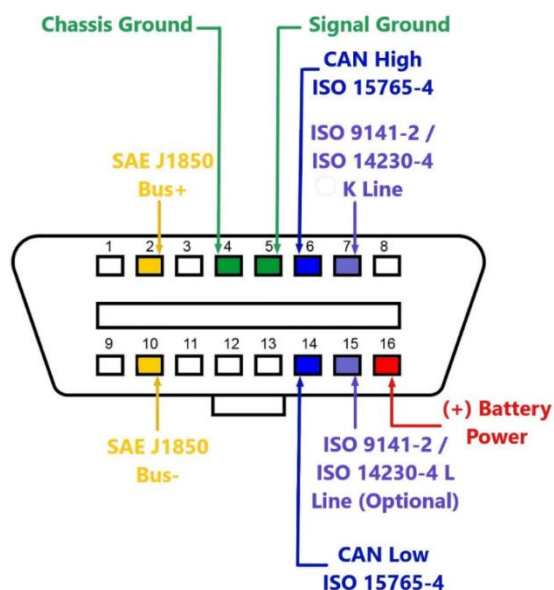
2.2.1. Az Off-board diagnosztika

Utóbbit szokás még **nem fedélzeti diagnosztikának** is nevezni, mely talán jobban kifejezi ezen állapot felmérés lényegét. A mérő berendezés, a hozzá tartozó jeladók és mérésvezérlők, valamint a kiértékelő egység a járműnek nem integrált része. Ezt az vizsgálati

metódust a múlt század első felében gyártott járműveken, főként harckocsikon és katonai járműveken alkalmazták állapotfelmérésére. Ezek a vizsgálatok jobbra olyan villamos mérések voltak jellemzően, mint például a gyújtórendszer elemeinek vizsgálata vagy a gyújtókörök vizsgálata.

2.2.2. Az On-board diagnosztika

A magyar szakirodalom **fedélzeti diagnosztika** néven is emlegeti, mely annyit tesz, hogy a fentebb felsorolt mérőberendezés, a jeladók, a különböző hardverek és szoftverek, illetve a mérésvezérlő a gépjármű szerves része. Ennek okán adott méréseket folyamatosan és periodikusan tud az egység elvégezni, nem kell hozzá különleges terepviszonyt kialakítani, nem kell szimulálni a közlekedési helyzeteket, hiszen a gépjármű normál üzeme közben is képes a méréseket végrehajtani és a keletkező információkat bizonyos mennyiségben és ideig tárolni, majd kapott adatokat kiértékelni. Manapság ez és ennek fejlesztett változatai a jellemzőek többnyire a gépjárműtechnikában. (Fekete et al, 2002) A fedélzeti diagnosztikai rendszerekhez való hozzáférést első sorban az amerikaiak kezdték el szabványosítani az 1980-as évek végén, ez lett az OBD csatlakozó, mely egy olyan szabványosított kétsoros csatlakozó melyre rácsatlakozva egy külső eszköz képes kommunikálni a gépjármű vezérlőegységével (ECU – Electronic Control Unit). Később az ISO szabványba is felvették EOBD néven és a 2000 után gyártott gépjárművekbe kötelezték az elhelyezését. (Tölgyesi, 2005) Az OBD csatlakozó kialakítása és általános lábkiosztása a 2.2. ábrán látható.



2.2. ábra: OBD II csatlakozó sematikus rajza és lábkiosztása (<http> 3, 2024.08.25.)

2.2.3. A fedélzeti diagnosztika felépítése

Jelen helyzetben számomra a fedélzeti diagnosztika a releváns példa, mert én is egy hasonló rendszer kialakítását tervezem elkészíteni, már ami az érzékelést és mérést tekintve. Először is érdemes szemügyre venni a fedélzeti diagnosztikák technikai kialakítását és gyakorlati megvalósítását. Mint fentebb leírtam a vizsgálati rendszer része, hovatovább főegysége és központi eleme, az elektronikus vezérlő egység, vagy az angol rövidítést használva ECU, melynek a feladata a különböző alegységek adatait kezelni, tárolni, továbbadni és igény szerint az OBD csatlakozón keresztül egy megfelelő külső eszközre átadni. Ezen felül az ECU felel a szabályzási és irányítási folyamatok lebonyolításáért is, szintén a kapott információk alapján. Továbbá az összefüggő folyamatok a különböző részegységekben könnyebben lekezelhető, mivel alá-fölé rendeltségi vagy hierarchikus rendszert tudunk kialakítani, mellyen egy fontossági sorrend is felállítható, tehát a lényeges dolgok, főként biztonsági berendezések, előbb kapnak jogosultságot egy menet jellemző megváltoztatására, mint a kevésbé fontos, például kényelmi folyamatok. Az ECU és a többi vezetés támogató, illetve mérő-, irányító-, szabályzó rendszer és alrendszer fontos jellemzője továbbá, hogy mikrovezérlő alapúak, lévén, hogy irányítási és valós idejű (real-time) mérések lebonyolítására ez az elektrotechnikai eszköz a leghasznosabb. (Tölgyesi, 2005; Lakatos és Nagyszokolyai, 2006) A vezérlőegységek belső felépítési struktúrája a 2.3. ábrán látható.



2.3. ábra: Gépjárművek vezérlőegységének belső felépítése (<http4>, 2024.08.26.)

2.2.4. Buszrendszerek

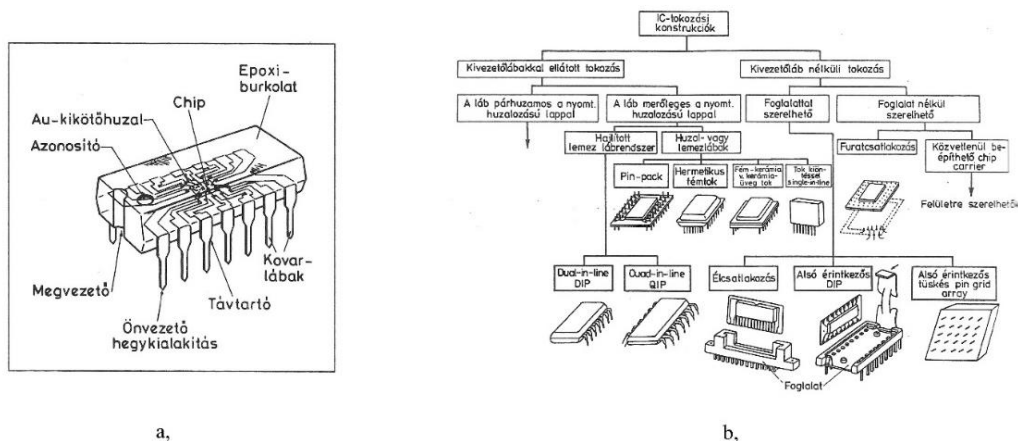
Ha a járműtechnikában adatgyűjtésről, mérésről és vezérlésről esik szó, nem mehetünk el szó nélkül a kommunikációs megoldások mellett. A járműmechatronika kezdetén a különböző vezérlő és mérő egységek a járművön belül úgy kommunikáltak egymással, hogy minden egyes egység a többivel közvetlen kapcsolatban volt. Ez rengeteg mennyiségű kábelt igényelt, ami ezzel együtt plusz súlyt és plusz költségeket is jelentett. Ennek a problémának a megoldására jöttek létre a különféle buszrendszerek. A rendszer alapja, hogy mindegy egység egy közös főszálra csatlakozik, csak úgy mint a közlekedésben az út mentén a buszmegállók, így csökkentve a járműben a szükséges vezetékek mennyiségét. Különböző besorolások, csoportok jöttek létre a buszrendszereken belül, melyek a csoportban szereplő szabványok adatátviteli sebességét határozzák meg. Ezek alapján 5 csoport alakult ki, az A csoport a lassú adatátviteli gyorsaság, a B csoport az átlagos adatátviteli sebesség, a C csoport a gyors adatáramlási sebesség, illetve a C+ és D csoportok melyek egyaránt extrém gyors adatátvitelt biztosítanak, de nem azonos mértékben. (Reif, 2014)

A buszrendszereken belül a CAN (Controller Area Network) szabványt szeretném kiemelni, a későbbiekben fontos lesz. A CAN buszrendszer létrejöttének oka, hogy több különböző alegységet és vezérlőt tudjunk csatlakoztatni a központi vezérlőegységhez. Alapja, hogy a különböző információkat bájtokban tárolja és egy hosszú adatláncként adja tovább a feldolgozó egységnek az információt. Ez az adatlánc a klasszikus CAN rendszerben három fő részből épül fel, a kódsor elején egy 11 bitből álló, úgynevezett arbitrációs mező, ez tartalmazza az azonosító értékeket. A második rész maga az adat 0-64 bit szélességben. Végezetül az üzenet végét záró bitek. Az adatbitek meghatározása a vezetékek, azaz CAN_H és CAN_L, közötti feszültség különbség vizsgálatával határozza meg. A vezetékeken fellépő különbség mindig szimmetrikus és tükörkép-szerű. Mivel a jelmeghatározás a feszültségdifferencia elvén működik, ezért a külső elektromos zavarok, például elektromágneses zaj, nincsenek hatással a jelre, hiszen azonos lesz a kitérés a két vezetéken, ezt közös modulusú zavarnak is hívják. A jelszint, ami az adatot hordozza magában, két fajta lehet, recesszív és domináns. Recesszív a jelszint, ha két vezetéken mérhető feszültségkülönbség kevesebb mint 0,5 V, ez feleltethető meg a logikai igaz értéknek. A domináns jelszint ezzel szemben a logikai hamis értéket képviseli és általában a vezetékek között fellépő 2 V-os feszültségkülönbség hatására jön létre. (http 16, 2024.10.08.; Reif, 2014)

2.3. Mikrovezérlők és az Arduino

2.3.1. Mikrovezérlők fejlődés története

A gyártástechnológia, a gépjárműipar és sok egyéb szakterületen az automatizálást egyszerű elemekkel igyekeztek megoldani. Ezen automatizálási rendszerek előnye az volt, egyszerű és könnyen előállítható alkatrészek alkották, mint például, rétegellenállások, kondenzátorok, tekercsek és relék. Azonban volt egy nagy hátrányuk, az egyszerű alkatrészek csak egy-egy feladatot tudtak ellátni, így viszonylag sok kisebb elemből lehetett komoly automatizáló kapcsolásokat készíteni, ennek okán ezek a rendszerek arányaiban nagyobb helyet igényeltek és a relék kapcsolása miatt még zajosak is lehettek. Erre a problémára érkezett elsőként megoldásként a nyomtatott vagy integrált áramköri technológia. Ennek az eljárásnak a lényege, hogy egy tokozásba kerüljenek azok a diszkrét elemek, amelyek korábban csak önálló tokozásban léteztek. Az integrált áramköri elemek alapja, a chip, amely minden logikai kapcsolatot tartalmaz, mellyel bizonyos feladatokat tudunk elvégeztetni az adott egységgel. Ezek a logikai kapcsolások félvezető alapúak, tehát diódák, tranzisztorok és FET-ek alkotják nagyrészt, de hagyományosnak tekinthető elemeket is tartalmaznak. Az így elkészített chipek több kivezetéssel rendelkeznek, a legelterjedtebb a **DIL** (*Dual-In-Line*) elrendezés, vagyis két soros láb kiosztás, melyek a tokozás két oldalán sorakoznak fel. Az úgynevezett kovarlábak alumínium kikötőhuzal segítségével csatlakoznak a chip megfelelő részeihez. (Tormai et al, 1984; Kovács, 1976) A 2.4. ábrán láthatjuk egy általános integrált áramköri elem felépítését.

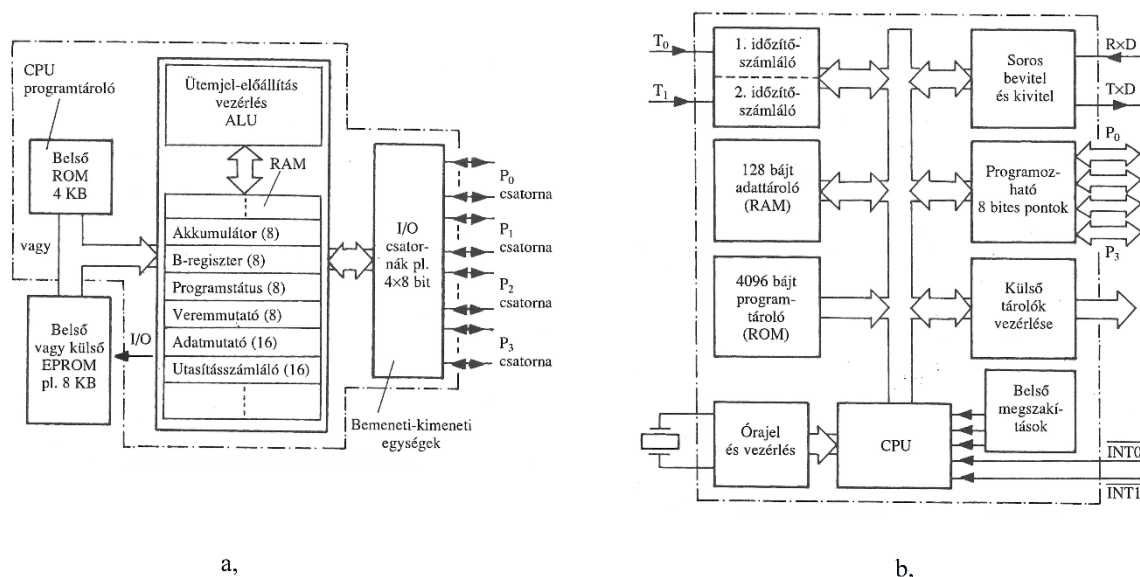


2.4. ábra: A DIL IC szerkezeti felépítése (a) és az IC tokozások és lábkosztásaik (b) (Saját szerkesztés, Mojzes, 1995 alapján)

Az integrált áramköri technológia két fő termékcsoportja a mikroprocesszorok és a különféle memóriák. A mikroprocesszor dolga az utasítások kezelése, kiadása és különféle számítások elvégzése, valamint a programfuttatás. A mikroprocesszor önmagában képtelen a

működésre, ezért egyéb külső elemekre van szükség, hogy automatizálási vagy szabályzási feladatokat végezhessünk a segítségével. Ilyen további külső egység például az órajel képző vagy a memóriák. A memóriák két fő része bonthatóak, vannak egyrészt a csak olvasható memóriák (**ROM** - Read-Only Memory), másrészt pedig az írható-olvasható memóriák (**RAM** - Random-Access Memory). Szokás ezeket továbbá nem illékony és illékony memóriáknak is nevezni. (Mojzes et al, 1995; Tormai et al, 1984)

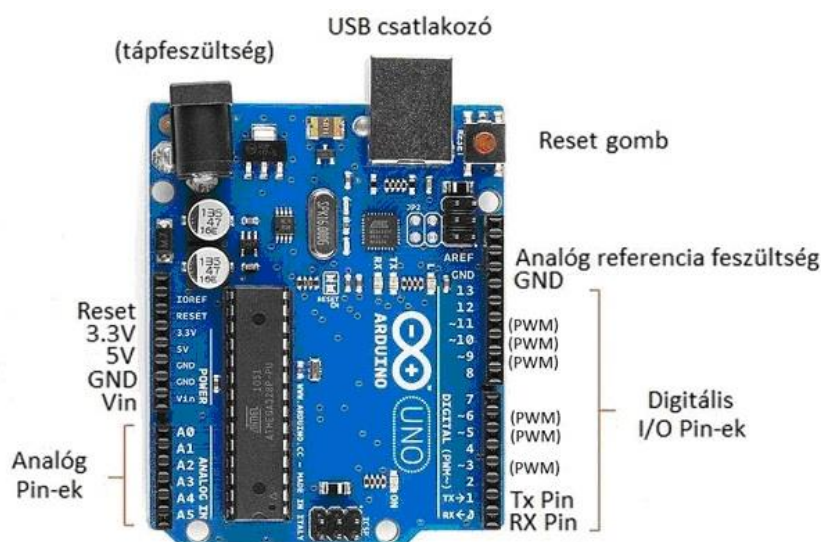
Korábban a vezérlési és szabályzási feladatokat mikroprocesszoros architektúrával oldották meg, viszont ezekhez a processzorokhoz minden esetben kellett más integrált áramköri elemek, mint például memóriák, órajelek, stb. A technológia fejlődésével egyre kisebb méretben volt lehetőség gyártani a különböző integrált áramköri egységeket, melynek eredményeként jöttek létre a komplett „mikroszámítógépek” vagy mikrovezérlők, melyek egy tokozásban, egy lapkán tartalmazzák a processzort, memóriát, óra jelet és egyéb elektronikai elemet, melyeket korábban külön egységként kellett szerelni. A mikrovezérlők olyan egylapra integrált áramköri egység, melyek mintegy célszámítógépként működnek. Előre felprogramozott paraméterekkel egy eszközként végzi el a kijelölt vezérlési és szabályzási feladatokat különböző szabványok, szabályok mentén. (Mojzes et al, 1995) A 2.5. ábrán látható milyen részegységek alkotnak átlagosan egy mikrokontrollert.



2.5. ábra: A mikrokontrollerek blokkvázlata (a), illetve be- és kimeneti csatlakozásai (b)
 (Saját szerkesztés, Fekete et al., 2002 alapján)

2.3.2. Az Arduino megjelenése

A fejlődés során a mikrokontrollerek széleskörben elterjedtek vezérlő és szabályzó rendszerek kiépítéséhez. Ennek apropójára 2003 és 2005 között Massimo Banzi és Tom Igoe egy olyan nyíltforráskódú, mikrovezérlő alapú eszköz létrehozásán kezdett el dolgozni, mely egyszerű, könnyen kezelhető és relatíve olcsó, ez lett az Arduino. Tehát az Arduino egy márkajelzés, melynek keretein belül mostanra többféle mikrokontroller alapú, szabadon használható és programozható (C, illetve C++ programozási nyelveken) eszközt, valamint kiegészítőt kínálnak. ([http 2, 2024.06.23.](#)) A fejlesztést elmondásuk szerint azért kezdték, hogy bátorítsák a diákokat, a hobbi elektronikusokat, valamint a kreatív embereket arra, hogy tervezzenek saját maguknak beépített eszközöket a lehető legkevesebb befektetett energiával és idővel. Az egyik legelterjedtebb változat, ami a márka égisze alatt készül, az **UNO**. Ezt használják a legtöbben, éppen ezért rengeteg alapprogram és bekötés található ehhez a konkrét típushoz. Az Arduino UNO felépítése a 2.6. ábrán látható.

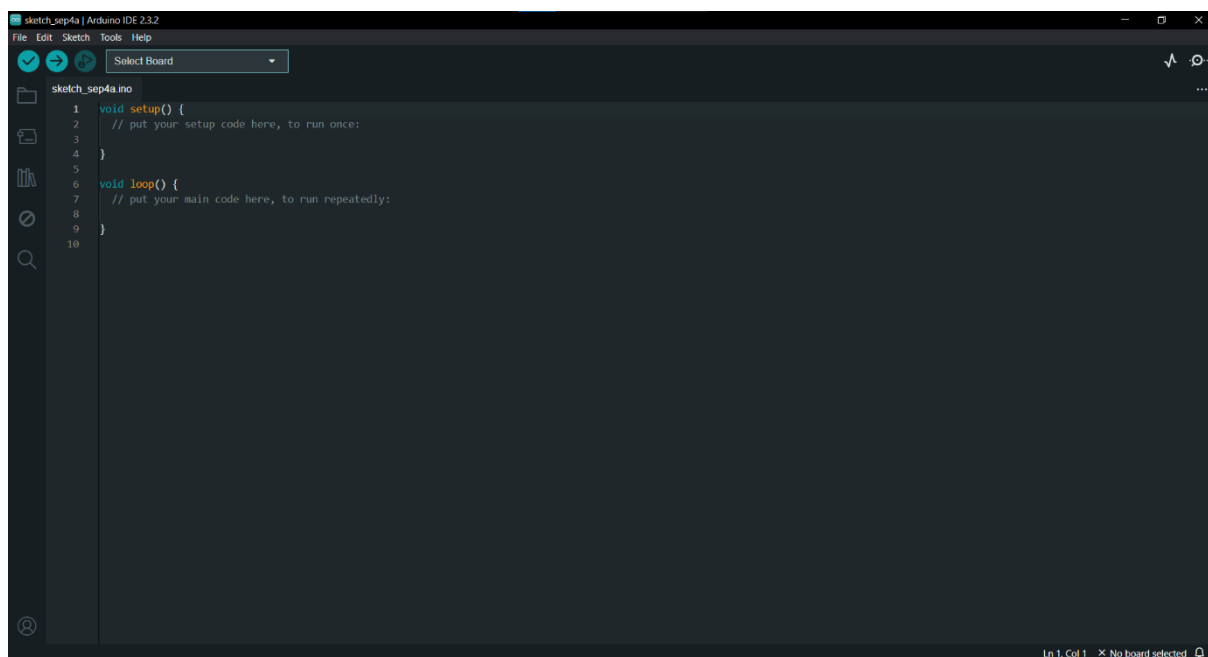


2.6. ábra: Az UNO platform felépítése és lábkiosztásai
([http 2, 2024.06.23.](#))

2.3.3. Az Arduino fejlesztőkörnyezete

Bármelyik típust is választjuk az Arduino kínálatából egy projekt megvalósítására, mindene esetben a cég által készített fejlesztő környezetet az Arduino IDE-t használhatjuk. Az IDE rövidítés az angol *Integrated Development Environment* (Integrált Fejlesztői Környezet) kifejezésre utal. Ez a program ingyenesen elérhető bárki számára, ezzel is könnyítve és segítve azokat, akik ilyen eszközökkel akarnak dolgozni, akár kisebb, akár nagyobb projekteken. A fejlesztőkörnyezet küllemét a 2.7. ábra szemlélteti.

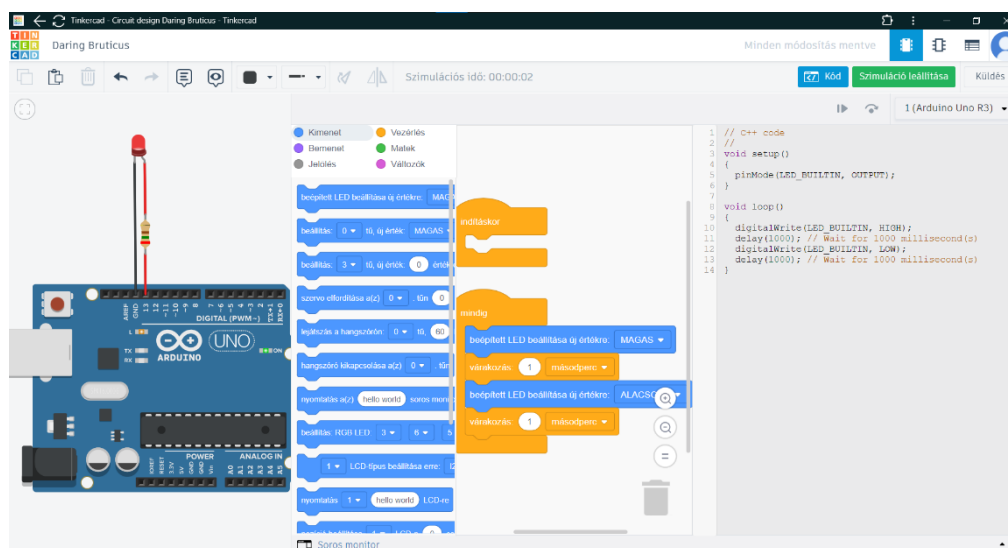
Megfigyelhető, hogy az Arduino eszközeire a programkód két részletben kerül fel. Az első az úgynevezett **Setup**, vagy beállítás rész, ide kerül a kódnak az a része, melyet az eszköz elindulásakor csak egyszer szeretnénk lefuttatni. A második a **Loop**, vagy ciklus rész, melyben olyan parancssort adhatunk meg, ami ha végig fut, újraindul mindaddig, amíg le nem állítjuk az eszközt. Utóbbi rész alkotja a programkód magját, ide kerül esetemben a mérésekhez kapcsolódó kód részlet is.



2.7. ábra: Az Arduino saját fejlesztőkörnyezete (*Saját kép*)

A fejlesztőkörnyezeteknél érdemes szót ejteni a Tinkercad nevű alkalmazásról is. A program egy weblapként indult 2011-ben. A weboldalt egy volt Google alkalmazott, Kai Backman és társa, Mikko Mononen készítették, abból a célból, hogy létrehozzanak egy 3D tervező alkalmazást, melyen ráadásul mindenki publikálhatta a saját készítésű modelljét. Nem sokkal később az Autodesk is meglátta a potenciált a webes programban, így 2013 májusában bejelentették, hogy felvásárolják a Tinkercad-t. 4 évvel később több kifutó programjukat is beintegrálták a Tinkercad-be, mely során került bele az áramkör tervezés, mikrokontrollerrel (Arduino UNO), egy lapos áramkörrel (Micro:bit) és ATtiny chippel, plusz ezek programozhatósága. (<http> 5, 2024.09.04.) A Tinkercad tartalmaz egy saját programozó fület, ahol választhatunk a hagyományos szöveg alapú programírási lehetőség és a blokk alapú programírás között, mely utóbbi nagyban megkönnyítheti a kezdő programozók dolgát, egyszerűsége és letisztultsága miatt. Mindemelett rengeteg ingyenes tananyag társul mind a

3D tervezés, mind az áramkör tervezés és programkészítéshez. A fent említetteket a 2.8. ábrán szemléltetem.



2.8. ábra: A Tinkercad fejlesztőkörnyezetének bemutatása LED villogtató kapcsolással
(Saját kép)

2.4. Érzékelők

2.4.1. Szenzorok definíciója

Az szakirodalom megkülönbözteti az érzékelőket a villamos érzékelőktől, míg előbbi meghatározása általánosabb, utóbbié konkrét. Érzékelők alatt, olyan eszközöket értünk, melyekre a mérendő fizikai, kémiai mennyiség közvetlenül hat. A villamos érzékelők ezzel szemben pedig olyan eszközök, melyek a mérendő mennyiséggel arányos villamos mennyiséget adnak. (Lambert, 2009)

2.4.2. Csoportosításuk

Több fajta szempont alapján lehet csoportosítani az érzékelőket, mint például bemeneti mennyiség, kimeneti mennyiség vagy működési alapelv szerint. Bementi érték szerint van mechanikai, optikai, villamos, termodinamikai és kémiai szenzor. Kimeneti mennyiség alapján két fajta lehet egy érzékelő villamos vagy optikai. Az érzékelőket működésüket tekintve két csoportba oszthatjuk be, vannak **passzív** és **aktív** szenzorok. Az elnevezés bár elsőre meglepő vagy nem feltétlenül triviális, de aktív érzékelők alatt azokat az érzékelőket értjük, melyek külső energiaforrás nélkül szolgáltatnak eredményt, míg passzívnak nevezzük azokat az érzékelőket, melyeknek szükségük van külső energiaellátásra (Fekete et al, 2002; Lambert, 2009).

2.4.3. Releváns érzékelők bemutatása

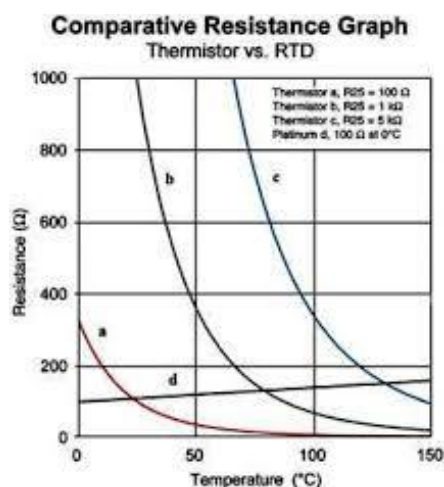
Fontos definiálnom ezen a ponton, hogy milyen paramétereket szeretnék vizsgálni, hiszen ez határozza meg a szükséges szenzorokat. Elsősorban a motor és az ahhoz szorosan kapcsolódó egységek jellemzőit szeretném vizsgálni, amik a következők:

- folyadék és levegő hőmérséklet,
- motor fordulatszám,
- önindító áramfelvétel,
- generátor töltési feszültség,
- motor kopogás.

A következőkben a fent felsorolt paraméterekhez kapcsolódó érzékelőket fogom megvizsgálni és jellemezni.

2.4.4. Hőmérséklet szenzorok

A hőmérséklet érzékelők egy részének működése ellenállás változáson alapszik. A hagyományosnak tekinthető ellenállás hőmérők fémes elemből készülnek, köztük az egyik legelterjedtebb Pt100-as hőmérséklet érzékelő is, amely platinából készül és ellenállása 0 °C értéknél 100 Ω . Egy másik ellenállás változásra alapuló, de félvezető technológiát használó szenzor típus a termisztor. A termisztorok ellenállás változása a hőmérséklet emelkedést tekintve eltér a hagyományos ellenállás hőmérőktől. míg egy ellenállás hőmérő karakterisztikája lineáris és a hőmérséklet emelkedésével az ellenállása is nő, addig egy termisztoré exponenciálisan csökken. Éppen ezért a termisztorok szűkebb hőmérséklet tartományban alkalmazhatók, mint az ellenállás hőmérők. ([http 6, 2024.09.24.](#)) A fent felsorolt szenzorok ellenállás karakterisztikája a 2.9. ábrán látható.



2.9. ábra: A termisztorok (a, b, c) és ellenállás hőmérő (d) hőmérséklet-ellenállás diagramja ([http6, 2024.09.24.](#))

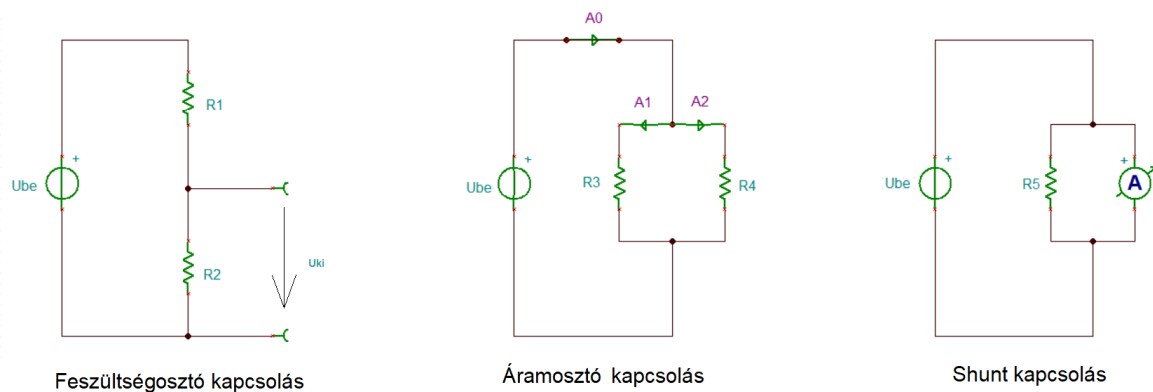
2.4.5. Fordulatszám szenzorok

A fordulatszám érzékelés esetén leginkább bevett gyakorlat a mágnesesség és a mágneses terek változásának kihasználása. Léteznek induktív érzékelők, melyek képesek a mágneses tér változásainak érzékelésére. A kialakításuk, viszonylag egyszerű, de csak olyan alkatrészek vizsgálhatók vele, melyek ferromágnesesek, azaz eredendően mágnesesek, illetve aktívan reagálnak a mágneses terekre. Éppen ez utóbbi tulajdonságuk miatt fogaskerekek környezetében szoktak ilyen típusú érzékelőket alkalmazni. Az induktív szenzorokon kívül a félvezető-alapú, úgynevezett magnetorezisztív, érzékelők terjedtek még el. Ezek a típusú érzékelők működési elve viszonylag egyszerű. A félvezető lapkán elhelyezett kötési, illetve bontási, felületeken keresztül áramot engedünk rá, mely áramnak az útjai egyenesek a kötési és bontási felület között mindaddig, amíg nem kerül az érzékelő környezetébe valamilyen mágneses tér. A mágneses tér hatására a korábban egyenes áramutak meggörbülnek, így okozva egyfajta ellenállást, minél közelebb helyezkedik el a mágneses tér az érzékelőhöz, annál jobban eltérülnek az áramutak, melynek következtében megnő az érzékelőn az ellenállás is. A fordulatszám mérése során az impulzus csúcsokat kell összeszámolnunk egységül választott idő alatt, ezek az impulzusok pedig az indukció vagy az ellenállás változásából adódnak. Ritkábban de alkalmaznak optoelektronikai érzékelőket is, melyek egy lézer és egy fotodióda segítségével érzékelik a fordulatszámot, oly módon, hogy a lézert ráirányítják a vizsgálandó alkatrészre, melyen fényelnyelő és fényvisszaverő felületek vannak kialakítva. A forgás következtében ezek az elnyelő és visszaverődő felületek váltakoznak, így kialakítva az impulzusokat, ugyanis a visszaverődő felületről a lézer fénye a fotodiódába érkezik és feszültséget indukál abban. ([http 8, 2024.09.24.](#); [http 9, 2024.09.24.](#))

2.4.6. Feszültség és áramerősség mérők

A villamos jellemzők mérésének alapja valamilyen egyéb fizikai hatás kimutatásával lehetséges. Példaképp a hagyományos árammérők működése a tekercsben folyó áram által létrejövő mágneses tér kihasználásával valósítható meg, amely egy mutatót térít ki. Ezzel szemben ma már léteznek digitális mérőeszközök, melyek sokkal pontosabb értékleolvasást tesznek lehetővé. A digitális műszerek a saját referencia értékeihez képest tudnak viszonyítani és ez alapján megmondani a mért értéket. Hasonló alapon alkalmazhatók a mikrokontrollerek is villamos mérésre. A legtöbb mikrovezérlőben most már van analóg-digitális jelátalakító, mely a beérkező feszültséget a referenciához képest összehasonlítja és típustól függően valamilyen mennyiségű bit-re osztja. Ezekhez a bitekhez pedig tudunk értéket társítani, és ez alapján a mérést számszerűsíteni. Érdemes megjegyezni, hogy a mérésekhez minden esetben kell valamilyen egyéb elektronikai kapcsolás, hogy a mérésre használni kívánt eszközt ne tegyük tönkre, illetve a működési tartományába tartsuk a mérni kívánt jellemzőket. Ilyen a feszültségosztó-, az áramosztó-, és a Shunt kapcsolás. A kapcsolásokról a 2.10. ábrán láthatunk sematikus kialakításokat.

Mikrokontrollerekkel a feszültség mérése egyszerű, hiszen tud mihez viszonyítani az eszköz. Viszont az áram mérése nem ennyire magától értendő, hiszen az áram értéke folyamatosan változik a szükséges teljesítmény függvényében, ezért van szükség a Shunt kapcsolásra, amely áram mérésnél egy alacsony ellenállással rendelkező elektronikai komponens. A kis mértékű ellenállás miatt a rajta folyó áram nem változik számottevően és az ellenálláson eső feszültség mérhető, így viszonylag elég pontosan mérhetünk akár nagyon magas áram értékeket is. Az áramosztóra pedig a gépjárművek sajátosságai miatt lehet szükség. Az önindítónak nagy mennyiségű áramra van szüksége a behúzó tekercs aktiválásához. Ez bizonyos esetekben akár 120-140 Amper is lehet. ([http 10, 2024.09.26.](#); [http 11, 2024.09.26.](#); [http 12, 2024.09.26.](#))



2.10. ábra: Feszültségosztó-, Áramosztó-, és Shunt kapcsolás egyszerű vázlata (*Saját ábra*)

2.4.7. Rezgés, kopogásérzékelők

Kopogás, csattogás a motortérből több okból is keletkezhet, melyek nem kívánatos jelenségek egy gépjármű üzemelése közben, ezért ezeket redukálni kell. Erre a célra készültek a kopogásérzékelők, melyek a motor rezgéseit vizsgálják. Az ilyen kopogásszenzorok a piezo-elektromos hatást használják ki. A piezo-elektromosság alapját a piezokristályok adják, melyek olyan különleges atomszerkezettel rendelkeznek, melyek he külső erő behatása alá kerülnek az atomjai a kristálynak úgy tudnak rendeződni, hogy a nyomott felületeken, vagy arra merőleges oldalon, potenciál különbség jön létre és így mérhető feszültség. Ennek fordítottja is működik, villamos feszültség hatása alatt ezek az anyagok képesek kitágulni vagy összehúzódni. A motorok rendellenes rezgései hatására a kopogásérzékelőben lévő kristályon alakváltozás lép fel, amelyen így feszültség keletkezik, ezt a feszültséget használhatjuk jelként, hiszen az alakváltozás mértékével arányos feszültség keletkezik a kristály megfelelő oldalain. ([http 13, 2024.09.26.](#); [http 14, 2024.09.26.](#))

2.5. Méréstechnika

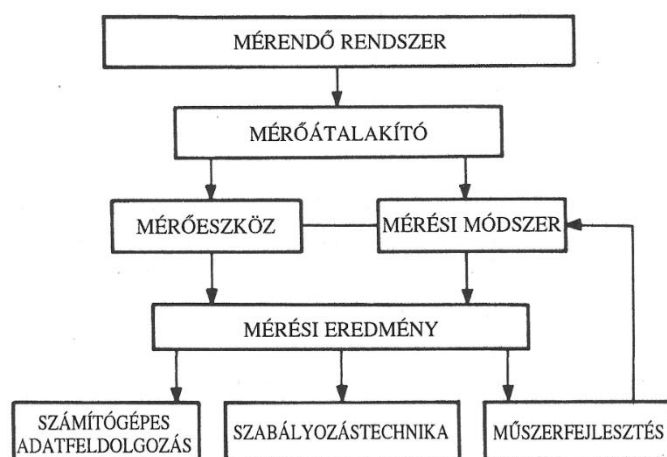
2.5.1. Mi a mérés?

Mivel az adatgyűjtés a méréstechnikán alapszik, lényeges ennek a témakörnek is a kifejtése. A **mérés** fogalma a különböző szakirodalmakban eltér némelyest, de a lényeg mindegyikben ugyanaz: olyan tervszerűen végzett tevékenységek összessége, melynek során

valamilyen fizikai, kémiai vagy egyéb mennyiség arányát, nagyságát kapjuk eredményül. Ezt szokás szerint mérőlánc (*Bouwens, 1984; Schnell és Tóth, 1967*) alkalmazásával végezzük, mely magába foglal minden olyan eszközt, berendezést, amely a mérés során valamilyen módon hozzájárul az eredményes méréshez.

2.5.2. Mérési folyamat

A mérés során a mérőlánc egyik „végén” egy szenzornak, vagy másnéven érzékelőnek, kell állnia, melynek úgy változik valamilyen villamos tulajdonsága, hogy az arányos legyen a mérendő fizikai vagy egyéb mennyiséggel (*Fekete et al, 2002; Kaposvári, 1980*). Az adatgyűjtési folyamatot az 2.11. ábrán vázolt diagram jellemzi.



2.11. ábra: Az információ szerzés folyamata
(*Fekete et al, 2002*)

Ezen kívül a mérés módszere szerint kétféle lehet, **közvetlen** vagy **közvetett**. Közvetlen mérésnek hívják a mérés technikában azt, amikor a vizsgált fizikai mennyiség közvetlenül hat a mérőberendezésre, például egy egyszerű feszültségmérés voltmérő segítségével. Közvetett mérés során pedig a vizsgált fizikai mennyiség valamilyen összefüggésben áll a mérőátalakító kimeneti jelével, például erőmérés során a mérőbéllyeg torzulása, vagy elmozdulása a villamos ellenállásának változását eredményezi, így az ellenállás képletével könnyen meghatározható az elmozdulás mértéke (*Kaposvári, 1980; Grave, 1968*).

2.5.3. A mérések bizonytalansága

Legyen szó bármilyen mérésről, mindig keletkezik egyfajta bizonytalanság a folyamat során. Ezt a bizonytalanságot nevezi a szakma a mérés hibájának, ezt a hibát igyekszünk minden esetben minimalizálni vagy bizonyos esetekben, bizonyos komponenseket zérusra redukálni. A mérés során kapott eredményt **mért értéknek** nevezik. Ez az érték csak

megközelítőleg azonos a **valódi értékkel**. Ezek alapján a mérés **abszolút hibája** a valós érték és a mérté érték különbségéből adódik. (*Schnell és Tóth, 1967; Kaposvári, 1980*)

$$H_i = x_0 - x_i \quad (1)$$

Ahol: H_i a mérés abszolút hibája, x_0 a valódi mennyiség, x_i a mért érték.

Könnyen belátható, hogy nem elhanyagolható milyen mérési tartományban keletkezik az adott mérési hiba. Példaképp nézzünk egy egyszerű feszültségmérést. Két esetet hasonlítunk össze, tegyük fel hogy az abszolút hiba mértéke mindkét esetben $\pm 0,1$ V, viszont az egyik esetben 20 V-os méréshatárra a másik esetben 2 kV-os méréshatárral dolgozunk. Ezalapján elmondható, hogy a relatív hiba, az abszolút hiba és a valódi mennyiség hányadosából határozható meg:

$$h_i = \frac{H_i}{x_0} \quad (2)$$

Ahol: h_i a mérés relatív hibája.

A mérési hiba jellegét tekintve sokrétű lehet. adódhat a mérés módszeréből, a mérés folyamatától, a környezeti hatásoktól, illetve a mérést végző személytől is. Ezeket a hibákat a szakma három fő csoportba sorolja: **durva hibák**, **rendszeres hibák** és **véletlen hibák**. A **durva hibákat** a legtöbb esetben az emberi tényező viszi a rendszerbe valamilyen figyelmetlenség vagy tévedés okán. Ilyenkor kiugró, akár irreális értékek is adódhatnak a mérés végeredményében. **Véletlen hibának** a különböző szakirodalmak azt nevezik, amikor valamilyen nem vagy nagyon nehezen meghatározható és korrigálható hibát találunk a mérés eredményei között. A véletlen hibát meghatározni és korrigálni nem lehet, így egy állandó bizonytalanságot visz a mérés eredményébe. A **rendszeres hibák** viszont ezzel szemben jól meghatározhatóak és korrigálhatóak, sőt ezeket a hibákat korrigálni is kell, hiszen jól meghatározható, azonos előjellel, rendszerességgel bekövetkező kiugró vagy eltérő értéket adnak a mérési rendszerbe és így a végeredménybe. (*Fekete et al, 2002*)

2.5.4. A mérés pontosítása

A mérés során keletkező hibák egy részét kompenzálassal lehet korrigálni, de legtöbb esetben egyszerűen csak több mérést végzünk, majd az értékek átlagával jellemezzük a mérést. A mérési bizonytalanságokat ezen túlmenően meg lehet becsülni, melynek két módszere van. Az egyik módszer alapján egy mérési sorozatot állítunk fel, több mérésből, majd az értékeket kiértékeljük statisztikai úton, a másik módszer alapja, az „egy mérés is mérés...” jelmondat.

Míg előbbi gyakran alkalmazzuk általánosságának mivolta miatt, utóbbit inkább kis elemszámú mérések során alkalmazhatjuk, illetve, ha a vizsgált mennyiséghez tartozó kialakuló hiba mértéke nem számottevő. (Kaposvári, 1980)

2.5.5. Mérési sorozatok

Mérési sorozatok készítésének 4 lehetősége van. Az első, hogy egy bizonyos mérést több alkalommal is elvégezzünk, így kellően sok mérési pont alakítható ki, melyek egymástól kellően függetlenek is lesznek. A második módszer, hogy több azonos tárgyat, vagy jellemzőt mérünk meg, ez a sorozatgyártásban nyer jelentőséget. A harmadik módszer, hasonlóan az előzőhöz a sorozat gyártásban alkalmazható, mondják mintavételnek is, kiválasztunk a sok tárgy, vagy jellemző közül n számú mintát, ezzel tesszük függetlenné a mérést. Az utolsó módszer pedig az időben lejátszódó folyamat, azonos időközökkel végezzük a mérést egy jellemzőn, tárgyon. A mérési sorozatok elemeit a mérés valószínűségi változóiként értelmezhetjük, melyből így egy eloszlás függvény írható fel. (Schnell és Tóth, 1967)

2.5.6. Mérési bizonytalanság kifejezése

A valószínűségi változók átlaga adja a lehető legpontosabb becslést a mérés várható értékére.

$$\bar{\xi} = \mu_{\xi} = \frac{\xi_1 + \xi_2 + \dots + \xi_n}{n} \quad (3)$$

Ahol: ξ a valószínűségi változók értéke, n a változók száma, μ a becslés várható értéke

A mérésre jellemző szórást egy tapasztalati összefüggés írja le. A tapasztalati szórásnégyzet és a korrigált szórás jellemzi a számunkra mekkora bizonytalansággal számolhatunk.

$$\sigma \approx \sqrt{s_n^2} = s_n = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (\xi_i - \bar{\xi})^2} \quad (4)$$

A korrigált szórásból ezután kiszámítható az átlaknak a szórása, ez lesz a standard bizonytalanság, mely bár nem kielégítő megbízhatósággal, de egy egész jó közelítéssel határozza meg a mérésünk bizonytalanságát. (Schnell és Tóth, 1967; Kaposvári, 1980)

$$u_{\bar{\xi}} = \frac{s_n}{\sqrt{n}} \approx \frac{\sigma}{\sqrt{n}} \quad (5)$$

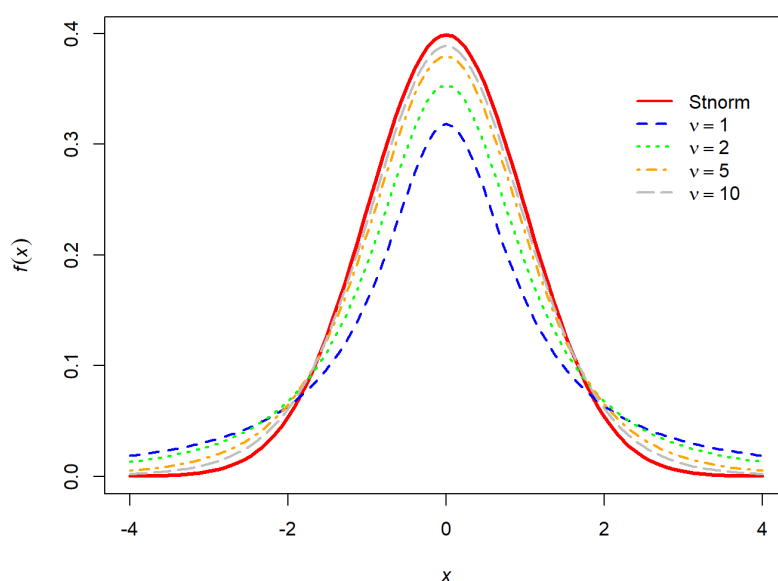
Ahol: u a standard bizonytalanság, s_n a korrigált szórás, n a mérések száma.

A bizonytalanság értékének pontosítása érdekében alkalmazhatjuk a kiterjesztett bizonytalanságot, melyet a kívánt a szinthez igazíthatjuk a standard bizonytalanság értékét. Így egy Student (t-) eloszlású függvény becslést kapunk, melyből meghatározható a kiterjesztési tényező is.

$$U = k \cdot u \quad (6)$$

Ahol: U a kiterjesztett bizonytalanság, k a kiterjesztési tényező.

A Student (t-) eloszlású függvény fő jellemzője a fokszám, mely közvetlen mérések esetén egyszerűen a mintavételek számának egyel való csökkentésével kapható, valamint közvetett méréseknél számítható az úgynevezett effektív szabadsági fok. Minél nagyobb az eloszlás fokszáma annál „karcsúbb” lesz a függvénykép, mely arra utal, hogy az egyes pontossági osztályokban egyre pontosabb lesz a bizonytalanság becslésének értéke. A 2.12. ábra bemutatja a Student (t-) eloszlás változását a fokszámok módosításával. (*http 15, 2024.09.29.*)



2.12. ábra: A Student (t-) eloszlás függvény képének változása a fokszám változtatásával (*http 15, 2024.09.29.*)

3. Konstrukció kialakítása

A korábban már meghatározott célok egyértelműsítik a feladatomat. A következőkben meghatározom az adatgyűjtő alapját, a mikrovezérlőt, és a vizsgálni kívánt jellemzőkhöz a szenzorokat a rendelkezésre álló jármű specifikációi alapján. Továbbá szemügyre veszem és beletervezem a konstrukcióba az egyéb kiegészítő elemeket. Nem végkövetelmény és nem is befolyásolja az eszköz működését sem pozitívan, sem negatívan, de szeretnék egy tároló egységet is tervezni és kialakítani a működtető elektronikának és a belső vezetékezésnek. Ezután megírom a mérés programját az Arduino saját fejlesztőkörnyezetében. Amint a konstrukció megvalósul és fizikai formát ölt, elvégzek több mérést is, lehetőség szerint más-más gép üzemállapotban, majd az általam mért értékeket kiértékelem. Egyfelől megállapítom a mérési hiba értéket és a szórást, másfelől összehasonlítom az adott jármű gépkönyvében szereplő standard értékeket a saját mérésem eredményeivel, valamennyi vizsgált jellemzőre vonatkozóan. Végezetül összegzem a tapasztalataimat, gazdasági elemzést végzek és javaslatot teszek az eszköz feljavítására, illetve további technikai és technológiai fejlesztésére, ha ennek van létjogosultsága.

3.1. Vizsgált jármű bemutatása

Mielőtt bármilyen tervezésbe is belekezdenék, tisztázom milyen járművön fogom elvégezni a méréseket. Mint az a bevezetésben is írtam, a tágabb értelemben vett családi környezetemben van egy jármű, ami a mai kor számára nem igazán modern. Ez a jármű egy Belarusz MTZ-50 típusú, kerekes, mezőgazdasági traktor. Ezt a típust és a különböző változatait az 1960-as évek elejétől több mint két évtizeden át gyártották. A mai napig közkedvelt kisebb gazdaságoknál és mezőgazdasággal foglalkozók háztájában. Egyszerűsége miatt viszonylag könnyen javítható, szerelhető és kezelhető, közkedveltsége okán pedig még mindig lehet hozzá alkatrészeket beszerezni, arányaiban egyszerűen. A 3.1. táblázat tartalmazza a jármű főbb tulajdonságait.

Motor típus	Motor névleges teljesítménye	Hengerek száma	Henger űrtartalom	Hajtóanyag típusa	Hajtóanyag ellátás
D-50	41 kW	4	4700 cm ³	Gázolaj	Dugattyús adagoló

3.1. táblázat: MTZ-50 traktor fontosabb jellemzői

A későbbi méréseim eredményét tudnom kell viszonyítani valamihez, hogy egyértelműen tudjak valamilyen konklúziót vagy konzekvenciát levonni. Emellett fontos szempont a paraméterek meghatározása a tervezés és méretezés tekintetében is. Ehhez az MTZ-50, 50L, 52, és 52L típusú traktorokhoz 1974-ben kiadott kezelési és karbantartási utasítás lesz a segítségemre. Az útmutatóból kikeresem és táblázatosan összegyűjtöm a számomra fontos és érdekes paramétereket, de mik is ezek? Nos, korábban mikor az érzékelőket bemutattam, már egyfajta meghatározást tettem, hogy milyen jellemzőket szeretnék, illetve érdemes vizsgálni. Tehát a 3.2-es táblázat tartalmazza a vizsgálatom alapját és viszonyítását képző paramétereket és jellemző értékeiket.

Hűtőfolyadék üzemi hőmérséklete [°C]	70 - 90
Motorolaj üzemi hőmérséklete [°C]	-
Beszívott levegő hőmérséklete [°C]	-
Motor (főtengely) névleges fordulatszáma [1/min]	1 700
Hátsó erőleadó tengely (TLT) névleges fordulatszáma [1/min]	883
Elektromos rendszer maximális áramfelvétele [A]	30
Generátor töltő feszültsége [V]	13,4 – 15,2

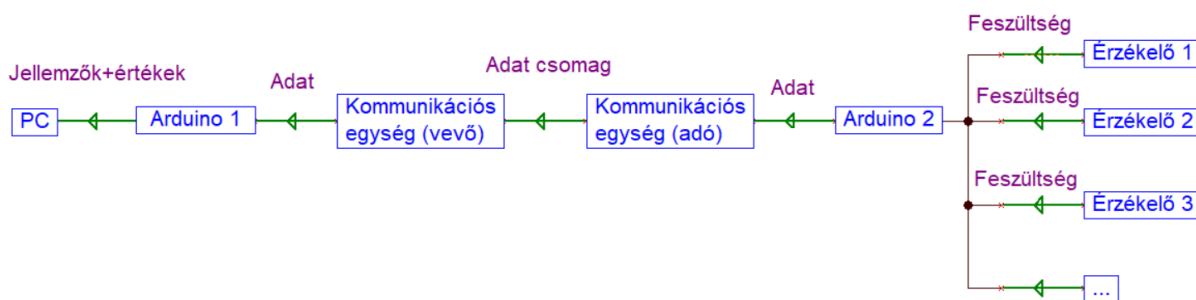
3.2. táblázat: MTZ-50 típusú traktor jellemző paraméterei

3.2. Tervezés, méretezés, programozás

Ahhoz, hogy megkezdjem a tervezést meg kell határoznom hozzávetőlegesen, hogy milyen eszközöket, érzékelőket kívánok használni a mérésre és adatgyűjtésre. Úgy vélem a mérést és adatgyűjtést érdemes két külön eszközzel végezni, mivel az javítja az egyes eszközök kihasználtságát, nagyban nem bonyolítja a feladat elvégzését és ezzel növelhető összeségében a konstrukció hatékonysága. Ez azt jelenti a gyakorlatban, hogy szükségem lesz két mikrokontrollerre, az érzékelőkre, a szenzorokhoz szükséges egyéb kiegészítő elemekre, illetve két olyan elemre, amely a két mikrovezérlő között teremti kommunikációs kapcsolatot. A 3.1-es ábrán az előbb felvázolt kapcsolásról készített egyszerű vonalas ábra mutat több információt.

3.2.1. A hardverek kiválasztása

Az előbb leírtak alapján tehát szükségem lesz két mikrokontrollerre, illetve két olyan eszközre, amely kapcsolatot teremt a mérő és adatgyűjtő között. Először is ki kell választanom a két mikrovezérlőt, hiszen a munka oroszlánrészét ezek végzik. A mérésre célszerű olyan Arduino variánst választani, melynek elegendő számban áll rendelkezésre be- és kimentik csatlakozás a szenzorok, valamint a kommunikáció miatt. Emellett kellően gyorsnak kell lennie, egyfelől a mérések miatt, hiszen vannak jellemzők melyek másodpercenként több százszor vagy ezerszer is változhatnak, mint például a fordulatszámok, másfelől mivel több jellemzőt is vizsgálni szeretnék ezért, az egyes jellemzők vizsgálatára nem szánhatok másodperceket és ezen kívül még az adatokat is továbbítani kell. Úgy szeretném kialakítani a teljes konstrukciót, hogy az tized vagy század másodpercenként szolgáltatassa az adatokat. A mérő elektronikával kapcsolatban van még egy követelmény, a memória kapacitás. A memória kapacitás azért kritikus a mérő elektronika szempontjából, mert nem csak mér, hanem adatot is továbbít, ezért arányaiban sok változót kell eltárolnia az eszköznek a programban, emellett maga a program is terjedelmesnek ígérkezik, aminek szintén el kell tudnia férni az eszközön.



3.1. ábra: A mérő és adatgyűjtő elektronika vázlatos felépítése

Az adatok fogadására és továbbítására igazán nincsenek ilyen szigorú elvárásaim. Mivel a feldolgozó elektronika feladata egyszerűbb, tehát az adatok fogadása a mérő elektronikától, illetve ezeknek az adatoknak a „lefordítása” számunkra értelmezhető, hasznos információra, ezért nincs szükség nagy memóriára, és rengeteg csatlakozó felületre. Ugyanakkor szükséges, hogy az adatfeldolgozó egység legalább akkora órajel kapacitással rendelkezzen mint a mérő rendszer, annak ellenére, hogy ha tized vagy század másodpercenként szeretnék eredményt kapni, akkor is bőven megfelel szinte bármelyik Arduino variáns.

A fent leírtak alapján választottam két Arduino kártyát, melyek a korábbi meghatározásomnak maradéktalanul megfelelnek. A méréshez egy Arduino Mega típusú

kártyát választottam, míg az adatfeldolgozáshoz egy Arduino Leonardot. A két fejlesztőkártya adatait a 3.3-as táblázatban összesítem.

		Mega 2560	Leonardo
Mikrokontroller típusa		ATmega2560	ATmega32u4
Processzor órajel [MHz]		16	16
Memória	Flash [KB]	256	32
	SRAM [KB]	8	2,5
	EEPROM [KB]	4	1
Be-, kimenetek	Analóg [db]	16 (10 bit)	12 (6+6) (10 bit)
	Digitális [db]	56 (15 PWM)	14 (6 analóg, 7 PWM)
Csatlakozók áramtűrése [mA]		20 (50 - 3.3V csatlakozón)	40 (50 – 3.3V csatlakozón)
Bemeneti feszültség [V]		6-20	6-20
Kommunikáció		UARTs, TWI, SPI	UARTs, TWI, SPI, CDC
Méret (H x Sz) [mm]		101,52 x 53,3	68,6 x 53,3

3.3. táblázat: Az Arduino Mega és Leonardo technikai adatai

Mint az látható a táblázatból kiolvastva a két Arduino változat nem sokban tér el egymástól, csak azokban a tulajdonságokban, amikben szeretném, hogy különbözzenek. Ugyanakkor érdemes megjegyezni, hogy ez az a két típus, amit a lehető költséghatékonyabban tudtam beszerezni és megfelelő paraméterekkel rendelkeznek a problémám megoldásához.

A mérő és adatfeldolgozó egység tehát megvan. A következő lépés a kommunikációs egységek meghatározása. Alapesetben két fejlesztőkártya közti kommunikáció nem annyira bonyolult, szimplán a megfelelő két-két csatlakozást (TX, RX) kell összekötni a kártyákon, ám ezzel van egy kis hátrány, az információ lassan kerül feldolgozásra és bonyolultabb tömbben küldeni az adatokat. Én inkább egy másik megoldást preferálok. A két Arduino között két CAN busz modul fog dolgozni. Az egyik modul, hasonlóan az autóiparban alkalmazottakhoz, a különböző adatokat lefordítja 8 bájt hosszúságú adatláncokra, majd ezeket továbbítja a másik modulnak, amely visszafordítja, dekódolja az adatláncot és használható információt biztosít az adatfeldolgozó egységnek. Az Arduinohoz készítenek egyben kapható CAN busz modulokat, így nem kell fáradni ennek a méretezésével. Az egyik ilyen modul, ami kereskedelmi forgalomban kapható az MCP2515-M, mely rendelkezik beépített adó-vevő rendszerrel és CAN V2.0B támogatással.

3.2.2. Szenzorok és a kiegészítő elemek méretezése

A szenzorok egy részét tekintve mondhatni nem lesz nehéz dolgom. Ezekhez a fajta járművekhez rengetegféle hagyományos, részben új, vagy teljesen új érzékelőket

forgalmaznak a mai napig. Mechanikus és elektronikus érzékelők sokasága érhető el arányaiban olcsón és nagyszámban. Ezért inkább azokra a szenzorokra helyezek nagyobb tervezési hangsúlyt, amelyeket a kereskedelmi forgalomban kaphatóktól eltérővel szeretném helyettesíteni, például költség megtakarítás végett. Ezen kívül, amennyiben szükséges, a további elektronikai elemek (ellenállások, kapcsolók, ...stb.) tervezését is elvégzem. A tervezés logikai menete a következő: kiválasztok egy érzékelőt, megállapítom, hogy milyen forrásból szerzem be, próba mérésekkel, vagy ha rendelkezésre áll az érzékelő adatlapja, meghatározom a szükséges kiegészítő elemek típusát, fajtáját és mennyiségét.

Elsőnek nem is egy hagyományos értelemben vett érzékelővel kezdek, hanem a feszültség és áramerősség mérésével. Ezeknek a jellemzőknek a vizsgálatakor két dolgot kell figyelembe vennem. Az egyik, hogy a mért feszültség ne haladja meg az 5 V-ot, mivel az Arduino-k bemenetei ezen a feszültségen működnek maximum. A generátor által leadott, közel állandó, feszültség értéke 13-14 V körüli, ezért azt egy feszültségosztó kapcsolással 2,5 V értékre kell redukálnom a biztonság miatt az esetleges feszültségcsúcsok károsító hatása ellen. A biztonságot szemelőt tartva a generátor által leadott feszültség értékét 14 V-nak tekintem és erre méretezem a feszültségosztót. A feszültség osztáshoz két ellenállásra lesz szükségem. Az egyiket megválasztom 20 k Ω értékűre. A másikat számítással tudom meghatározni. (7)(8)

$$U_{ki} = U_{be} \cdot \frac{R_2}{R_1 + R_2} \rightarrow R_2 = R_1 \cdot \frac{U_{ki}}{U_{be} - U_{ki}} \quad (7)$$

$$R_2 = 20\,000 \cdot \frac{2,5}{14 - 2,5} = 4\,347,826 \text{ } [\Omega] \quad (8)$$

A számításaim alapján egy 4 347,826 Ω értékű ellenállásra lenne szükségem, viszont ilyen értékkel nem gyártanak rétegellenállásokat. Ezért a biztonsági okokból az értéksorban megelőző, kisebb ellenállású elemet választom ki, ez 4,32 k Ω , mellyel változik a mért maximális feszültség értéke is. (9)

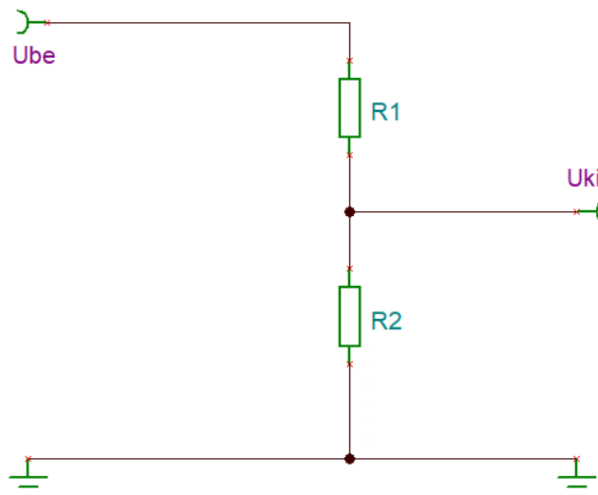
$$U_{ki} = U_{be} \cdot \frac{R_2}{R_1 + R_2} = 14 \cdot \frac{4\,320}{20\,000 + 4\,320} = 2,487 \text{ } [V] \quad (9)$$

Ezen kívül meg kell határoznom azt is, hogy mekkora teljesítménnyel kell számolni az egyes ellenállásokon, mert ha nem megfelelő teljesítmény osztályból választom az elemeket az az ellenállások sérülésével, leégésével járhat. (10)(11)

$$P_{R1} = \frac{(U_{be} - U_{ki})^2}{R_1} = \frac{(14 - 2,487)^2}{20\,000} = 6,637 \text{ } [mW] \quad (10)$$

$$P_{R2} = \frac{U_{ki}^2}{R_2} = \frac{2,487^2}{4\,320} = 1,432 \text{ [mW]} \quad (11)$$

Az eredmények azt mutatják, hogy az R1-el jelölt ellenállásomnak és az R2 jelűnek is elég a 0,4 W teljesítményosztályból választani ellenállást. A feszültségosztó kapcsolást a 3.2. ábra szemlélteti.



3.2. ábra: Feszültségmérés kapcsolási vonalas ábrája

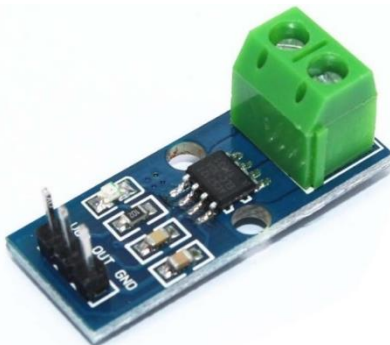
Az ábrán látható U_{be} csatlakozás a generátorról levett feszültség, míg az alatta látható testelés, valahol a jármű karosszériáján lesz kialakítva. Ennek megfelelően lesz beépítve a két ellenállás, és az U_{ki} jelzésű csatlakozás fog egyenesen a Mega kártya egyik bemenetére csatlakozni.

A másik nagyon fontos tényező, ami minden egyéb érzékelésnél is fontos, hogy a Mega ki-, bemeneteit nem terhelhetem jobban mint 20 mA, mivel a nagyobb áramterhelés tönkre teheti a vezérlőt. Ezért egy Hall-elemes árammérő modult fogok a rendszerbe építeni. A kiindulásom alapja a főáram mérésére az, hogy a jármű villamos rendszerében nem alakulhat ki 30 A áramerősségnél nagyobb érték, hiszen a rendszer védelmére szolgáló biztosíték táblában sincs ennél nagyobb áramtűrésű biztosíték, így ezt vehetem maximális mérendő mennyiségnek. Ehhez mérten az **ACS712-30A-M** típusú modult fogom használni, melynek a specifikációi alább, a 3.4. táblázatban lesznek láthatók.

Működési hőmérséklet [°C]	Mérés tartomány [A]	Érzékenység [mV/A]	Tápfeszültség [V]	Méretei [mm]
-40 - +85	± 30	66	4,5 - 5,5	31 x 13 x 11,7

3.4. táblázat: Az ACS712-30A-M típusú Hall-elemes árammérő modul főbb technikai adatai

Az ACS712 típusú Hall-elemes érzékelő modulokat sok felé alkalmazzák, inkább a hobbi elektronikával foglalkozók, de a mai napig az egyik legjobb és legköltséghatékonyabb megoldás a piacon kínált egyéb modulokhoz képest. A modul kialakítása és a tervezett szükséges bekötés a 3.3. ábrán látható.



3.3. ábra: Az ACS712 típusú Hall-elemes árammérő modul felépítése
(https://www.elektrobot.hu/items/4588_4.jpg)

Az ábrán vázolt modulnak 4 része van. Először is az egység közepén található maga a Hall-elem, tőle balra a kiegészítő és védelmi elektronika, végül a két szélén a be-, és kimenetek. A zöld csatlakozó felületek biztosítják a referencia feszültséget és a terhelte áramkör bejövő jelét. Az alsó 3 csatlakozó pedig a táp ellátásért és jelkibocsátásért felel. Az eszköz 5 V egyenfeszültségen üzemel, ehhez mérten van egy pozitív csatlakozó pont, ami az Arduino kártya 5V csatlakozópontjához köttetik, valamint van egy GND jelű csatlakozó felület, amely a testelésért felel, ez a jelölés az Arduino-n ugyanígy megtalálható. Az OUT jelű láb fogja biztosítani számomra a terhelőárammal arányos mértékű feszültséget.

A következő érzékelő a fordulatszámszenzor. Erre a célra igazából egy induktív közelítés kapcsolót fogok használni. Az általam kiválasztott típus, amit úgy ítélt meg, hogy alkalmas lehet a feladat elvégzésére, egy LJ18A3-8-Z/EX, melynek a műszaki adatait a 3.5. táblázatban összegzem.

Tápfeszültség [V DC]	Max. terhelő áram [mA]	Mintavétel sebessége [Hz]	Kapcsoló alaphelyzete	Érzékelt anyagok
6-36	300	500	NO	acél

3.5. táblázat: Az LJ18A3-8-Z/EX típusú induktív közelítés kapcsoló főbb jellemzői

A műszaki adatokon túl érdemes elmondanom, mi képezi itt a mérést. Lévén, hogy egy kapcsolóról van szó, ez a szenzor akkor bocsát ki „jelet”, ha az érzékelési távolságon belül

érkezik valamilyen fémes tárgy, lehetőleg acél. Az érzékelő hasonlóan működik mint egy hagyományos kapcsoló, két állapotot tud felvenni. Az alaphelyzete az érzékelőnek **NO** (*Normally Opened*), azaz alaphelyzetben nyitott, ez azt jelenti, hogy amíg nem kerül az érzékelési távolságon belül az alkatrész, addig a kapcsoló rész bontja az áramkört, és amint az érzékelő útjába kerül az alkatrész, az indukció hatására zárja az áramkört, így szabadon átfolyik rajta az áram. Ezeket a kapcsolásokat fogom a mikrokontrollerrel vizsgálni, majd a megfelelő matematikai átalakításokkal kinyerhető lesz a fordulatszám értéke. Továbbá fontos kiemelni, hogy a két indukciós kapcsoló 12 V-os egyenfeszültségen fog üzemelni, míg az Arduino bemeneti csatlakozásai maximum 5 V feszültségen működnek. Az érzékelőben lévő tekercsen eső feszültség zárt állapotban közelítőleg 3 V, így a ténylegesen jelként szolgáló feszültség értéke 9 V. Ez túl magas érték ahhoz, hogy közvetlenül a Mega bemenetére engedjem, ezért egy felező feszültségosztáson keresztül juttatok jelet a mikrovezérlőnek, ami így már csak 4,5 V értékű feszültség lesz. A feszültségosztáshoz érzékelőnként két darab 10 kΩ értékű réteg ellenállást használok. A nagyértékű ellenállásra azért van szükség, hogy a megengedett áram értékét se lépjem túl, amivel tönkre tehetnem a mikrovezérlőt. (12)

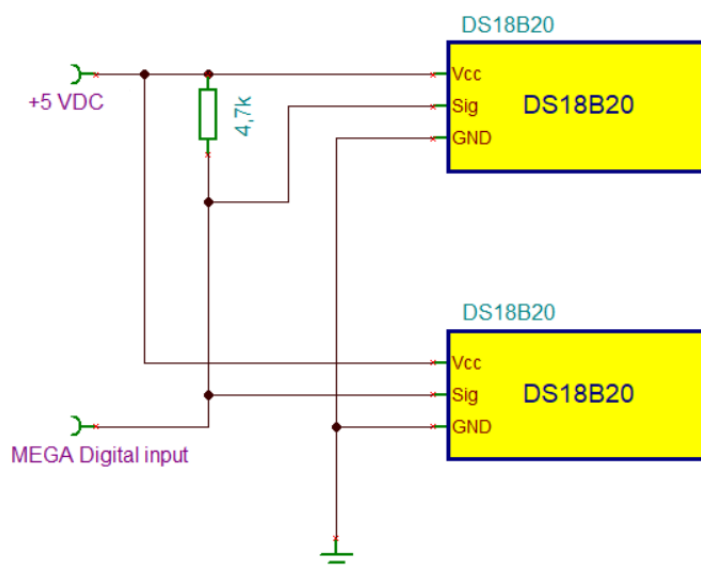
$$I = \frac{U}{R} = \frac{4,5}{10\,000} = 0,00045 [A] = 0,45 [mA] \quad (12)$$

A hűtőfolyadék hőmérsékletét és az olajteknőben lévő olaj hőmérsékletének a méréséhez ugyanolyan érzékelőket használok. A technikai adottságai miatt két darab DS18B20 típusú digitális hőmérő szondát alkalmazok majd, melyeknek az Arduino rendszerhez van fejlesztve külön könyvtár, így gyorsan és egyszerűen tudom majd az adatokat kinyerni az érzékelőkből. A legtöbb hagyományos ellenállás hőmérővel szemben ez a szonda magában kiértékeli a hőmérséklet érzékelőből érkező jelet és azt bit szöveggé alakítja. Ezzel nekem is könnyítve a dolgomon, hiszen nem kell külön elektronikai elemeket közbe építenem a hőmérő ellenállás változásának mérésére. Ezen felül a hőszonda további pozitívuma, hogy eleve vízhatlanítva van, egy rozsdamentes acélsőben kapott helyet a hőszonda elektronikája. Az érzékelő fontosabb tulajdonságai a 3.6. táblázatban lesznek láthatóak.

Tápfeszültség [V]	Hőmérsékletmérési tartomány [°C]	Mérési pontosság [°C] (-10 - +85 °C között)	Adatszélesség [bit]
3,3 – 5	-55 – +125	±0,5	9-12

3.6. táblázat: A DS18B20 digitális hőszonda főbb technikai paraméterei

Ennek az érzékelőnek két nagyon fontos tulajdonságát kell kiemelnem. Az első tulajdonság igazán az érzékelőhöz tartozó plusz elemeket érinti, ugyanis egy ilyen típusú hőmérséklet szenzorhoz be kell építeni egy úgynevezett felhúzó ellenállást párhuzamosan az érzékelő táp vezetéke és jel vezetéke közé. Erre azért van szükség, mert ez a hőszonda beépített kiértékelő elektronikával van szerelve, ami így egy kódsorként küldi az adatokat a mikrovezérlőnek, viszont ez az elektronika önmagában nem képes a „magas” állapotú jeleket jól továbbítani. Ezért a felhúzó ellenállás segíti a szenzort abban, hogy ezek a „magas” szintű jelek is a megfelelő feszültségszintet adják tovább a mikrovezérlőnek. A másik nem elhanyagolható tulajdonsága ezeknek a hőszondáknak, hogy a beépített kiértékelő elektronika miatt, akár több ilyen érzékelőt is köthetünk ugyanarra a mikrovezérlő bemenetre, mert a különböző szondáknak különböző széria kódjuk van, így egy programozási eljárással kinyerhető, hogy hány szenzort alkalmazunk és a különböző érzékelők milyen hőmérsékletet érzékelnek. Ez utóbbi tulajdonság két szempontból is megkönnyíti a dolgom, hiszen amellet,



3.4. ábra: Az olaj- és vízhőmérés bekötési vázlata

hogy kevesebb vezetéket és bemeneti csatlakozást tudok használni még a felhúzó ellenállások számát is lecsökkenthetem egyre, akármennyi érzékelőt is használnék. A bekötés vázlata a 3.4. ábrán látható.

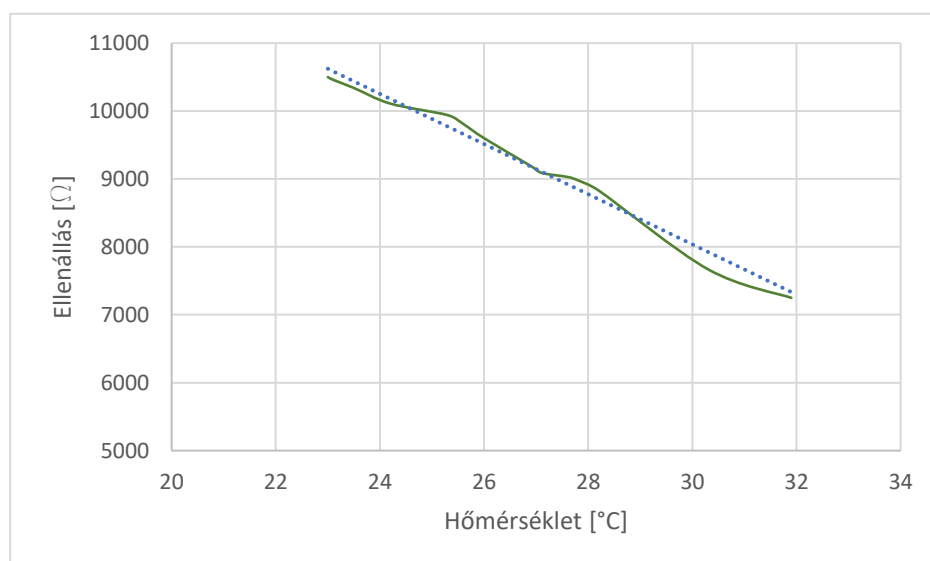
Következőnek az égéstérbe kerülő levegő, azaz a beszívott levegő hőmérsékletének mérésére szolgáló szenzort jellemezném. Erre a célra egy másik járműből származó hőmérőt szeretnék alkalmazni, mely eredeti felhasználását tekintve a klíma által szolgáltatott levegő hőmérsékletét mérte. A szenzorról azt hitték hibás, de mint később kiderült, nem azzal volt a

gond. Emellett mivel az erő- és munkagép technikában nem terjedt el a szokás, hogy adatlapokat adjanak ki az érzékelőkhöz, ezért én sem találta. hiába kerestem. Épp ezért méréssel tudom majd megállapítani az érzékelő tulajdonságait, illetve ezekkel a mérésekkel bizonyíthatom az érzékelő megfelelőségét is. Az érzékelő hő koefficiens karakterisztikájának meghatározásához szükséges valamiféle referenciát kialakítanom, aminek a dolga az lesz, hogy a lehető legpontosabb hőmérséklet értéket mutassa. A referenciámat egy Arduino-s méréssel határozom meg, ehhez pedig egy ASAIR DHT 11 típusú hőmérséklet és pára szenzort használok. A szenzor specifikációi a 3.7. táblázat tartalmazza.

Tápforrás [V]	Méréshatár (pára) [%]	Méréshatár (hő) [°C]	Mérési pontosság (pára) [%]	Mérési pontosság (hő) [°C]
3,3 – 5,5	5 - 95	-20 - +60	±5	±2

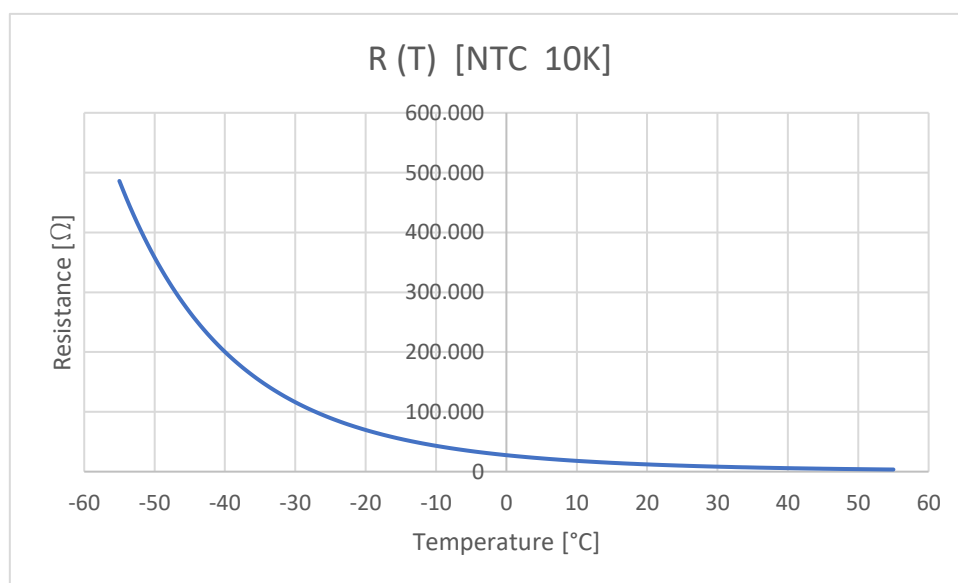
3.7. táblázat: Az ASAIR DHT11 hő- és páramérő modul technikai adatai

Elmondható, hogy a mérési pontossága ennek az érzékelőnek nem kimagaslóan jó, ugyanakkor megközelítőleg elég jó értéke és támpontot ad a léghőmérő tulajdonságának méréséhez. Több különböző, állandósultnak tekintett állapotban megmértem a léghőmérő ellenállását és leolvasom a DHT szenzor által mért hőmérsékletet, ezeket feljegyzem egy Excel táblában és diagramot készítek belőle, amelyre egy megfelelő típusú trendvonalat helyezve, meghatározható a léghőmérő hőfokfüggése. Az elkészült diagram a 3.5. ábrán látható.



3.5. ábra: A Magneti Marelli A 860 002.0.0 típusú ellenállás hőmérő próbamérési diagramja

Habár, a referenciául szolgáló hőmérő teljes mérési tartományának egy viszonylag kis méretű szakaszán vizsgáltam a léghőmérőmet, három fontos információ olvasható le a próbamérésem diagramjáról. Az első, ami talán a legnyilvánvalóbb, hogy a mérésem koránt sem pontos, viszont a második, hogy a diagramból egyértelműen látszik, hogy egy NTC (Negative Temperature Coefficient), azaz negatív hőfoktényezőjű ellenállás hőmérőről van szó, tehát a hőmérséklet növekedésével a hőmérő ellenállása csökken. A harmadik pedig, hogy a 25 °C-hoz tartozó ellenállás értéke hozzávetőlegesen 10 kΩ. Az ilyen típusú hőszensoroknak bár első ránézésre lineáris a karakterisztikája, ez csak logaritmikus osztás esetén mondható igaznak. A 3.6. ábrán egy tipikus NTC 10K termisztor karakterisztikát láthatunk.



3.6. ábra: Az NTC 10K típusú termisztor hőmérséklet-ellenállás karakterisztikája

Jó közelítéssel a 0-55 °C hőmérséklet intervallumon a függvény lefutása lineáris. Ez szerencsés eset, hiszen ritkán fordul elő, főként a hidegebb évszakokban, nullától kisebb értékű környezeti hőmérséklet. Éppen ezért erre a tartományra elég jó pontossággal tudom alkalmazni a β kiszámításához felírható képlet átrendezett formáját a beszívott levegő hőmérsékletének meghatározására. (13)

$$\beta = \frac{\ln\left(\frac{R_0}{R_1}\right)}{\left(\frac{1}{T_1} - \frac{1}{T_0}\right)} \rightarrow T_1 = \frac{1}{\frac{1}{T_0} + \frac{1}{\beta} \cdot \ln\left(\frac{R_1}{R_0}\right)} \quad (13)$$

A β értéke mutatja meg egy bizonyos tartomány hőfokfüggését, melyet érdemes olyan szakaszán felvenni a függvénynek, amely közel lineáris. Érzékelőnként változhat ez az érték, de átlagosan 3 400 K körül adódik. Esetemben a forrásként szolgáló táblázat alapján a β

pontos értéke 3 380 K, ezt használom fel a kiértékelő program számításaihoz. Az R_0 és T_0 értékek a referencia hőmérséklethez tartozó adatok, melyek esetemben: $R_0 = 10\,000\ [\Omega]$, $T_0 = 25\ [^\circ\text{C}] = 298,15\ [\text{K}]$.

További szempont a léghőmérésnél, hogy amint azt korábban is említettem az érzékelő a hőmérséklet változásával változtatja az ellenállását, amit közvetlenül nem tudok mérni. Ugyanakkor egy egyszerű feszültségosztó kapcsolással viszont meg tudom határozni az ellenállás változásával járó feszültség változást. Az érzékelőt 12 V tápfeszültséggel látom el, tehát a feszültségosztáshoz használni kívánt ellenállást úgy kell megválasztanom, hogy az egy választott maximális hőmérsékleti szinten (minimális ellenálláson) adjon le az 4,5 V feszültséget. Az előbb elmondottakkal ellentétben inkább fordítva közelítem meg a probléma megoldását és választok egy ellenállást majd kiszámolom mi az az ellenállás érték és ezzel az a hőmérséklet szint, amin maximálisan 4,5 V feszültség esik a választott ellenálláson. (14) Fontos még megjegyezni, hogy a hőmérséklet számítás során a hőmennyiség értékek K (kelvin) mértékegységben használatosak. (15) A választott ellenállás értéke 3 k Ω .

$$R_1 = \frac{R_2 \cdot U_{be}}{U_{ki}} - R_2 = \frac{3\,000 \cdot 12}{4,5} - 3\,000 = 5\,000[\Omega] \quad (14)$$

$$T_1 = \frac{1}{\frac{1}{T_0} + \frac{1}{\beta} \cdot \ln\left(\frac{R_1}{R_0}\right)} - 273,15 \quad (15)$$

$$T_1 = \frac{1}{\frac{1}{298,15} + \frac{1}{3380} \cdot \ln\left(\frac{5\,000}{10\,000}\right)} - 273,15 = 44,417\ [^\circ\text{C}]$$

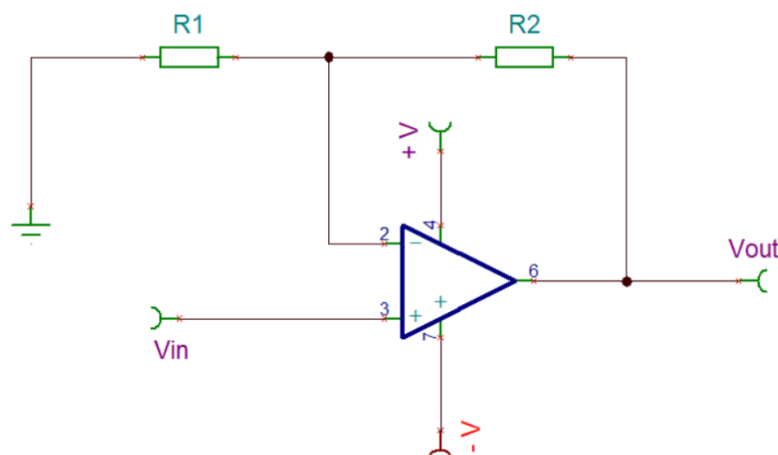
Tehát a maximális mérhető léghőmérséklet valamivel 44,5 $^\circ\text{C}$ alatt adódik. A legalacsonyabb hőmérséklet meghatározása szükségtelen, hiszen elég nagy részletességgel lehet majd meghatározni a hőmérsékletek értékét. Viszont egy nagyon fontos negatívumát a termisztoroknak ki kell küszöbölnöm, mégpedig az önfűtést. Az önfűtés jelensége azért alakul ki, mert a változó ellenállás mellett állandó marad a tápfeszültség értéke. Az önfűtés hatására kialakuló hőmérséklet torzulás számítással elég pontosan korrigálható. Az általam talált és alapul vett termisztor hőmérséklet disszipációs együtthatója $\delta = 0,5\ [^\circ\text{C}/\text{mW}]$. Ezek alapján kiszámítható a disszipációból eredő hőmérsékleti érték torzulása. (16)(17)

$$P_T = \frac{U_{be}^2 \cdot 1000}{R_1} [mW] \quad (16)$$

$$T_D = \delta \cdot P_t \quad (17)$$

Tehát a kiszámított TD értéket minden esetben le kell vonnom az ellenállás mérés során meghatározott hőmérséklet értékéből. Ezzel pontosítható a mérés, de ez sem vezet teljes megoldáshoz, hiszen ezzel csak azt tudjuk meghatározni, hogy egy adott ellenállás szinten mekkora a disszipációs hő viszont maga az önfűtés hatása már korábban el kezd hatni, ahol egy másik értékű fűtő hatás érvényes. Ugyanakkor így már kellőképp javítható a mérésem pontossága.

Végezetül a kopogás szenzorról szeretnék néhány szót ejteni. A kopogás szenzorral is hasonló a helyzet mint a beszívott levegő hőmérsékletérzékelővel. Ezt is egy másik jármű motorjáról szerelték le, melynek egyéb mechanikus problémája volt, viszont az érzékelői kifogástalan állapotban vannak. Viszont az előző esethez képest egy kissé egyszerűbb a dolgom, ugyanis az interneten találtam a kapott érzékelőhöz technikai információkat. A kapott érzékelő egy Bosch KS-1-P típusú piezo-kerámiás kopogás szenzor. Az adatlapja alapján ez a fajta kopogás szenzor 1-20 kHz frekvenciatartományban működik és az érzékenysége 5 kHz frekvencián $26 \pm 8 \text{ mV/g}$, ahol a „g” a Föld gyorsulási állandóját szimbolizálja ($g = 9,81 \text{ m/s}^2$). Ezek alapján kijelenthető, hogy egy igen kis mértékű feszültséget kellene mérjek, ezért alkalmazok egy műveleti erősítőt, hogy a mérendő feszültség arányosan magasabb legyen, így fokozva a mérés hatékonyságát. A műveleti erősítő önmagában nem elég, kellenek kiegészítő elemek, pontosan fogalmazva ellenállások, hogy egy olyan kapcsolást hozzak létre, mely a kopogás érzékelő által kibocsátott jelet ötszörösére erősíti. Az erősítésre egy úgynevezett neminvertáló kapcsolást alkalmazok, hogy a szenzor kimenő feszültségének negációja ne történjen meg, azaz ne fordítson előjelet az erősítés során. Az alapkapcsolás vázlatát a 3.7. ábra tartalmazza.



3.7. ábra: Nem-invertáló erősítő alap kapcsolás vonal diagramja

Jól látszik hogy a műveleti erősítőn kívül két darab ellenállásra lesz szükségem a teljes erősítő kapcsoláshoz. Ilyen bekötés esetén az erősítés mértékét csak és kizárólag a két ellenállás aránya határozza meg. Ebben az esetben is az áram értékét alacsonyan tartásának érdekében viszonylag magas értékű ellenállásokat kell használnjak. Azt a kritériumot támasztottam az erősítő kapcsolással szemben, hogy az a beérkező feszültséget ötszörösére növelje. Ahhoz hogy ezt elérjem ki kell számítanom a két ellenállás értékét a már korábban említett képlet segítségével, de ehhez előbb választok tetszőlegesen egy ellenállást, ez legyen 20 kΩ. (18)

$$A = 1 + \frac{R_1}{R_1} \rightarrow R_2 = (A - 1) \cdot R_1 = (5 - 1) \cdot 20\,000 = 80\,000 [\Omega] \quad (18)$$

A számítások alapján egy olyan ellenállásra lenne szükségem, melyet nem tudok beszerezni. Ugyanakkor, ha a 20,5 kΩ-os ellenállást választom ahhoz párba tudom állítani a 82 kΩ-os ellenállást és megmarad az erősítés értéke. Ezen kívül gondoskodnom kell a műveleti erősítő tápellátásáról, mellyel ezzel együtt meg adom az erősítő kimeneti jelének maximumát, hiszen a tápforrástól nem képes sem nagyobb se kisebb értéket kibocsátani magából a műveleti erősítő. Annak érdekében, hogy az Arduino-t se tegyem tönkre a tápfeszültséget 5 V értékűnek választom.

Az érzékelők és a CAN kommunikációs modulok mellett még egy eszközt alkalmazok, még hozzá egyfajta tápforrást. A tápforrással szemben támasztott elvárásaim a következők:

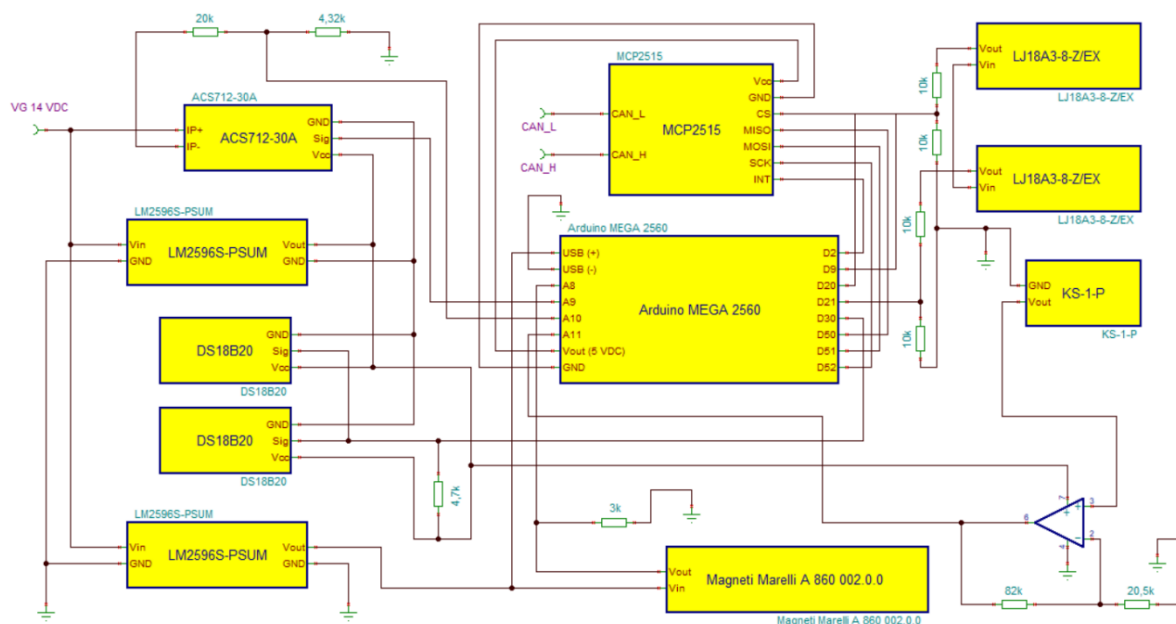
- A kimeneti feszültséghez képest magas bemeneti feszültség tűrés,
- kis méret,
- gazdaságos kialakítás,
- diszkrét és stabil feszültség szint tartás.

A fentiek alapján egy igazán egyszerű, rendkívül gazdaságos, kis méretű tápforrást sikerült találnom, melynek igen magas a bementi feszültségtűrése és amellet, hogy alap feszültség szintek állíthatóak rajta (3,3V, 5V, 12V), még egészen kis lépésközű „fokozatmentes” feszültség beállításra is lehetőség van. A választott tápellátó modul egy LM2596S-PSUM típusú, úgynevezett Step-down feszültség forrás, melynek a főbb technikai adottságairól a 3.8. táblázat nyújt átfogó információt.

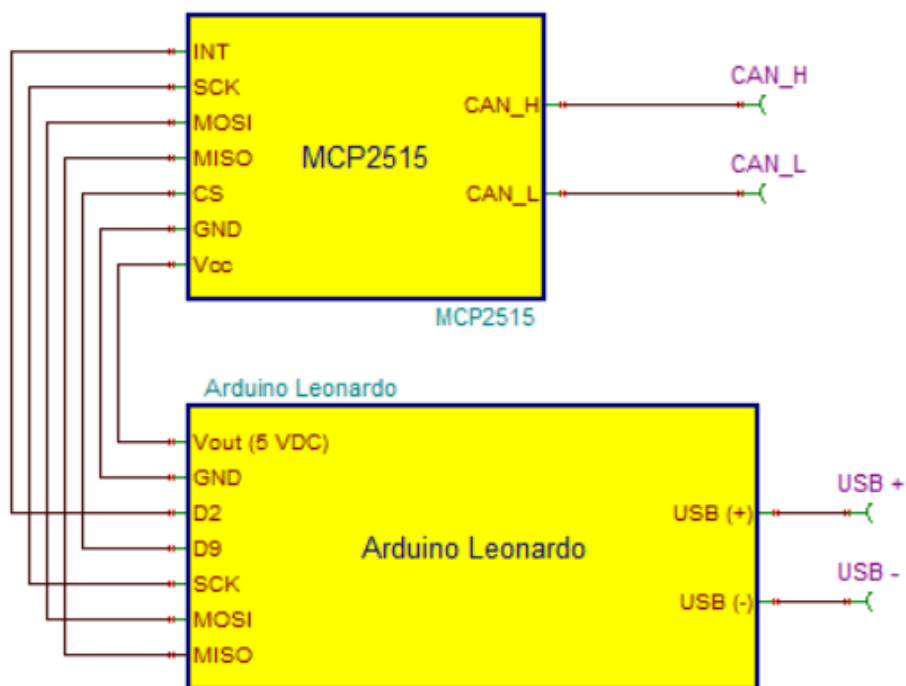
Bemeneti feszültség [V]	Kimeneti feszültség (állítható) [V]	Áramtűrés [A]	Kapcsolási frekvencia [kHz]
4,5-35	1,5-35	3	120

3.8. táblázat: Az LM2596S-PSUM tápegység modul főbb villamos jellemzői

Az ilyen típusú feszültség források működése a mágneses energia tárolásán és kiengedésén alapszik. A modul egy tekercsen tárolja ezt az energiát, melyet a kapcsoló szerkezet aktiválásakor tölt fel a rendszer és amint feltöltődött a kapcsoló ráengedi a kimenetre a feszültséget. Amint az elektronika érzékeli egy visszajelző szálon keresztül, hogy csökken a tárolt mágneses energia és ezzel együtt a kimeneti feszültség, újra feltölti a tekercset. Ezzel egy igen stabil és viszonylag állandó szintű tápforrást lehet létrehozni. Ez is volt az egyik oka annak, hogy egy ilyen típusú, stabil tápot találja, hiszen a jármű villamos berendezéseinek kapcsolgatása feszültség csúcsokat és völgyeket eredményezhet, ami instabilitást visz a rendszerembe. tehát szükséges az egyenletes tápforrás a mérő- és értékelő elektronikának. A fent említett tápmodul egyik hátránya, hogy egy bementi forrásról csak egy kimeneti forrást képes biztosítani, így ilyen modulból 2 darabot használok majd, egyet 5V előállítására, mely az érzékelőim egy részét látja el tápfeszültséggel. A másik modul a többi érzékelő és az Arduino Mega mikrokontroller tápforrásáért felel majd 12V-os feszültség értékkel. Az érzékelők egy csökkentett méretű próbapanelen lesznek ellátva feszültséggel, illetve ezen a próbapanelen kap helyet minden egyéb villamos kapcsolás is. A teljes elektronika vázlatos vonalrajza a 3.8. és 3.9. ábrán lesz látható.



3.8. ábra: A mérő elektronika bekötési vázlata



3.9. ábra: A kiértékelő elektronika bekötési vázlata

3.2.3. Az elektronika tokozása

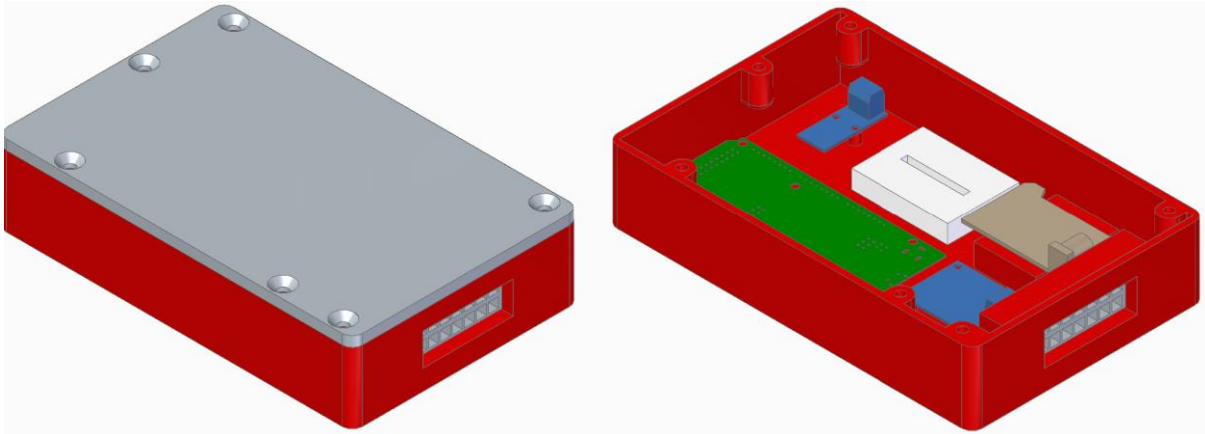
Amint azt korábban már említettem, célom nem csak úgy a levegőben lógva hagyni az elektronikát, már csak az üzemelést szem előtt tartva sem praktikus, de úgy vélem a teljeskörű kidolgozottság érdekében ez is szükséges. A tároló egységekkel támasztottam néhány alapszabályt, amit a tervezés során igyekeztem maximálisan követni, ezek a kritériumok a következők:

- kompakt és egyszerű felépítés,
- könnyű össze- és szét szerelhetőség,
- független kialakítás,
- kis súly.

A fent felsoroltakat követve arra jutottam, hogy a legegyszerűbb az lenne, ha az elektronikát tartalmazó tokok 3D nyomtatási technológiával készülnének. Egy ismerősöm vállalta is, hogy legyártja nekem őket, amennyiben megtervezem.

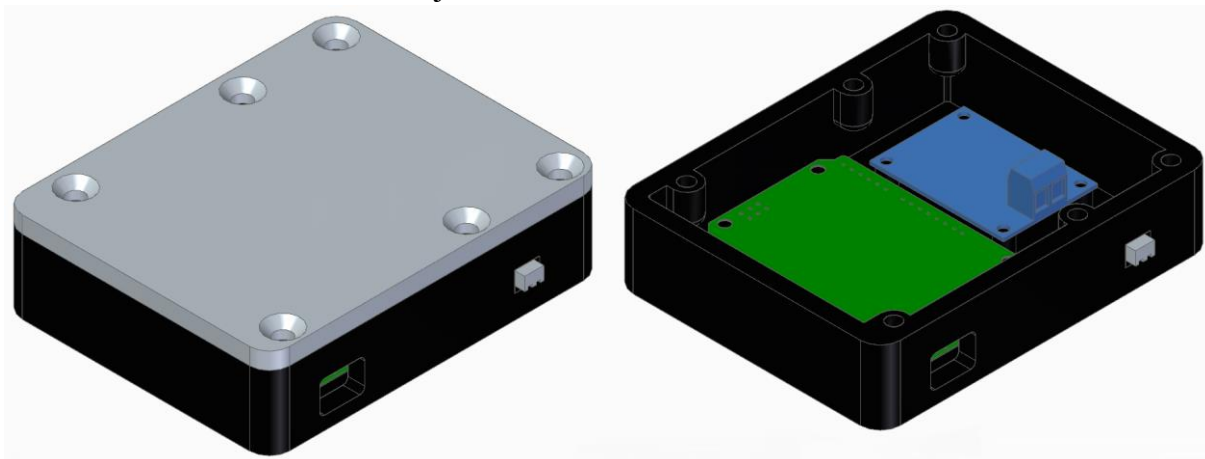
Tehát, szükségem volt a mikrovezérlők, CAN modulok, az árammérő és tápmodul, valamint az egyéb eszközök, mint a csatlakozók és a próbapanel modelljeire. Ezek közül a két mikrokontroller alap modelljét megtaláltam a Traceparts nevű, ingyenes 3D modelleket

tartalmazó oldalon, ezzel tudtam gyorsítani a tervezés folyamatát, viszont a többi elemet saját magamnak kellett modelleznem. A tervezést a Solid Edge nevű 3D tervezőszoftver 2021-es diák verziójának segítségével készítettem el.



3.10. ábra: A mérő elektronika tároló tokozásának háromdimenziós modellje

Minden szükséges modellt csak annyira részletesen terveztem meg, amennyire azt a feladat megkövetelte, így tehát a modulok és kiegészítő elemek fontos dimenziója van kiemelve és modellezve részletesen, ez vonatkozik a kiterjedésre (szélesség, hosszúság, magasság), alaki sajátosságokra és a felfogatási lehetőségekre. A tokozások függetlenségét egymástól a csatlakozók teszik majd lehetővé, mely célra három megoldás született. Az első megoldás a mérő rendszerre vonatkozik, ezen a tokozáson egy csatlakozó felületet alakítok ki egy régi számítógép táp és az ellendarabjául szolgáló alaplapi csatlakozóval. A kiértékelő elektronikán két felület lesz, egy ami közvetlen hozzáférést biztosít a Leonardo kártyához, és egy külön csatlakozás a CAN modul adatkábeleinek. Tehát a két elektronika, két külön tokozásba kerül, ezekről a 3.10. és a 3.11. ábrák nyújtanak betekintést. A tokozások pontos méreteit a mellékletek közé tett műszaki rajzok tartalmazzák.

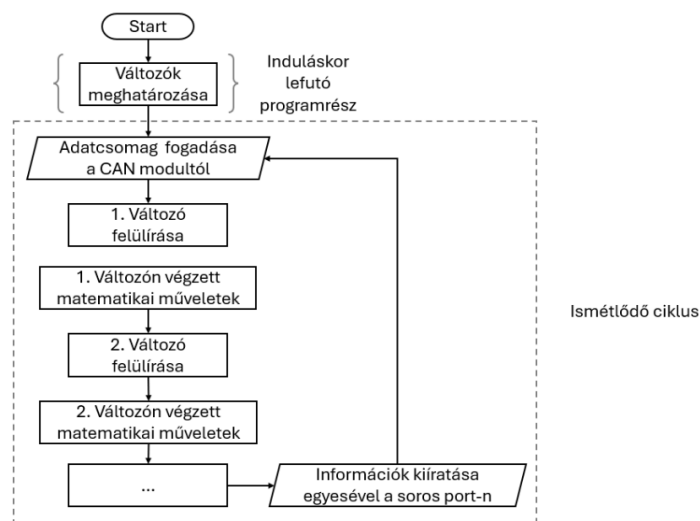


3.11. ábra: A kiértékelő elektronika tároló tokozásának háromdimenziós modellje

3.2.4. A mérés és kiértékelés programterve

Mivel a mérést és az mérésből származó adatok feldolgozását két külön mikrokontrolleren végzem majd, szükségem lesz két programra is. Az egyik végzi majd a mérést, még hozzá oly módon, hogy az diagnosztizáló eszköz bekapcsolása után várakoztatom a rendszert, illetve egy önellenőrző programrésszel vizsgálom, hogy minden érzékelő rendben van-e és tesztelem a kommunikációs egységet is, majd ezt követően indítja el a program a mérést. A mérés folyamatosan ismétlődik egy előre meghatározott periódus szerint, ez biztosítja majd a folyamatos információáramlást. Ezt az adat halmazt küldi majd tovább a CAN egységnek a mérőelektronika minden ciklus végén. A CAN egységek átadják egymásnak az adat köteget és így kerül az információ halmaz a kiértékelő elektronikához. Itt jön a képbe a második program. Ez végzi majd az adatok fogadását, rendszerezését és továbbítását a számítógépre, de már sokkal inkább rendezettebb formába, indexálva.

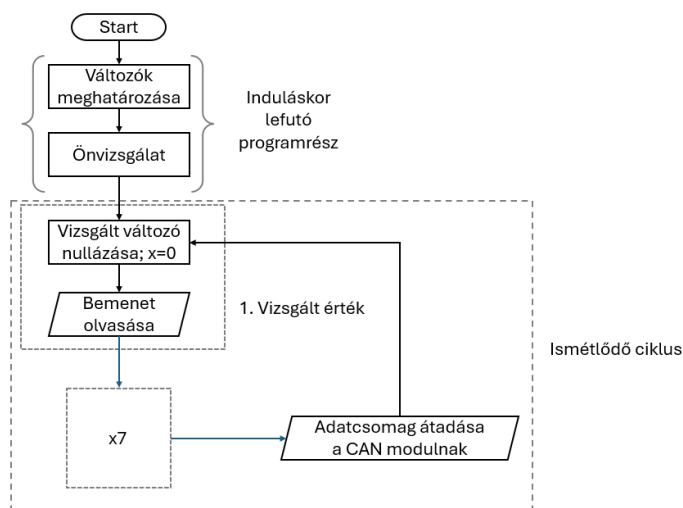
A programírás előtt érdemes minden esetben egy folyamat ábrával szemléltetni, és sorra venni a program feladatát. Ez a módszer megkönnyítheti a későbbi dolgomat, illetve a laikusok, és érdeklődők számára is könnyebb betekintést tesz lehetővé a program működésébe. A fent említett két programhoz külön-külön folyamatábrát készítek, ezek a 3.12. ábrán és a 3.13. ábrán lesznek láthatóak.



3.12. ábra: Az adatfeldolgozáshoz tartozó program folyamatábrája (vázlat)

Mint ahogyan azt az előbb is leírtam, ezek az ábrák iránymutatóak és igazán csak a programlefutás menetének megértésére szolgál. Azért készítettem csak vázlatos folyamatábrákat, mert ezek jó iránymutatásként szolgálnak a dolgozatom további részében.

Ugyanakkor fontosnak tartom ezen a ponton kiemelni, hogy a végleges programok nem lesznek ennyire egyszerűek és triviálisak, hiszen például a különböző paraméterek vizsgálata vagy mérése eltérnek egymástól, viszont az alapja mindnek ugyan az, ennek mentén folytatom a tervezést.



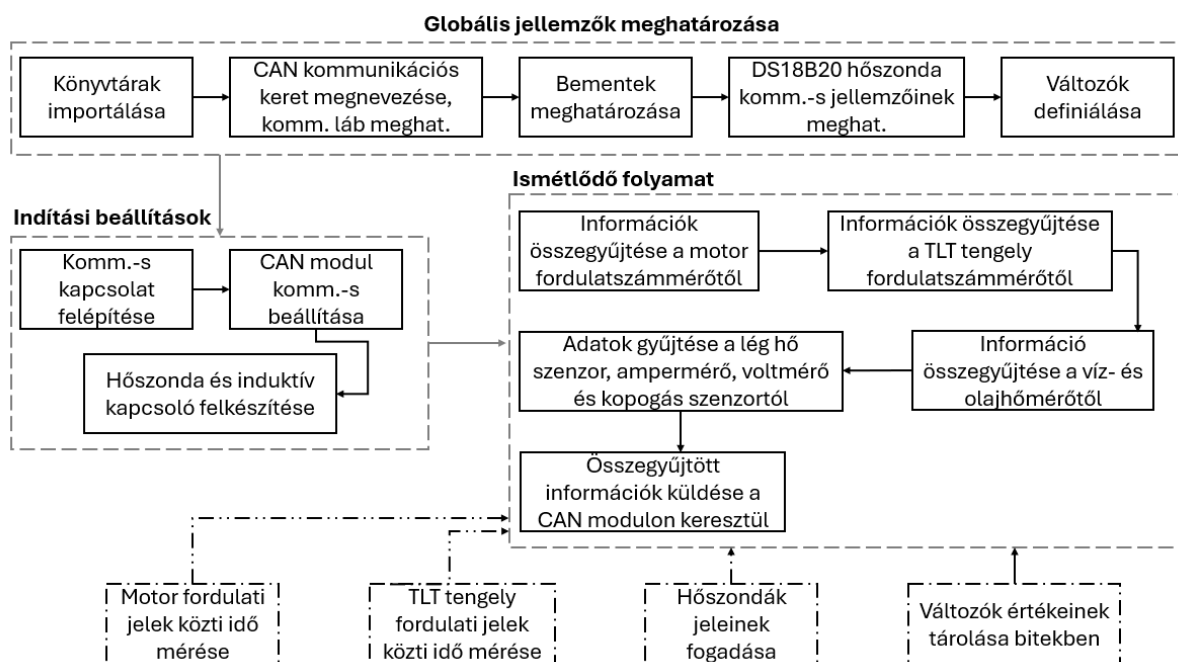
3.13. ábra: A mérő program folyamatábrája (vázlat)

A programtervezésnek további két fontos eleme van. Az egyik abból fakad, hogy az Arduino és a hozzá kapcsolódó különböző elemek (érzékelők, modulok, kijelzők, kommunikációs egységek) külső forrásból is származhatnak, tehát nem fontos az Arduino csoport által kínált megoldásokat használni, hanem a fejlesztői kártyákhoz úgynevezett harmadik féltől származó (angolul *third-party*) eszközök is csatlakoztathatóak. Viszont ezek többsége nem üzemel csak úgy magától, ezért különféle könyvtárakat kell letölteni és hozzáadni az Arduino fejlesztőkörnyezethez, amelyek tartalmazzák egy adott érzékelő, modul vagy egyéb külső egység működéséhez szükséges gépkódjainak hívójelét. Ezek a hívójeltek alkalmazhatóak a fejlesztőkörnyezetben, így akár egy hívójellel egy komplexebb kódhalmaz megírását is meg lehet spórolni, ezzel időt nyerve és könnyítve a felhasználást. A dolgozat szempontjából ez lényeges, hiszen alkalmazok olyan eszközöket, melyeknek szükségük van a saját könyvtárukra a működéshez, ilyen például a CAN modul is.

A másik kiemelendő dolog, hogy a fejlesztői környezet tartalmaz alapkapcsolásokhoz tartozó komplett kódsorokat. Ezek a fent említett könyvtárakkal bővíthetőek. A legtöbb könyvtár tartalmaz alap programokat is például teszteléshez, melyek a kezdő lépéseket segítik elő, nagyon gyakran részletes megjegyzéseket is írnak a kódsorba, mely az értelmezést és a könyvtár tartalmának könnyű megtanulását segíti elő, továbbá a szintaktikai formát is megadja.

A saját programom tervezése kapcsán mindkét segítséget aktívan használok majd, hiszen az utóbbit mindenképpen alkalmaznom kell, hogy működtetni tudjam a különböző érzékelőimet és a kommunikációs egységet, az előbbi pedig a program megírásának gyorsítására szolgál, hogy ne kelljen újra feltalálnom a már létező és szükséges program sorokat. Ez természetesen nem jelenti azt, hogy nem módosítom, vagy nyúlok bele paraméterek értékeibe, csupán azt, hogy alapként, egyfajta mankóként használok az előre megírt segédprogramokat, és úgy formálom, hogy a feladat elvégzéséhez megfelelő kódhalmazt kapjak végeredményül.

A dolgozat során nem szándékozom mély programírás elméletekbe, illetve technikai részletekbe merülni, a használt kódsorok a mellékletek között megtalálható lesz. Inkább egyfajta működés leírást, bemutatást szeretnék felvázolni, hogy a lehető legegyszerűbb és legközhértebb legyen. A megírt program szintaktikai és terminológiai jellemzői ez esetben másodlagosak, igazán a lejátszódó folyamatok összessége több információt hordoz magában. Éppen ezért újabb szemléltető ábrákkal szeretném leírni a tervezett programok felépítését. A 3. 14. ábra a mérő program pontos folyamatát, lefutását követi végig.

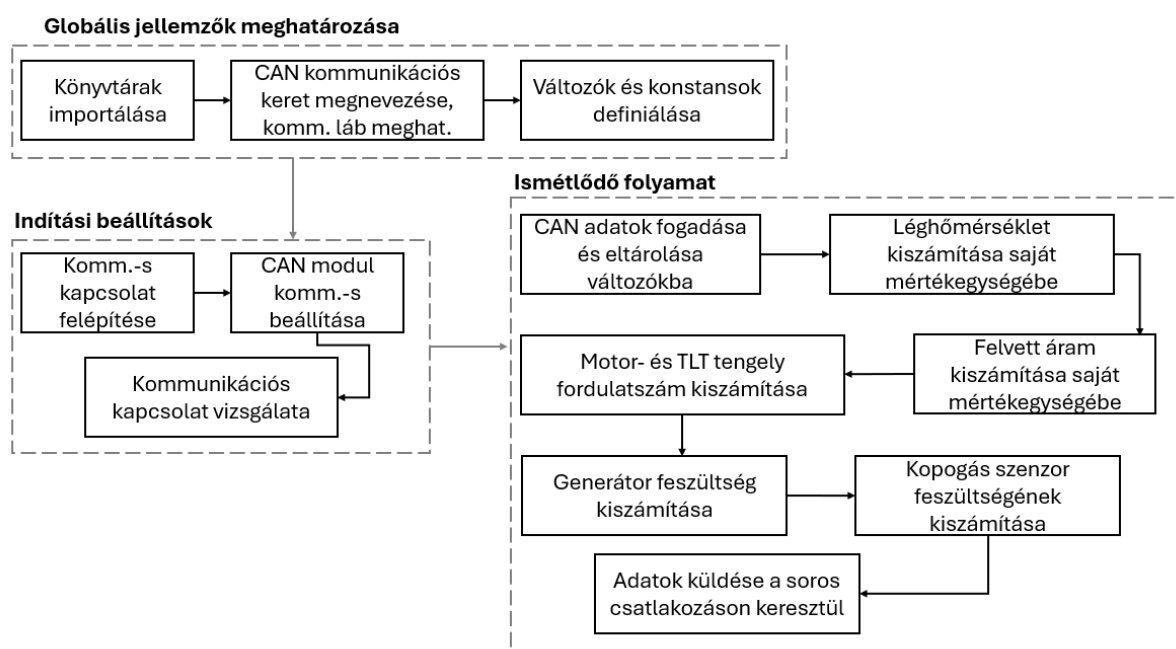


3.14. ábra: A mérőprogram lefutás diagramja

Ez azért fontos, mert mindegyik jellemzőt egy kicsit másként kell mérni, példának okáért a hőmérséklet mérése nem egy bonyolult folyamat, alapvetően pár sornyi kódból meg lehet határozni a mérést, ezzel szemben viszont a fordulatszám mérés bonyolultabb és egy sokkal hosszabb kód tartozik alá.

Hozzátenném hőmérés és hőmérés között is lehet különbség, például a két darab DS18B20 hőszondának van külön letölthető könyvtára, amely megkönnyíti az adatgyűjtést a programszinten is, míg a léghőmérőnek nincs ilyen könyvtára, így tehát kénytelen vagyok matematikai összefüggéseket, egyenleteket használni a hőmérséklet értékénke meghatározásához.

A mérőprogram struktúrájához képest a kiértékelő elektronikához tartozó kódsor valamivel egyszerűbb és kevesebb is. Ez abból adódik, mivel az adatok egy része egy az egyben kész jellemzőként érkezik a kiértékelő egységhez, másfelől a kiértékelés igazi mibenléte a beérkező információk számunkra hasznos formába öntése, melyek matematikai számításokkal, átalakításokkal érhetőek el. Ezen túlmenően az adatok hosszabb távú eltárolását Excel táblázat segítségével végzem, melyben egy beépített modul segítségével közvetlenül a mikrovezérlőről tudom az adatokat felvezetni, majd ezt követően utómunkával kiértékelni, diagramok formájában. A kiértékelő elektronika program lefutási diagramja a 3.15. ábrán látható.



3.15. ábra: A kiértékelő program lefutás diagramja

3.3. Fizikai megvalósítás

A dolgozatom teljesértékűségéhez céloomul tűztem ki a tervezett adatgyűjtő elektronika megépítését is. A megépítés és megvalósítás részletes dokumentációja ebben a részben kerül

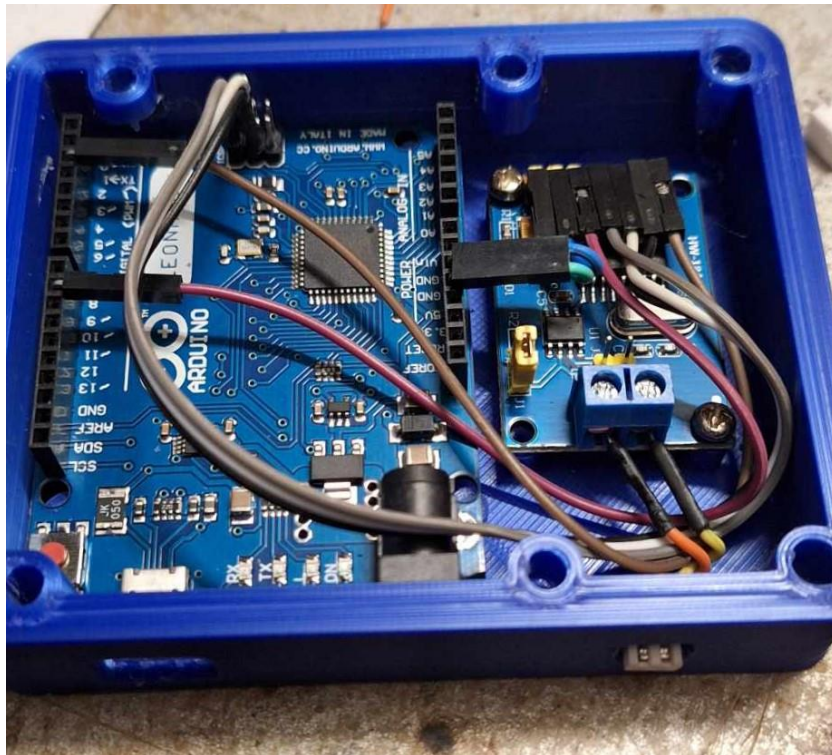
feljegyzésre. Az építés kiterjed külön a mérő- és kiértékelő elektronika összekötésére, a tokozásba szerelésre és a csatlakozók forrasztására. Ezen felül a választott jármű érzékelőkkel való felszerelése és az ehhez szükséges egyéb elemek, tartók kialakítása is ebben a fejezetben kerül kitárgyalásra, valamint ide venném a jármű bekábelezését is.

3.3.1. Az elektronika összeépítése

Legyen szó szinte bármilyen elektronikáról a valóságban megépített formájuk sokszor ránézésre nem annyira triviális mint mondjuk egy bekötési rajz. Ez úgy vélem ez esetben sincs másként annak ellenére, hogy nem nevezném az általam kialakított konstrukciót igazán bonyolult rendszernek, de kellőképp komplex ahhoz, hogy könnyen hibát vétsek a bekötések során. Éppen ezért egy szisztéma felvezetése vagy egy bekötési sorrend terv mankóként segítheti a munkámat, annak érdekében, hogy sikeresen összeálljon a rendszer viszonylag kis hibalehetőséggel. Szóval első lépésként meghatározom a bekötési sorrendet mindkét elektronika esetében.

Elsőként a könnyebbik elektronikával, a fogadó- és feldolgozó egység bekötésének sorrendjét és összeszerelését határoznám meg. Ez tényleg annyira egyszerű talán, hogy nem is feltétlen szükséges hozzá terv, viszont érdemes ettől függetlenül elvégezni, mert a mérő elektronika összeszerelési tervénél hasznát vehetem, hogy már egyszer kitaláltam bizonyos lépéseket. Tehát a szerelést azzal kezdem, hogy előkészítem a tartó dobozt, ez annyit tesz, hogy kis furatokat készítek az emelvényként szolgáló tűskékre. Ezt a tervezés során azért nem tettem meg, mert nem tudtam minden modult és elemet pontosan beméretezni, illetve az interneten is megoszlóak voltak az információk a modulok méreteit illetően. Ezért hozzávetőlegesen és elnagyoltan alakította, ki ezeket a tűskéket, így bebiztosítva, hogy jó helyre kerülnek majd a modulokat a helyükön tartó csavarok. Ezt követően behelyezhető a mikrokontroller és a CAN modul is. Később vettem észre csak sajnos, hogy az általam tervezett tartó doboz magassága nem elég ahhoz, hogy normál csatlakozókat használjak, sőt a fedél leszorításához kialakított egyik furatot is le kellett rövidítenem, mert nem fért be alá a mikrovezérlő. Ennek a problémának a megoldására a CAN modul csatlakozó tűskéit elhajlítottam annyira, hogy ne okozzon gondot a fedél felhelyezésénél, viszont a megfelelő kontaktust is fenn kellett tartanom. A mikrovezérlőbe futó vezetékek csatlakozójának végét is meg kellett rövidítenem, hogy azok se akadjanak bele a fedélbe felhelyezésnél. Az átalakításokat, a modulok behelyezését és rögzítését követően csak a bekötés maradt vissza.

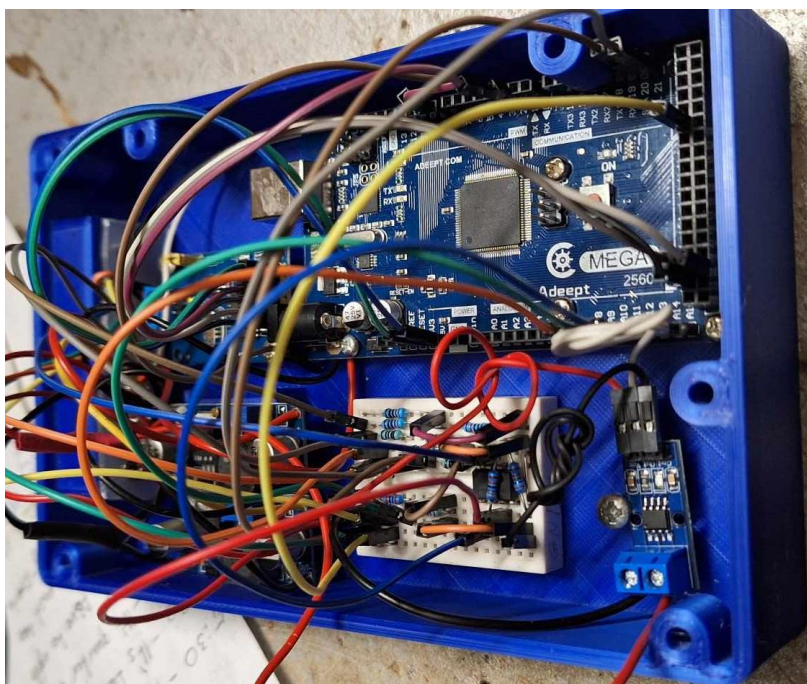
A bekötésnél két szempontra ügyeltem, az egyik, hogy a vezetékek, ne csak úgy szanaszét legyenek elhelyezve, hanem legyen egy rendszer benne és amennyire csak lehet rejtsem el azokat, így esztétikusabb kivitelt elérve. A másik szempont, hogy igyekeztem viszonylag egyértelműen, különböző színű borítással rendelkező vezetékeket használni, ez egyébként a bekötést és a hibakeresést könnyíti meg. A bekötéshez tartozik a CAN egység CAN_H és CAN_L vezetékeinek bekötése is, melyet egy egyszerű két pin-es csatlakozóval oldottam meg. Az elkészült bekötést a 3.16. ábra mutatja be. Végso lépésként felkerült a fedél a tároló tetejére.



3.16. ábra: Az összeszerelt kiértékelő elektronika

A mérő elektronika összeszerelése csak részben tér el az előzőktől. A modulok és a mikrovezérlő felfogatási pontjait ez esetben is meg kellett fúrnom és kisebb anyagleválasztásra volt szükség a nagy csatlakozó helyénél mert nem tudtam a két felet össze tolni. Ezen felül viszont egyéb módosítást nem kellett eszközölnöm a tároló egységen, ennek a magasságát megfelelőre terveztem. Következő lépésként a csatlakozó felület beillesztését és rögzítését végeztem el, majd a csatlakozó kivezetéseire ráforrasztottam a szükséges vezetékeket. Szintén forrasztást igényeltek a tápforrások ki- és bemenetire kerülő vezetékek, valamint a mikrovezérlő tápforrásául szolgáló vezetékek elhelyezése. Ezután bekerültek a modulok és a próbapanel, majd rögzítettem őket egyesével. Miután beszereltem mindent a tároló dobozba a próbapanelen elkezdtem összeszerelni a különböző érzékelők egységeit, kiegészítő elemeit,

majd ezzel együtt a mikrovezérlőbe is bekötöttem a megfelelő helyekre az érzékelők kivezetéseit sorban, így lecsökkentve az esélyét annak, hogy valamit elhibázok. A bekötés során különös figyelmet kellett fordítanom arra, hogy az érzékelők majd a belső 12 V-os, valamint 5 V-os feszületeiről kapják majd a tápforrást, és a fő csatlakozón keresztül jönnek vissza az adatot hordozó kábelek. Az érzékelők feszültséggel való ellátásáról a tároló egységen kívül gondoskodok majd, és a jármű vezetékezésé során kötöm össze ezeket a kábeleket. Ezesetben is igyekeztem ügyelni az esztétikumra és praktikumra, tehát a vezetékek színjelöléssel, akkurátusan elrendezve kerültek beépítésre. a belső elektronika összekötése után felkerült a fedél is a tároló tetejére. Az elkészült mérőelektronika a 3.17. ábrán látható. Itt jegyezném meg azt is, hogy a mérőelektronika csatlakozójának másik fele egy régi számítógép tápról lett leszerelve, levágva, így ennek a csatlakozónak a színjelölését nem variáltam. Ugyanakkor, hogy ne keveredjek bele melyik kábel milyen funkciót lát el, felcímkéztem őket a szerelés során.



3.17. ábra: Az összeszerelt mérő elektronika

3.3.2. A jármű felszerelvényezése

Elsősorban, mindennek meg kellett találnom a megfelelő helyet, az érzékelőknek, a kábeleknél és a tároló egységeknek. Ez koránt sem annyira egyértelmű mint ahogy azt elsőre gondolni lehet. Figyelembe kell venni minden egyes apró részletet és az alapján kialakítani a megfelelő pozíciót a különböző egységeknek. Az érzékelők elhelyezése viszonylag egyszerű

mert vagy már van kialakított helye a hagyományos érzékelőknek, vagy egyszerűen adott, hogy hova a legcélszerűbb elhelyezni az adott érzékelőt. Ugyanakkor, vannak érzékelők, melyeknek adott a helye, de átalakításokat, kiegészítéseket, toldást igényelnek. A kábelek elvezetése is egyfajta szabályt kell kövessen, főleg ha a motortéren keresztül kell mennie, hiszen üzemelés közben a motor hője károsíthatja a nem megfelelően elhelyezett vezetékeket. Emellett a tároló dobozok felszerelése, pontosabban a mérő egység helye is kritikus, főként a hőterhelés miatt, de az egyéb külső környezeti hatások miatt is. Ezért az utóbb felsorolt szempontokat követve végig vettem mit és hova helyezhetek el biztonsággal és úgy hogy megfelelő hatások ériék az érzékelőimet és a többi eszközt is.

Először is két elektronika elhelyezésével kezdenék, mert ezeknek egészen konkrét helyet szántam. Annak érdekében, hogy a jármű vizsgálata könnyen és kézenfekvően működjön a kiértékelő elektronikának a fülkében kell helyet kapnia, így egyszerűen rá tudok csatlakozni egy lappal és a járművet különböző üzemállapotaiban is tudom vizsgálni. Tudom emelni és csökkenteni a fordulatszámot, valamint kapcsolni a különféle villamos berendezéseket. Ezek mind fontos részei lesznek a mérésnek. A mérőelektronikát ezzel szemben a motortérben helyezem el, hiszen ott van a legközelebb a legtöbb érzékelőhöz, így csökkentve a vezetékek hosszát, ami szintén javít a mérés hatásfokán, még ha csak egy keveset is, de nem torzít annyit



3.18. ábra: A kiértékelő elektronika elhelyezése

a vizsgált értékeken a kábelhosszon kialakuló ellenállás. Azonban a motortér egy olyan részén kell elhelyeznem a tartómat az elektronikával, ahol nem éri akkora hőterhelés, hiszen az károsíthatná a tartót is és az elektronikát is, nem beszélve arról, hogy alapvető pontosság csökkenést is okozhat a motor hőterhelése. A mérő elektronika elhelyezése a jármű leges legelején lévő kereszt tartón kapott helyet és alulról lett csavarral felfogatva. Ezt a 3.18 és 3.19. ábra szemlélteti az elektronikát tartalmazó egységek elhelyezését.

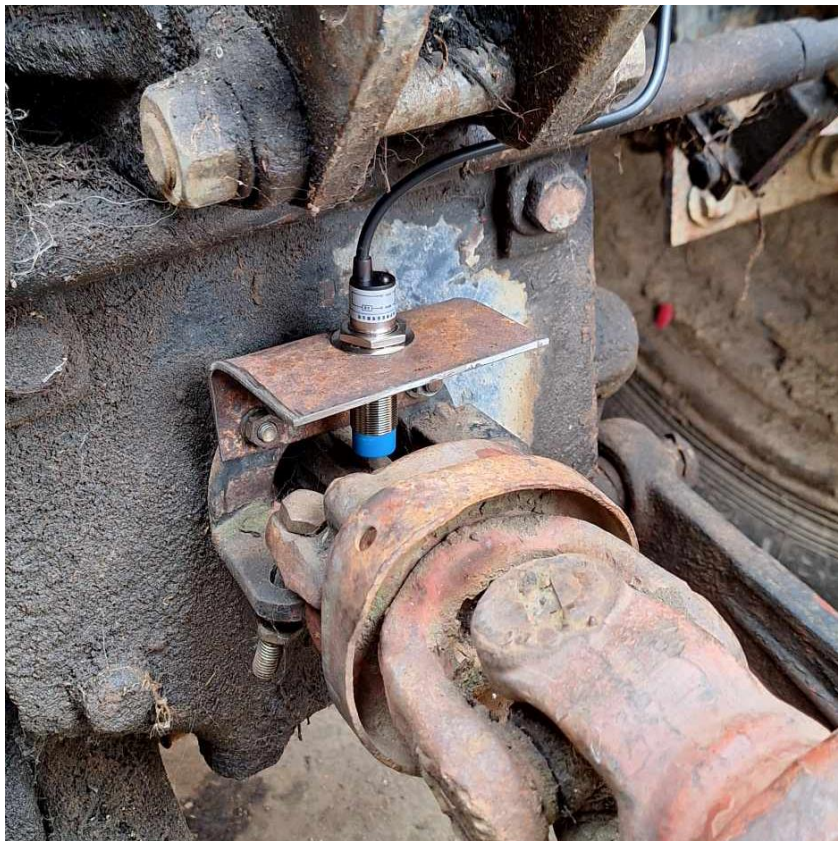


3.19. ábra: A mérőelektronika elhelyezése

A villamos jellemzők méréséhez a fizikai kontaktust is a jármű fülkéjében alakítottam ki. A tápforrást és a generátor töltését vizsgáló alegység csatlakozását az ülés alatt található akkumulátorról vettem le egy egyszerű csipesz rögzítésével. Ezen kívül az árammérő elektronika bemeneteinek másik felét a járműben található hagyományos árammérő egység

kivezetése után kötöttem be, így közvetlen az indítás és izzítás nem teszi tönkre a mérőmodult, viszont a jármű egyéb villamos berendezéseinek áramfelvétele mérhető marad. A bekötéséhez a tartó csavarok leszerelése után kifordítottam az egész műszerfalat, így szabaddá vált a bekötéshez minden. Ezután mindent visszaállítottam az eredeti pozíciójába a fülkén belül.

A két fordulatszám jeladó felszerelése már nem volt ennyire egyszerű. Először is a hátsó erőleadó tengely méréséhez szükséges érzékelő elhelyezését bontanám ki. A jármű hátsó felére fel volt szerelve egy szalagfűrés, ezért az egész hátsó részhez való hozzáférésem korlátozott volt, lekasztásra lehetőségem nem volt. Ennek ellenére meg tudtam oldani a szenzor felszerelését. Először is, közvetlenül a kihajtó tengely felett illetve körül találtam egy rögzített lemezt, melyen csavarok voltak elhelyezve. Ezekhez a csavarokhoz rögzítettem később az érzékelő tartólemezét. A tartólemez egy hagyományos szerkezeti acél lemezből vágtam ki majd feljelöltem a két furat helyét és kifúrtam. A furatok elkészítése után méretre hajtottam a lemezt egy 90°-os hajtással. Ezután az érzékelőnek is készítettem egy furatot, ügyelve arra, hogy jó pozícióban lásson rá a kihajtó tengely bordázott végére. Végül a már fent lévő csavarokra felfogattam a tartó lemezt és beállítottam a szenzort, úgy hogy a bordacsúcsok közeledésével érzékeljen a szenzor, a bordavölgyek viszont ne képezzenek jelet. Az elkészült kialakítást a 3.20. ábra mutatja be.



3.20. ábra: A hátsó erőleadó tengely jeladójának elhelyezése

A motor fordulatszámának vizsgálatát a forgattyús tér felől oldottam meg. A motor főtengelyének elülső kihajtása a vezérlést szabályozza és vezérli mechanikusan, fogaskerékes hajtásokkal. Ennek a fogaskeréknek a fogait és fogárkait használtam fel a motor fordulatszámának meghatározásához. A rögzítést az olajteknő megfűrésével értem el és a beállítás koránt sem volt egyszerű, hiszen az olajtartájt minden egyes alkalommal le kellett vennem mikor állítani akartam az érzékelő távolságát, viszont ahhoz, hogy tudjam jó helyen van-e az érzékelő vissza kellett helyeznem azt. A beállítást követően gondoskodtam a megfelelő rögzítésről és szigetelésről, hogy a motorolaj ne tudjon szivárogni az érzékelő mellett. Ezután véglegesen felhelyeztem az olajteknőt és elvezettem a szenzor vezetékeit biztonságosan.

A kopogásérzékelő elhelyezésénél a fő szempont az volt, hogy könnyen hozzá lehessen férni, ne kelljen komolyabb bontás alá vetni a jármű motorterét. ezért az üzemanyagszűrőt nem messze a hármashenger közelében megfűrtam a motorblokkot és bemeneteltem. Az érzékelő érintkező felületét a motorblokkon körültekintően letisztítottam és felcsiszoltam. Ez fontos a megfelelő érintkezéshez, ami elengedhetetlen a jó méréshez. Ezután csak fel kellett helyeznem és gondosan meghúznom a rögzítő csavart. A 3.21. ábra bemutatja az érzékelő pontos helyét.

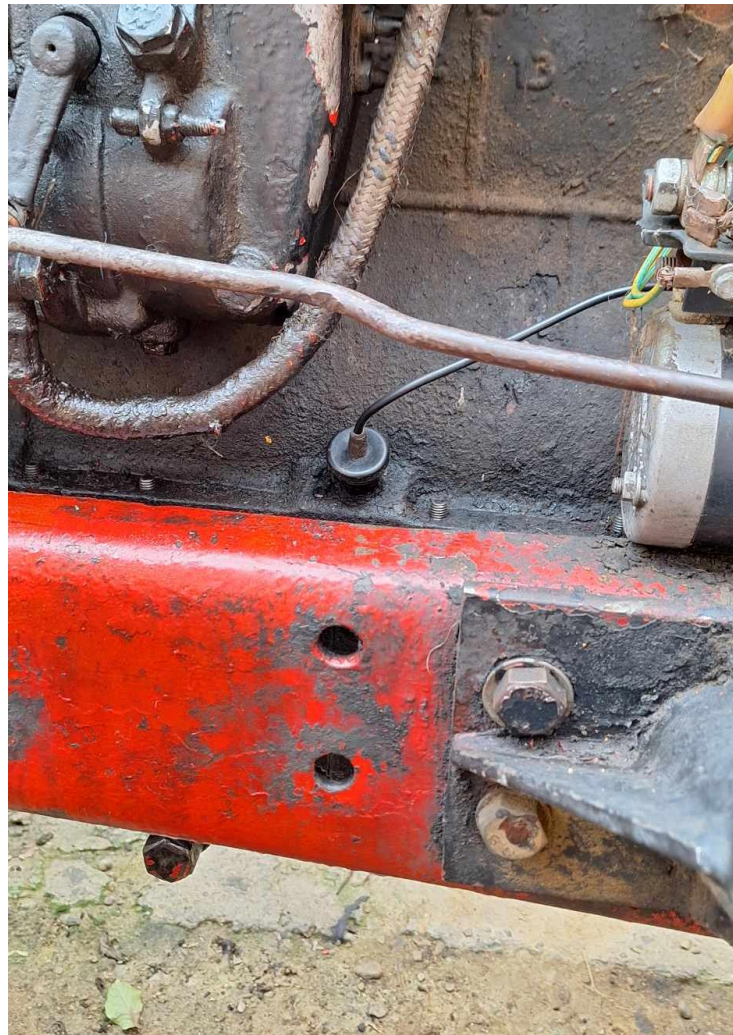


3.21. ábra: A kopogásérzékelő elhelyezése

A beszívott levegő hőmérsékletének érzékelését, mondhatni egyértelmű módon a szívócsőben kell elvégezni. Ennek megoldásához a szenzort a szívócső egy adott szakaszán található gumicsőben helyeztem el. Ehhez a gumicsövet meg kellett bontanom, melyet egy

éles szike segítségével végeztem. Egy megfelelő méretű lyukat készítettem a cső tetején, melybe így már bele tudtam helyezni az érzékelőt amennyire csak tudtam. Külön rögzítés itt nem szükséges, hiszen a gumicső rugalmassága elegendő szorító erőt képez a szenzor megtartásához.

Végül a két folyadék hőmérőszonda elhelyezése maradt. Elsőként a motorolaj hőmérséklet érzékelőjét helyeztem el. Az hőszondát nem annyira praktikus, de egyszerű módon a szintjelző pálcának kialakított csövön engedtem le. Viszont hogy a motor járása közben az olaj ne csapódjon ki a szabad csővégen ezért egy műanyag zárókupakon is keresztül engedtem majd gondosan letömítettem a vezeték környékét, ezzel megakadályozva az olaj-kiverődést és az érzékelő elmozdulását. A kupak elegendően nagy belső süllyesztéssel rendelkezett, ezért nem volt szükségem ezt módosítani. A 3.22. ábrán látható a megvalósítás.



3.22. ábra: Az olaj hőszonda elhelyezése

A vízhőfokszonda elhelyezése nehezebb volt. Úgy kellett megoldanom, hogy a jármű eredeti, hagyományos vízhőfok jeladóját ne iktassam ki, de üzemeljen az általam beszerelt érzékelő is. Megoldásként az eredeti jeladó felfogatásán alakítottam ki az én szenzorom helyét is. A csatlakozás kilógó csövének összeeresztésénél egy ferdén fúrt furattal oldottam meg a saját érzékelőm beeresztését a motor vízterébe. Ehhez előbb ki kellett tekernem az eredeti érzékelőt, hogy ne tegyek kárt abban, majd a furat elkészülte és a hőszonda behelyezése után visszahelyezhettem a jármű saját érzékelőjét. A teljes összeszerelés után a készített furatot le kellett tömítenem, hogy a hűtőfolyadék ne szökhessen el a szenzor vezetéke mellett. Ezt kétkomponensű epoxy gyanta alapú ragasztóval oldottam meg, melyet a furat és a vezeték köré körben gondosan elkentem.

A vezetékek csatlakozása és elvezetése a jármű járás szerkezetéül szolgáló „U” alakú vázszerkezetében került elhelyezésre és rögzítésre. A kábeleket gyorskötöző használatával összerendeztem és fixáltam is egyben a vázban. A fülkébe a mérőelektronikától csak a CAN kommunikációhoz szükséges vezetékek futnak fel egészen a kiértékelő elektronikáig. Ezzel összeállt az adatgyűjtő szerkezet.

4. A mérés menete

A mérést időszűke miatt sajnos nem tudtam elvégezni, azonban mindenképp fontosnak tartottam ezt a részt is megtölteni tartalommal. Ezért ebben a fejezetben a tervezett mérés elméleti lefolyását tárgyalom, valamint hogy milyen lépések, mérések lettek volna ha elegendő időm állt volna rendelkezésre. Mindenek előtt még egyszer részletezném mit is takart volna a mérés. A vizsgálat során a legfontosabb motor jellemzők mellett villamos tulajdonságok mérése lett volna a fő feladat. A motorikus jellemzők közül a hűtőfolyadék hőmérséklete, a motorolaj hőmérséklete, a beszívott levegő hőmérséklete, a főtengely fordulatszáma, valamint a motor kopogás vizsgálata volt tervben. Villamos jellemzők közül a jármű villamos hálózatának áramfelvétele és a generátor töltésének meghatározása lett volna a feladat. Emellett nem tisztán motorikus jellemző, de közvetve odaköthető a hátsó erőleadó tengely fordulatszáma is. Ennek, az összesen nyolc jellemzőnek mindenképp fontos egy gépjármű diagnosztikai vizsgálata során mérni, főként az erőgép technikában.

Három típusú mérést terveztem elvégezni. Egyet állandó fordulatszám mellett, ezzel vizsgálva mennyire képes a jármű állandó fordulatszámot tartani, valamint megfelelő-e a kopogási kép, tehát hogy a motor egyenletesen és zavartalanul képes járni. Emellett a villamos jellemzőknek is állandónak kell maradniuk ebben az esetben vagy legalábbis egyenletesen változónak. A hőmérsékleteket tekintve ez esetben a hűtőfolyadék és motorolaj hőmérsékletének lassan fel kellene kúsznia az üzemi hőfokra. A hűtőfolyadék túlzott melegekedése hűtési problémákra vezethető vissza, legtöbbször valamilyen termosztát hibára. A túlzott olajmelegeedés is hasonló hűtési, illetve kenési problémákra utalhat. Ezért fontos állandó fordulaton járó motor vizsgálata, hiszen rengeteg információt hordoz egy ilyen vizsgálat eredménye.

A második fajta mérés egy dinamikus mérés lett volna. Ezúttal egy alapláthatoz közeli fordulatszámról indulva, folyamatos gázadással közel maximális fordulatiá járatam volna a járművet, majd elérve a csúcs fordulatiot egyenletes gázelvétellel csökkentettem volna motor fordulatiát alapláthat, vagy fulladás közeli állapotiá. Ezesetben is főként a hőmérséklet érzékelők kaptak volna nagyobb hangsúlyt a vizsgálat közben. Szintén az üzemanyagok (hűtőfolyadék, motorolaj) melegekedése, illetve ideális esetben hűtése vagy a hőmérséklet szinten tartása az érdekes, hiszen a fordulatszám emelésével a motor hőmérséklete is növekedésnek indul a szaporább égési ciklus okán. Emiatt kritikus ez a vizsgálat is, hogy

vizsgáljam a motor kenési és hűtési rendszerét. Emellett a kopogás érzékelés kapott volna még nagyobb hangsúlyt, mert a kopogási képnek a változása a fordulatszám tekintetében egy nagyon látványos jelenség. A kopogás érzékelésnek a vezérlés hibáinak kimutatására is alkalmas, hiszen a „csattogó” motor járást is jól képes szemléltetni. Sőt a befecskendező rendszerben fellépő hibákat vagy hiányosságokat is szemléletesen kimutatja ez a szenzor. Ennek a vizsgálatnak egy másik hozadéka, hogy a generátor töltését is tudjuk vizsgálni a változó fordulatszám mellett. Pontosabb esetben inkább a feszültség szabályzó elektronikát, ugyanis a változó motorfordulatszám és így a generátor fordulatszámának ellenére is viszonylag állandó töltő feszültséget kell biztosítania a generátornak.

A harmadik vizsgálat szintén egy stationer vizsgálat lett volna, viszont ez esetben inkább a villamos rendszer fogyasztását és áramfelvételét emeltem volna ki. Ennek a vizsgálatnak a keretein belül a jármű különböző villamos berendezéseit, mint fényszórók, fülke ventilátor, kűrt, vagy például munkalámpák és féklámpák. Ezúttal a villamos berendezéseket állandó fordulaton járó motor mellett egyesével kapcsolgattam volna be és ki, majd egymás után többet működtettem volna egyszerre. Ennél a vizsgálatnál fontos, hogy ne egy időpillanatban legyen minden felkapcsolva, mert az aktiválás időpillanatában minden fogyasztó nagyobb áramértéket vesz fel, mint már működés közben. Viszont az egyes fogyasztók egyszerre történő működtetése során a rajtuk folyó áram értéke összeadódik, ezért itt ügyelnem kell arra is, hogy véletlenül se terheljem túl a Hall-elemes mérőmodult mert az esetleg tönkre teheti azt. Továbbá megfigyelhető lett volna, hogy az egyes fogyasztók bekapcsolásakor a feszültség egy pillanatra lecsökken. Ezek a feszültség csökkenések kis mértékűek, nagyság rendileg 0,1 V vagy akár még kevesebb, így tehát nem annyira látványos de érdekes tény, mely vizsgálható és kimutatható is akár.

Ezt követően már csak a kapott és gyűjtött adatok kiértékelése maradt volna hátra. A mérések eredményeit egy Excel bővítmény segítségével közvetlenül a mikrovezérlőtől egy táblázatban gyűjtöttem volna, ahol egyszerűn készíthettem volna szemléltető diagramokat minden egyes vizsgált jellemzőre vonatkozóan. Ezeket a diagramokat aztán megvizsgálva konklúziót vontam volna a mérésre vonatkozóan, illetve a szenzorokra és az elektronikára is. Feltártam volna az esetleges hibákat, mind a járműét, mind az adatgyűjtőét, melyekre megoldási javaslatot is tettem volna javításra vagy fejlesztésre, esetleg teljes konstrukciós átalakításra.

5. Összefoglalás

Összegezvén a dolgozatom tapasztalatait elmondható, hogy az elektronika és programozás, ma már olyan szinten van, hogy viszonylag egyszerű ilyen és ehhez hasonló projekteket véghez vinni. Hovatovább, rengetegen foglalkoznak járműipari fejlesztésekkel, illetve mások is ütköztek hasonló problémába mint én, szintén egy dolgozat vagy beadandó feladat keretein belül, így én is könnyedén utána tudtam nézni az esetlegesen felmerülő problémáknak, habár hozzá kell tegyem, főként angol nyelvű forrásokat és megoldásokat kerestem. Az érzékelők kiválasztása, méretezése az egyéb villamos és elektronikus elemek kiszámítása még így is komplex feladat, ahhoz hogy hobbi szinten művelje az ember, és tény hogy egy hasonló projekt kivitelezése komoly felkészülést és háttértudást igényel. Mindig is érdeklődtem a villamos rendszerek és a diagnosztika iránt, ezért adott volt ennek a feladatnak a választása minden bonyolultsága ellenére. További kihívást jelenthet olyan érzékelők használata, melyről kevés információ áll rendelkezésre és próbamérésekkel lehet csak megállapítani az adott szenzor sajátosságait és tulajdonságait. Mindemellett a korrekciók és kompenzációk alkalmazása is elengedhetetlen ha mérésről van szó.

A méretezésen és számításokon túl érdekes feladat és tapasztalás egy már meglévő konstrukció oly modul átalakítása, hogy azzal ne befolyásoljam a végbemenő folyamatokat és mégis használható és értékelhető adatokat nyerjek ki, melyek utalnak az adott gépjármű aktuális állapotára. Ebbe nem csak az érzékelők és vezetékek elhelyezése játszik szerepet, hanem magát a mérést és kiértékelést végző elektronika felszerelése is. Számomra talán ez volt az egyik legizgalmasabb része a feladatnak, mikor láttam a papíron és elméletben kitalált szerkezet fizikai megvalósulását. Elengedhetetlennek tartom egy tervezési folyamatban a prototípus vagy próba építés folyamatát, hiszen annnyival többet tud adni egy konstrukció véglegesítéséhez, mint semmi más.

Végül a tesztelés és mérés, mely szintén fontos része ennek a dolgozatnak, sőt alapját képezte mindennek, ugyanis a fő cél egy olyan eszköz tervezése és megvalósítása, mely képes bizonyos járműjellemzőket vizsgálni és hasznosítható adatot készíteni belőle, ezzel utalva a jármű állapotára, esetleges meghibásodásaira. A mérés, mint folyamat véleményem szerint az egyik legkomplikáltabb mérnöki tevékenység. Végtelen lehetőség, milliónyi probléma megoldására, villamos és hagyományos mérnöki ismereteket, valamint informatikai tudást és programozási készségeket egyaránt igényel.

6. Summary

Summarizing the experience of my thesis, I can be said that electronics and programming are now at such a level that it is relatively easy to carry out such and similar projects. Furthermore, there are many people dealing with vehicle developments, and others have encountered a similar problem to me, also a thesis or an assignment to be submitted, so I was also able to easily look up any problems that might arise, although I must say that I mainly looked for sources and solutions in English. The selection and sizing of sensors and the calculation of other electronic elements is still a complex task for a hobbyist, and it is a fact that the implementation of a similar project requires strong preparation and background knowledge. I have always been interested in electrical systems and diagnostics, and that is why I chose this task despite all its complexity. An additional challenge can be the use of sensors for which little information is available, and the specifics and properties of the given sensor can only be established with test and measurements. In addition, the application of corrections and compensations is essential when it comes to measurement.

Beyond sizing and calculations, it is an interesting task and experience is the transformation of an existing construction in such a way that I do not influence the ongoing processes and still obtain usable and evaluable data, which indicate the current condition of the given vehicle. Not only the placement of the sensors and wires plays a role in this, but also the installation of the electronics that carry out the measurement and evaluation. For me, this was perhaps one of the most exciting parts of the task, when I saw the physical realization of the structure invented on paper and in theory. I consider the process of building a prototype or test to be essential is a design process, since it can give so much more to the finalization of a construction than anything else.

Finally, testing and measurement, which is also an important part of this thesis, and even formed the basis of everything, because the main goal is the design and implementation of a device that can examine certain vehicle characteristics and produce useful data from it, thus indicating the condition of the vehicle and possible malfunctions. In my opinion, measurement as a process is one of the most complicated engineering activities. Endless possibilities to solving millions of problems and require both electrical and traditional engineering skills, as well as IT knowledge and programming skills.

7. Köszönet nyilvánítás

Nem sokan, de a környezetemben lévők szinte mindegyike segítette munkámat és hozzájárultak ahhoz, hogy a dolgozatom sikeresen elkészítsem. Nekik, nektek szeretnék ezúttal köszönetet nyilvánítani.

Elsőként szeretném megköszönni egyetemi konzulensem Dr. Gergely Zoltán Albert alázatos munkáját és köszönetem illeti azért, hogy terelte, igazgatta a dolgozatom elkészülését, illetve hogy jótanácsokkal és hasznos tippekkel látott el, amikor csak szükségem volt rá. Köszönet illeti külső konzulensem is Vizeli István Andrást segítőkészségéért és tudásának átadásáért, valamint közbenjárásáért, ami a beszerzett érzékelőket illeti. Ezen felül az egyik legnagyobb hálát vótársam Sári Csabát illeti, aki rendelkezésre bocsátotta saját sárművét az átalakításra és szintén hasznos tanácsokkal látott el a konstrukció kialakításában, megvalósításában. Külső konzulensem közbenjárásával szeretném megköszönni Zanykó Konrádnak, hogy a használt, de jó állapotú érzékelőit feláldozta a dolgozatom elkészítésének sikeressége érdekében. Végezetül szeretném megköszönni anyasomnak Fers Ferencnének és feleségemnek Agárdi-Fers Adél Rózsának a végtelen türelmüket és a szüntelen támogatásukat, mert úgy vélem enélkül koránt sem készült volna el ez a dolgozat.

8. Irodalomjegyzék

- [1] Bouwens, A. J. (1984): *Digital instrumentation*. New York: McGraw-Hill Book Company
- [2] Buzás M., Dr. Nagyszokolyai I. (1993): *Gépjármű-elektronika I.* H.n.: Jaurinum Bt.
- [3] Dr. Ember M., Dr. Jánossy Gy., Dr. Szijjártó O. (1983): *Mezőgazdasági gépek javítása*. Budapest: Mezőgazdasági kiadó
- [4] Dr. Fekete A., Dr. Judák E., Dr. Sembery P., Dr. Tóth L. (2002): *Elektronika és automatika a mezőgazdaságban*. Budapest: Szaktudás Kiadó Ház
- [5] Dr. Grave, H. F. (1968): *Nemvillamos mennyiségek villamos mérése*. Budapest: Műszaki Könyvkiadó
- [6] Dr. Kaposvári Z. (1980): *Méréselmélet*. Budapest: Tankönyvkiadó Vállalat
- [7] Dr. Kovács F. (1976): *MOS integrált áramkörök*. Budapest: Műszaki könyvkiadó
- [8] Dr. Lakatos I., Dr. Nagyszokolyai I. (2003, 2006): *Gépjárműdiagnosztika*. H.n.: Képzőművészeti Kiadó Kft.
- [9] Dr. Mojzes I. (főszerk.) (1995): *Mikroelektronika és elektronikai technológia*. Budapest: Műszaki Könyvkiadó
- [10] Dr. Reif, K. (2014): *Automotive Mechatronics – Automotive Networking, Driving Stability, Electronics*. Friedrichshafen: Springer Vieweg
- [11] Dr. Schnell L., Tóth M. (1967): *Általános metrológia*. Budapest: Tankönyvkiadó Vállalat
- [12] Dr. Tornai K. (szerk.) (1984): *Mikroelektronikai berendezésorientált áramkörök tervezése*. Budapest: Edusystem
- [13] Lambert M. (2009): *Szenzorok – elmélet és gyakorlat*. Budapest: IM Kiadó és Mérnöki Iroda Kft.
- [14] Mészáros I., Sitkei Gy. (1965): *Mezőgazdasági gépek vizsgálata*. Budapest: Akadémia Kiadó
- [15] Staudt, W. (1988): *Gépjárműtechnika*. Székesfehérvár: OMÁR Bt.
- [16] Tölgyesi Z. (2005): *OBD-Fedélzeti diagnosztika*. Budapest: Maróti Könyvkereskedés és Könyvkiadó Kft.
- [17] http 1: https://www.ksh.hu/stadat_files/gsz/hu/gsz0003.html (2024.03.23.)
- [18] http 2: https://robotepites.blog.hu/2017/11/12/arduino_alapok (2024.06.23.)
- [19] http 3: <https://www.flexihub.com/oobd2-pinout/> (2024.08.25.)
- [20] http 4: <https://www.asocietygroup.com/en-us/news/electronic-control-unit> (2024.08.26.)

- [21] http 5: <https://en.wikipedia.org/wiki/Tinkercad> (2024.09.04.)
- [22] http 6: <http://ecoursesonline.iasri.res.in/mod/resource/view.php?id=3605> (2024.09.24.)
- [23] http 7: <https://mernokkapu.hu/hogyan-mukodnek-a-nyomaserzekelo-szenzorok/>
(2024.09.24.)
- [24] http 8: <https://open-the-hood.com/motorfordulatszam-jelado-mukodese-es-diagnosztizalasa/> (2024.09.24.)
- [25] http 9: <https://www.dewetron.com/2023/01/how-do-rpm-and-angle-sensors-work/>
(2024.09.24.)
- [26] http 10: <https://www.hioki.com/us-en/learning/methods/voltage-measurement-methods.html> (2024.09.26.)
- [27] http 11: <https://dronebotworkshop.com/dc-volt-current/> (2024.09.26.)
- [28] http 12: <https://www.instructables.com/Voltage-Measurement-Using-Arduino/>
(2024.09.26.)
- [29] http 13: <https://club.autodoc.hu/magazin/mi-kopogaserzekelo> (2024.09.26.)
- [30] http 14: <https://hu.wikipedia.org/wiki/Piezoelektromoss%C3%A1g> (2024.09.26.)
- [31] http 15: <https://valstat.ktk.pte.hu/tdist.html> (2024.09.29.)
- [32] http 16: <https://automotive.softing.com/standards/bus-systems/can-iso-11898-can-fd-iso-11898-1.html> (2024.10.08.)

9. Nyilatkozatok

NYILATKOZAT

AGÁRD RÓBERT FLÓRIÁN (név) (hallgató Neptun azonosítója: P65092)
konzulenseként nyilatkozom arról, hogy a
záródolgozatot/szakdolgozatot/diplomadolgozatot/portfóliót¹ áttekintettem, a hallgatót az
irodalmi források korrekt kezelésének követelményeiről, jogi és etikai szabályairól
tájékoztattam.

A záródolgozatot/szakdolgozatot/diplomadolgozatot/portfóliót a záróvizsgán történő
védésre javaslom / nem javaslom².

A dolgozat állam- vagy szolgálati titkot tartalmaz: igen nem^{*3}

Kelt: 2024 év 11 hó 11 nap

Geszty Zoltán
belső konzulens

¹ A megfelelő dolgozattípus meghagyása mellett a többi típus törlendő.

² A megfelelő aláhúzandó.

³ A megfelelő aláhúzandó.

NYILATKOZAT

a szakdolgozat nyilvános hozzáféréséről és eredetiségéről

A hallgató neve: Agárdi Róbert Flórián
A Hallgató Neptun kódja: P65O9Z
A dolgozat címe: Gépjármű-adatgyűjtő tervezése mikrokontrollerrel
A megjelenés éve: 2024
A konzulens intézetének neve: Műszaki Intézet
A konzulens tanszékének a neve: Járműtechnika Tanszék

Kijelentem, hogy az általam benyújtott szakdolgozat egyéni, eredeti jellegű, saját szellemi alkotásom. Azon részeket, melyeket más szerzők munkájából vettem át, egyértelműen megjelöltem, és az irodalomjegyzékben szerepeltettem.

Ha a fenti nyilatkozattal valótlan állítottnak tudomásul veszem, hogy a záróvizsga-bizottság a záróvizsgából kizár és a záróvizsgát csak új dolgozat készítése után tehetek.

A leadott dolgozat, mely PDF dokumentum, szerkesztését nem, megtekintését és nyomtatását engedélyezem.

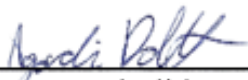
Tudomásul veszem, hogy az általam készített dolgozatra, mint szellemi alkotás felhasználására, hasznosítására a Magyar Agrár- és Élettudományi Egyetem mindenkori szellemi tulajdon-kezelési szabályzatában megfogalmazottak érvényesek.

Tudomásul veszem, hogy dolgozatom elektronikus változata feltöltésre kerül a Magyar Agrár- és Élettudományi Egyetem MATER Hallgatói Dolgozatok repozitóriumba. Tudomásul veszem, hogy a megvédett és

- nem titkosított dolgozat a védést követően
- titkosításra engedélyezett dolgozat a benyújtásától számított 5 év eltelte után

nyilvánosan elérhető és kereshető lesz az Egyetem MATER Hallgatói Dolgozatok repozitóriumban.

Kelt: 2024. év 11. hó 01. nap


Hallgató aláírása

10. Mellékletek

1. melléklet: A kiértékelő elektronika programja

```
#include <SPI.h>                //Library for using SPI Communication
#include <mcp2515.h>            //Library for using CAN Communication
                                //https://github.com/autowp/arduino-mcp2515/
#include <math.h>               //Library for mathematical references

struct can_frame canMsg;

MCP2515 mcp2515(9);            //Set the CAN communication wire to the digital pin 9

//Set constant value to calculate air temperature
const int R0 = 10000;          //Reference resistance [ohm]
const int Coeff = 3380;        //Reference coefficient [1/°C]
const double T0 = 298.15;      //Reference temperature [°C]
const double K_C = 273.15;     //Change number from [K] to [°C]
const int R2 = 3000;          //Voltage divider secondary resistance [ohm]
const float delta = 0.5;       //Thermistor dissipation coefficient [°C/mW]

//Set constant value to calculate current
const double resADC = 0.004883; //input voltage resolution (5/1024) [V]
const double zeroP = 2.5;       //mid-point voltage [V]
const double scale = 0.066;     //scale factor (30A) [mV/A]

const double rate = 5.6296;     //Rate of the two resistance in voltage measurement ((20
000 + 4 320)/4 320)

//=====

void setup()
{
    Serial.begin(115200);        //Begins Serial Communication at 115200 baudrate
    SPI.begin();                //Begins SPI communication

    mcp2515.reset();
    mcp2515.setBaudrate(CAN_1000KBPS, MCP_8MHZ); //Sets CAN at speed 500KBPS and Clock 8MHz
    mcp2515.setNormalMode();    //Sets CAN at normal mode

    Serial.print("Initialization...");

    delay(2000);

    while (mcp2515.readMessage(&canMsg) != MCP2515::ERROR_OK) //To receive data (Poll Read)
    {
        Serial.println("Connection: ERROR");
        delay(100);
    }

    Serial.println("Connection: OK");
    delay(1000);
}

//=====

void loop()
{
    if (mcp2515.readMessage(&canMsg) == MCP2515::ERROR_OK)
    {
        int WaterTemp = canMsg.data[0]; //Store the sent water temperature data
        int OilTemp = canMsg.data[1];   //Store the sent oil temperature data
        int AirTemp = canMsg.data[2];    //Store the sent air temperature data
        int EngineRPM = canMsg.data[3];  //Store the sent engine rotation duration data
        int TLTRPM = canMsg.data[4];     //Store the sent TLT axle rotation duration data
        int CurrentV = canMsg.data[5];   //Store the sent system current usage data
        int Voltage = canMsg.data[6];     //Store the sent generator output voltage data
        int Knock = canMsg.data[7];      //Store the sent engine knocking data

        //Calculate the air temperature value in [°C]
        double R1 = (((R2*12)/(AirTemp*resADC))-R2);
        double PT = ((pow(12,2)*1000)/R1);
        double TD = (delta*PT);
        double AirTempC = (((1/((1/T0)+(1/Coeff)*log(R1/R0)))-K_C)-TD);
    }
}
```

```

//Calculate the current value in [A]
double Vout = (resADC*CurrentV);
double Current = ((Vout-zeroP)/scale);

//Calculate the engine rotation speed in [1/min]
unsigned long RPM_E = ((60000000ul / EngineRPM)*20);

//Calculate the TLT axle rotation speed in [1/min]
unsigned long RPM_TLT = ((60000000ul / TLTRPM)*8);

//calculate the voltage value from bits to [V]
double V = (resADC*Voltage*rate);

//Calculating the output voltage of the knock sensor in [mV]
double K = ((resADC*Knock) / 1000);

Serial.println("Water temperature: ");
Serial.print(" °C");
Serial.println("Oil temperature: ");
Serial.print(" °C");
Serial.println("Air temperature: ");
Serial.print(" °C");
Serial.println("Engine RPM: ");
Serial.print(" 1/min");
Serial.println("TLT axle RPM: ");
Serial.print(" 1/min");
Serial.println("System Current usage: ");
Serial.print(" A");
Serial.println("Generator output voltage: ");
Serial.print(" V");
Serial.println("Knock sensor's output voltage: ");
Serial.print(" mV");

//delay(1);
}
else{
    Serial.println("Connection: LOST");
    delay(1000);
}
}

//=====

```

2. melléklet: A mérő elektronika programja

```

#include <SPI.h> //Library for using SPI Communication
#include <mcp2515.h> //Library for using CAN Communication
(https://github.com/autowp/arduino-mcp2515/)
#include <OneWire.h> //Library for using one wire communication
#include <DallasTemperature.h> //Library for using one wire temperature sensor

struct can_frame canMsg;

MCP2515 mcp2515 (9); //Set the CAN communication wire to the digital pin 9

#define DS_TEMP 30 //Water and oil temperature sensor connected to digital pin 30
int var_3 = A8; //Intake air temperature connected to analog pin 8
byte var_4 = 20; //Engine rotation speed sensor connected to digital pin 20
byte var_5 = 21; //TLT axle rotation speed sensor connected to digital pin 21
int var_6 = A9; //Current sensor connected to analog pin 9
int var_7 = A10; //Voltage "sensor" connected to analog pin 10
int var_8 = A11; //Knock sensor connected to analog pin 11

OneWire oneWire (DS_TEMP); //Setup a oneWire instance to communicate whit any
OneWire devices

DallasTemperature sensors(&oneWire); //Pass oneWire reference to Dallas temperature

int deviceCount = 0; //Variable to the number of DS18B20 sensors

//Variable to store the value of water and oil temperature

```

```

float temp_W;
float temp_O;

int t_A = 0;           //Set variable to store the intake air temperature sensor value
int C = 0;             //Set variable to store the current sensor value
int V = 0;             //Set variable to store the voltage value
int K = 0;             //Set variable to store the knock sensor value

//Set variables for engine rotation speed counting
unsigned long E_revMicros;
unsigned long E_prevRevMicros;
unsigned long RPM_E;
unsigned long E_revDuration;
volatile unsigned long E_isrMicros;
volatile unsigned long E_isrCount;
volatile bool E_newIsrMicros = false;

//Set variables for TLT axle rotation speed counting
unsigned long TLT_revMicros;
unsigned long TLT_prevRevMicros;
unsigned long RPM_TLT;
unsigned long TLT_revDuration;
volatile unsigned long TLT_isrMicros;
volatile unsigned long TLT_isrCount;
volatile bool TLT_newIsrMicros = false;

//=====

void setup()
{
    while (!Serial);
    Serial.begin(115200);           //Begins Serial Communication at 115200 baudrate
    SPI.begin();                   //Begins SPI communication
    sensors.begin();               //Begins the BS18B20 temperature sensors' communication

    mcp2515.reset();
    mcp2515.setBitrate(CAN_1000KBPS, MCP_8MHZ); //Sets CAN at speed 1000KBPS and Clock 8MHz
    mcp2515.setNormalMode();        //Sets CAN at normal mode

    deviceCount = sensors.getDeviceCount(); //Returns the number of BS18B20 sensor

    E_isrCount = 0;
    TLT_isrCount = 0;
    attachInterrupt(digitalPinToInterrupt(var_4), revDetectorE, RISING); //Attach
interrupt function to the engine rotation speed counting
    attachInterrupt(digitalPinToInterrupt(var_5), revDetectorTLT, RISING); //Attach
interrupt function to the TLT axle rotation speed counting
}

//=====

void loop()
{
    getDataE();                   //Receive data from engine rotation measurement
    getDataTLT();                 //Receive data from TLT axle rotation measurement
    TempSens();                   //Receive data from water and oil temperature measurement
    t_A = analogRead(var_3);      //Read data from air temperature measurement
    C = analogRead(var_6);        //Read data from current measurement
    V = analogRead(var_7);        //Read data from voltage measurement
    K = analogRead(var_8);        //Read data from engine knock measurement

    Message();                    //Call the message sending function
}

//=====

void getDataE()
{
    if (E_newIsrMicros == true)
    {
        E_prevRevMicros = E_revMicros; //Store the previous value
        noInterrupts();
        E_revMicros = E_isrMicros;
        RPM_E = E_isrCount;
        E_newIsrMicros = false;
        interrupts();
    }
}

```

```

        E_revDuration = E_revMicros - E_prevRevMicros;           //Gives the time between two
        pulses in microseconds
    }
}

//=====

void getDataTLT()
{
    if (TLT_newIsrMicros == true)
    {
        TLT_prevRevMicros = TLT_revMicros;           //Store the previous value
        noInterrupts();
        TLT_revMicros = TLT_isrMicros;
        RPM_TLT = TLT_isrCount;
        TLT_newIsrMicros = false;
        interrupts();
        TLT_revDuration = TLT_revMicros - TLT_prevRevMicros;     //Gives the time between two
        pulses in microseconds
    }
}

//=====

void revDetectorE()
{
    E_isrMicros = micros();
    E_isrCount ++;
    E_newIsrMicros = true;
}

//=====

void revDetectorTLT()
{
    TLT_isrMicros = micros();
    TLT_isrCount ++;
    TLT_newIsrMicros = true;
}

//=====

void TempSens()
{
    for (int i = 0; i < deviceCount; i++)
    {
        if(i == 0)
        {
            temp_W = sensors.getTempCByIndex(0);           //Store the water temperature sensor value
        }
        else
        {
            temp_O = sensors.getTempCByIndex(1);           //Store the oil temperature sensor value
        }
    }
}

//=====

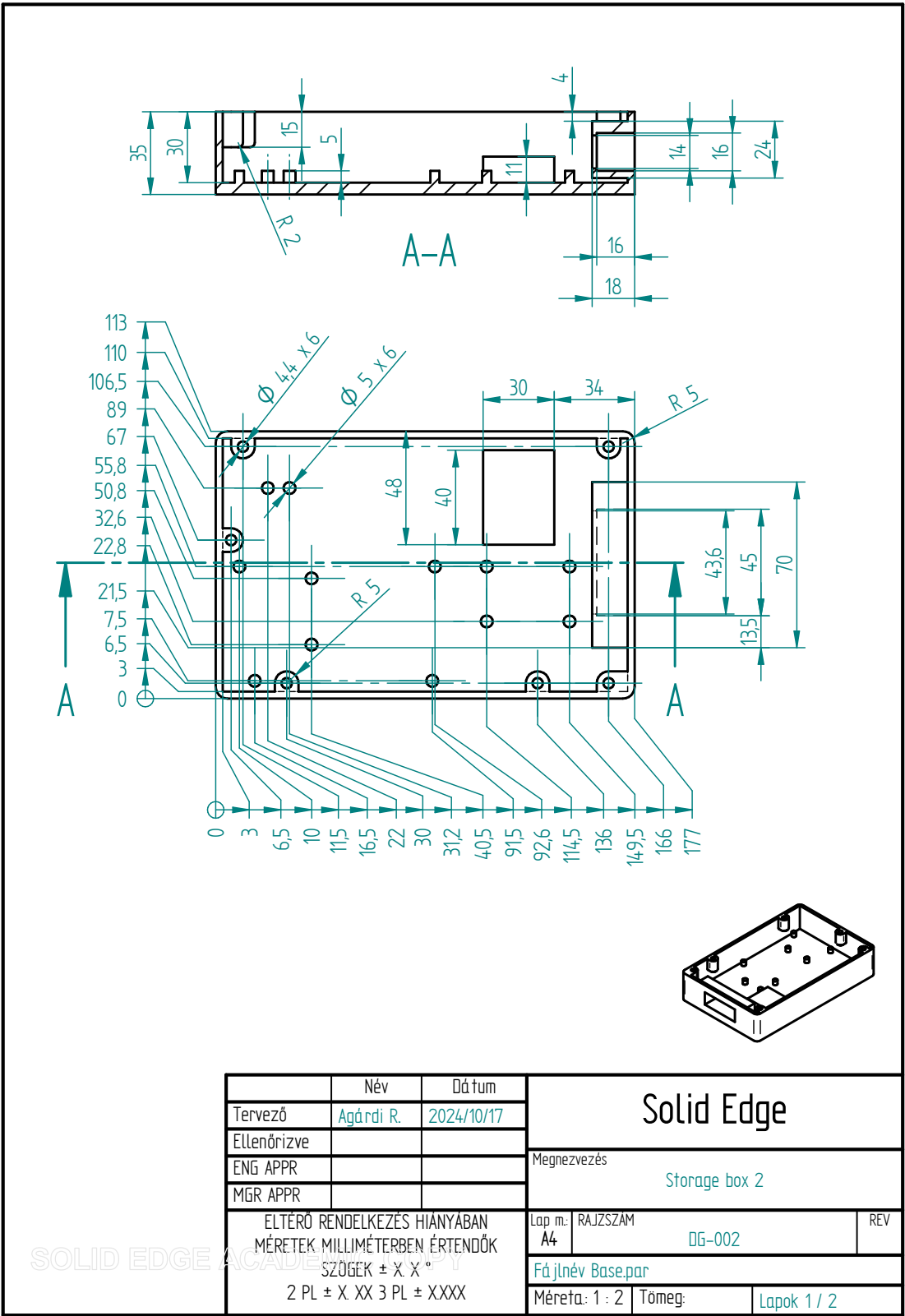
void Message()
{
    canMsg.can_id = 0x036;           //CAN id as 0x036
    canMsg.can_dlc = 8;              //CAN data length as 8
    canMsg.data[0] = temp_W;         //Update water temperature value in [0]
    canMsg.data[1] = temp_O;         //Update oil temperature value in [1]
    canMsg.data[2] = t_A;            //Update air temperature value in [2]
    canMsg.data[3] = E_revDuration;  //Update engine rotation value in [3]
    canMsg.data[4] = TLT_revDuration; //Update TLT rotation value in [4]
    canMsg.data[5] = C;              //Update current value in [5]
    canMsg.data[6] = V;              //Update voltage value in [6]
    canMsg.data[7] = K;              //Update knocking value in [7]

    mcp2515.sendMessage(&canMsg); //Sends the CAN message
}

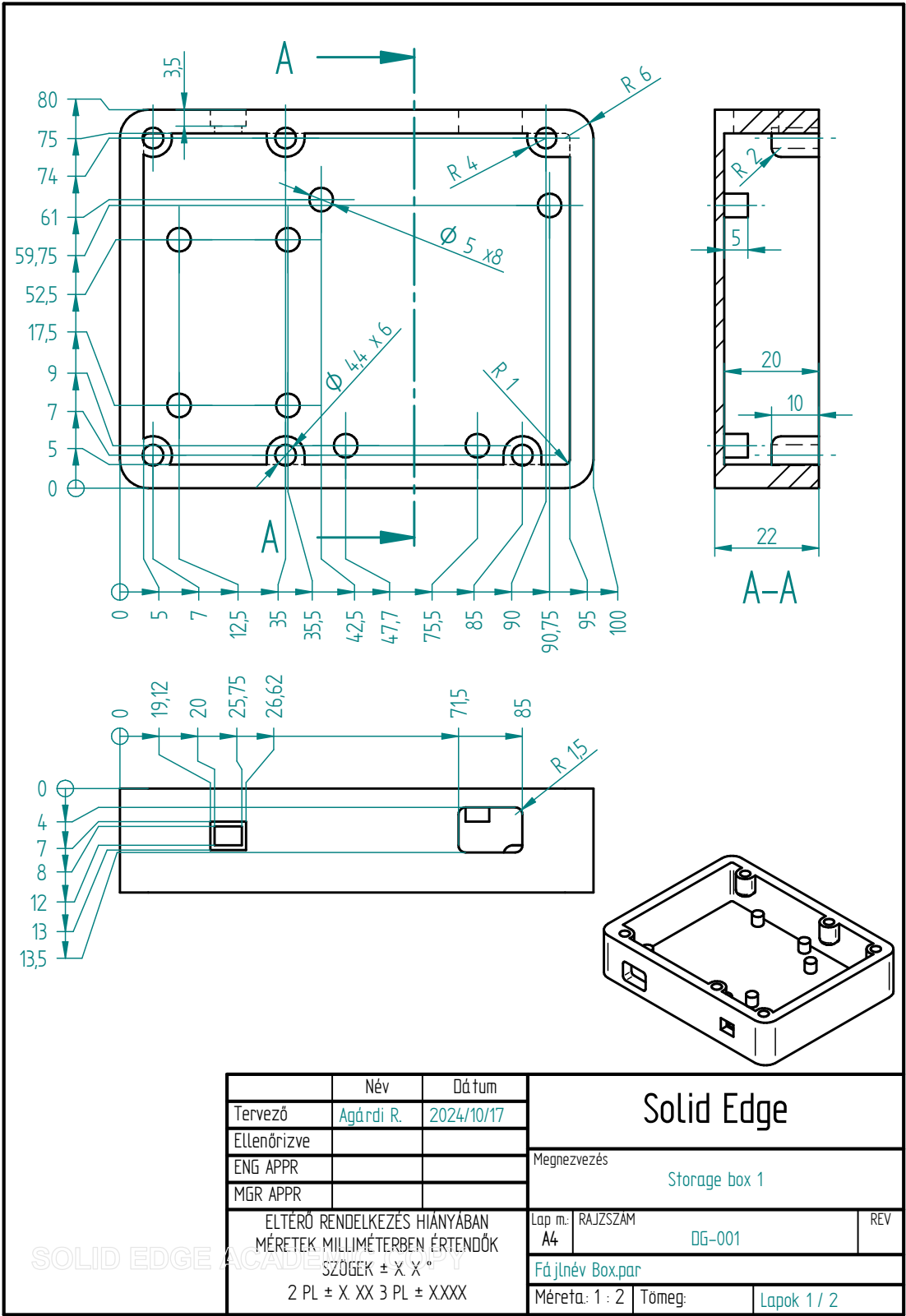
//=====

```

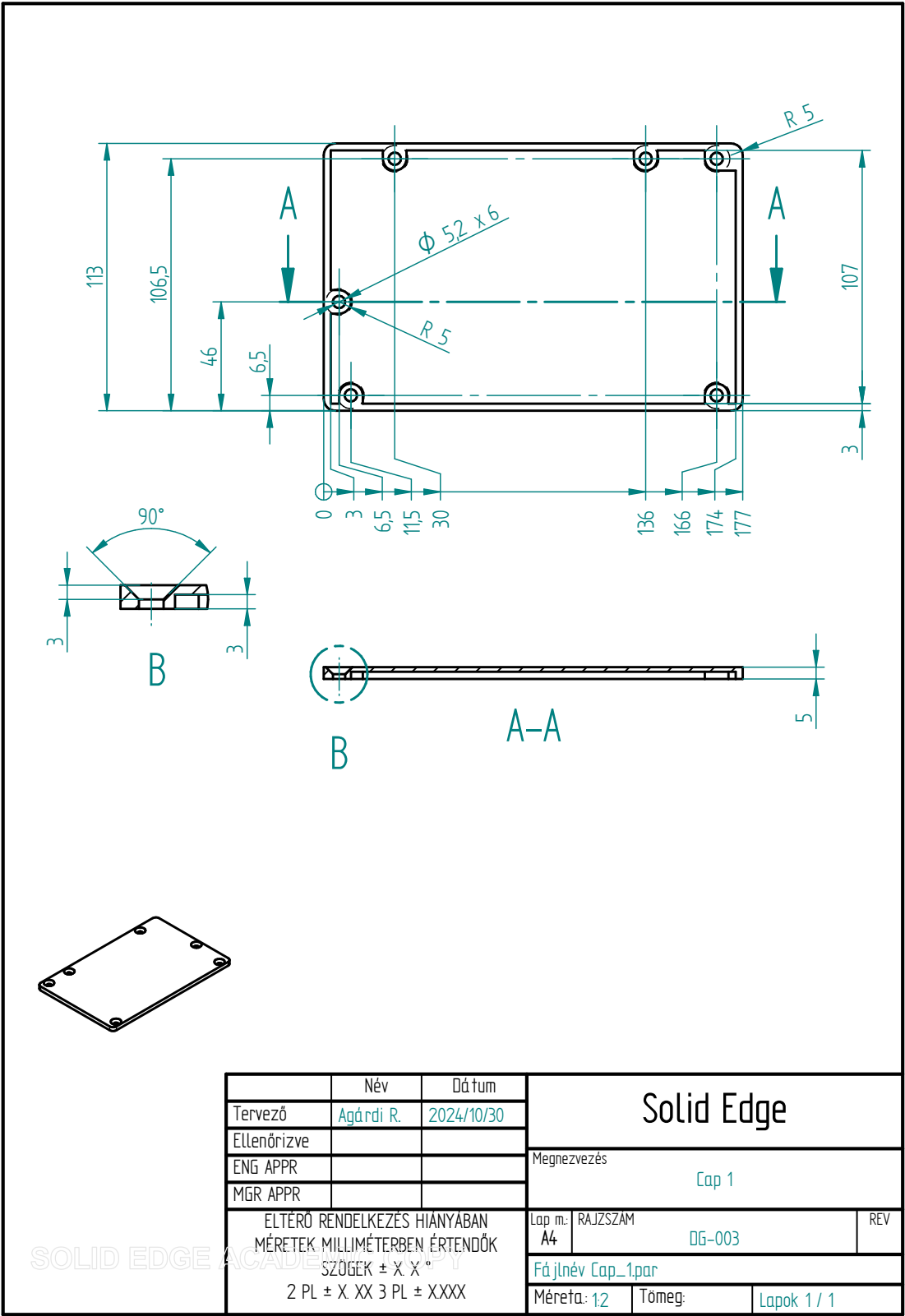
3. melléklet: A mérő elektronika tároló dobozának rajza



4. melléklet: A kiértékelő elektronika tároló dobozának rajza



5. melléklet: A mérő elektronika doboz fedelének rajza



6. melléklet: A kiértékelő elektronika doboz fedelének rajza

