



ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

KANDÓ KÁLMÁN VILLAMOSMÉRNÖKI KAR
HÍRADÁSTECHNIKA INTÉZET



TDK DOLGOZAT

OKOS AKKUMULÁTOR FELÜGYELŐ RENDSZER JÁRMŰVEKBE

**OE-KVK-HTI
2022**

**Dragschitz Xavér Elek
F2JF04**

TARTALOMJEGYZÉK

1	Bevezetés	2
2	A feladat leírása	3
2.1	Valós felhasználási körülmények	3
2.2	Mobilkommunikációs lehetőségek elemzése	3
3	Tervezés és összeszerelés.....	7
3.1	Rádiókommunikációs modul(ok) kiválasztása	7
3.2	Alkatrészek kiválasztása.....	7
3.3	Áramkör tervezése	8
3.4	Kompatibilitás kivitelezése a chip hiány tekintetében.....	11
3.5	NYÁK tervezése	13
3.6	A kész NYÁK átvizsgálása	16
3.7	Forrasztás és hiba keresés.....	17
4	Programozás.....	19
4.1	Az elvárt funkcionalitások és feladatok.....	19
4.2	A program blokkvázlatosan.....	20
5	Tesztelés.....	21
5.1	Tesztelés fázisai	21
5.2	Mobilhálózati mérés teszt körülményei.....	21
5.3	Mobilhálózati mérés eredmények.....	22
5.4	Okosotthon környezetében végzett vizsgálatok	26
6	Összegzés	27
7	Irodalomjegyzék	28
8	Ábrajegyzék	29

1 BEVEZETÉS

A projekt, amit kitűztem magam elé, egy olyan eszköz tervezését és tesztelését foglalja magában, ami megkönnyíti az akkumulátorokkal kapcsolatos problémák kezelését (itt most arra gondolok, hogy fel van-e töltve az adott jármű akkumulátora). Ezt az információt, egy mikrokontroller segítségével egy okos otthonba integrálható rendszerré szeretném kialakítani, hogy akár a telefonunkról is láthassuk, ha töltésre van szüksége a járműnek. Természetesen ez csak olyan esetekben releváns amikor a házunk közelében helyezkednek el a járművek (autó a kerítés előtt vagy a garázsban, elektromos bicikli a tárolóban és hasonló). Az újabb autókban főleg a hibridekben és a tisztán elektromosokban ez a funkció már alapfelszereltségnek számít, így autók terén főleg a régebbi évjáratokban előnyös egy ilyen eszköz használata. A funkcionalitás kiterjesztése miatt, az eszközbe integrálásra kerül még egy hőmérséklet és páratartalom mérő szenzor, így az adott jármű külső páratartalom és hőmérő állomásként is üzemelhet igény esetén. Abban az esetben, ha elhagyjuk az otthoni WiFi hálózat hatósugarát és továbbra is szeretnénk használni ezt az eszközt, akkor szükségünk lesz egy másodlagos kommunikációs módszerre is. Sokféle opció közül választhatunk LoRa, GSM, NB-IoT (Narrow Band Internet of Things), CAT-M esetleg 5G-t is integrálhatunk igény esetén. A választás során viszont figyelembe kell vennünk, az adott technológiához tartozó hálózati lefedettséget, energia takarékossgot és a jelenleg is erősen fennálló chip-hiányt, ami korlátozza az adott modulok beszerezhetőségét. Ezek a modulok a legtöbb esetben GNSS (Global navigation satellite system)-et is tartalmaznak, így egy plusz funkció is bekerül az eszközbe, ezzel nyomon követhetjük a járművet, ha esetleg ellopnák azt. Természetesen ezek a modulok és a perifériák sok áramot fogyaszthatnak, ha nem figyelünk erre a tervezéskor, továbbá igen kontraproduktív lenne, ha éppen a felügyelt akkumulátorról működne folyamatosan. Ezért az eszköznek rendelkeznie kell egy saját akkumulátorral, ami kellően nagy kapacitású és a töltését a mikrokontroller szabályozza, hogy csak egy bizonyos töltöttség alatt kapcsoljon be a töltőáramkör, ezzel is csökkentve a készenléti áramfogyasztást. A tervezés és az összeszerelés után fontos a tesztelés (mérés), ehhez viszont szükségesek a megfelelő eszközök és berendezések, aminek egy részével magam is rendelkezem.

A mobilhálózati és az ahhoz kapcsolódó eszközökkel történő méréseket a Magyar Telekom TDI (Technology Delivery International) SSC laboratóriumában végeztem el, ahol támogatták a munkámat ebben a fázisban.

2 A FELADAT LEÍRÁSA

2.1 Valós felhasználási körülmények

Maga az alapötlet a koronavírus által kialakult helyzet szülte: a pandémia első pár hónapjában (2020 február-június) sokan nem használták az autójukat a kötelező home office-ok miatt és sokan az első „hagyományos” munkanapon szembesültek azzal, hogy lemerült az autójuk akkumulátora. De nem szükséges, hogy ennyire szélsőséges helyzet alakuljon ki ahhoz, hogy lemerüljön az autó, elég, ha felkapcsolva hagyjuk a rádiót, belső világítást vagy a tompított fényszórót és másnapra máris megalapoztunk egy kellemetlen helyzetet. Ugyan ilyen kellemetlen helyzet érhet minket, ha egy hobbi járművet szeretnénk használni és már rég lemerült az akkumulátora.

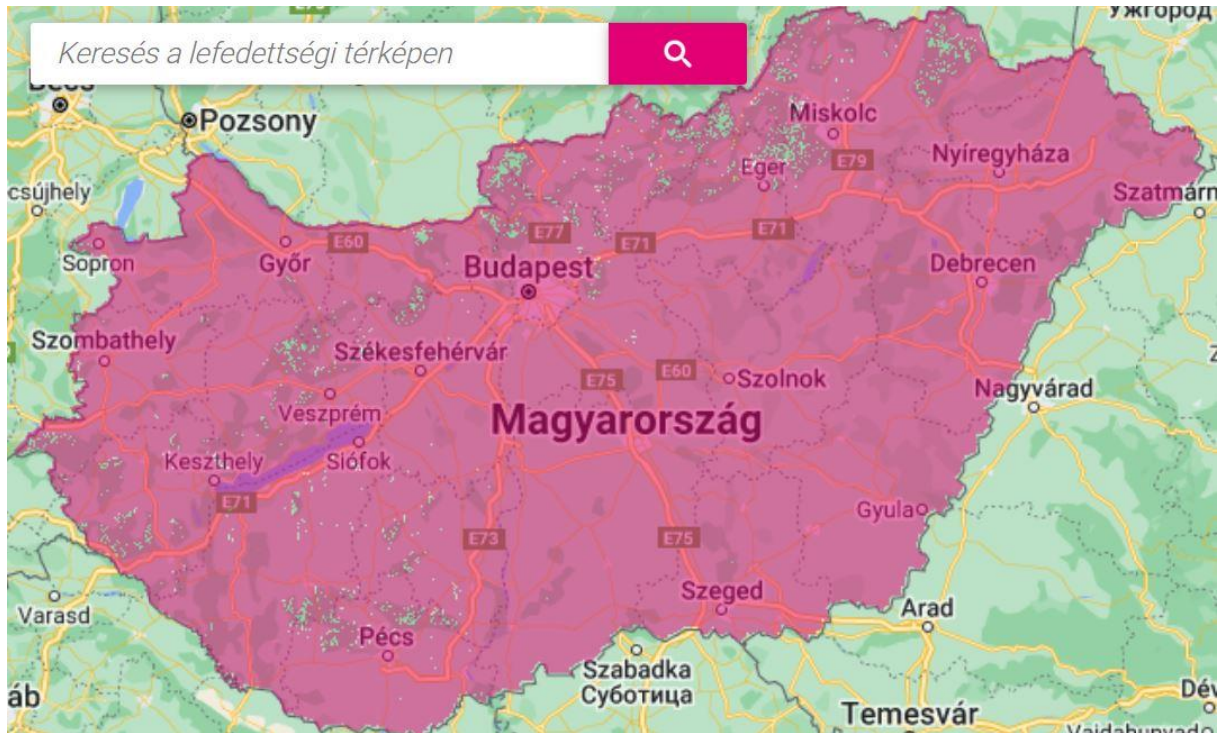
2.2 Mobilkommunikációs lehetőségek elemzése

Ma Magyarországon sokféle mobilkommunikációs technológiával találkozhatunk, amiknek más-más a lefedettsége és a képességei is, így válogathatunk közülük bizonyos keretek közt. Jelenleg kettő releváns generáció van, ami közül választanunk kell egyet. Az egyik a 2G azaz GSM (2.5G GPRS) a másik meg a 4G, amin belül több, különféle opcióból választhatunk.

A NB-IoT (Narrow Band Internet of Things) aminek a lényege, hogy egy keskeny sávot kap az LTE sáv szélességből (nagyjából 180-200kHz) és alacsony adatátviteli sebességgel kommunikál rajta (pár száz kbit/s-os nagyságrendben), közben hosszú akkumulátor időt garantál, viszont a kommunikáció üzenet alapú, így relatíve a késleltetése. A másik a már említett Cat-M vagy másnéven LTE-M. Itt egy nagyobb sáv szélességet kap a kommunikáció (nagyjából 1-1.4MHz) és az adatátviteli sebesség jelentősen magasabb (1 Mbit/s környékén) és szintén hosszú akkumulátor időt garantál, ellentétben az NB-IoT-val itt a kommunikáció IP alapú és jelentősen kisebb a késleltetés. Jelenleg az utóbbi kettő közül mindkettő megfelelne a célnak mivel a késleltetés nem fontos szempont a mi esetünkben, viszont figyelembe kell venni a GSM-ben lévő lehetőségeket is. Az utóbbi kijelentés furcsán hathat, mivel a GSM-et félig-meddig már le szeretnék kapcsolni a szolgáltatók, de robusztussága miatt nehezen tudják kiváltani, illetve az összes telefon kompatibilis vele, mint alap kommunikációs technológia. Olyan régiókban, ahol nem elérhető a VoLTE szolgáltatás, ott telefonunk az adatot az LTE hálózaton keresztül továbbítja, viszont a hívások GSM-en keresztül történnek (a két technológia között hallható különbség van hívás közben). Mivel az eszköz, amit tervezni szeretnénk, csak

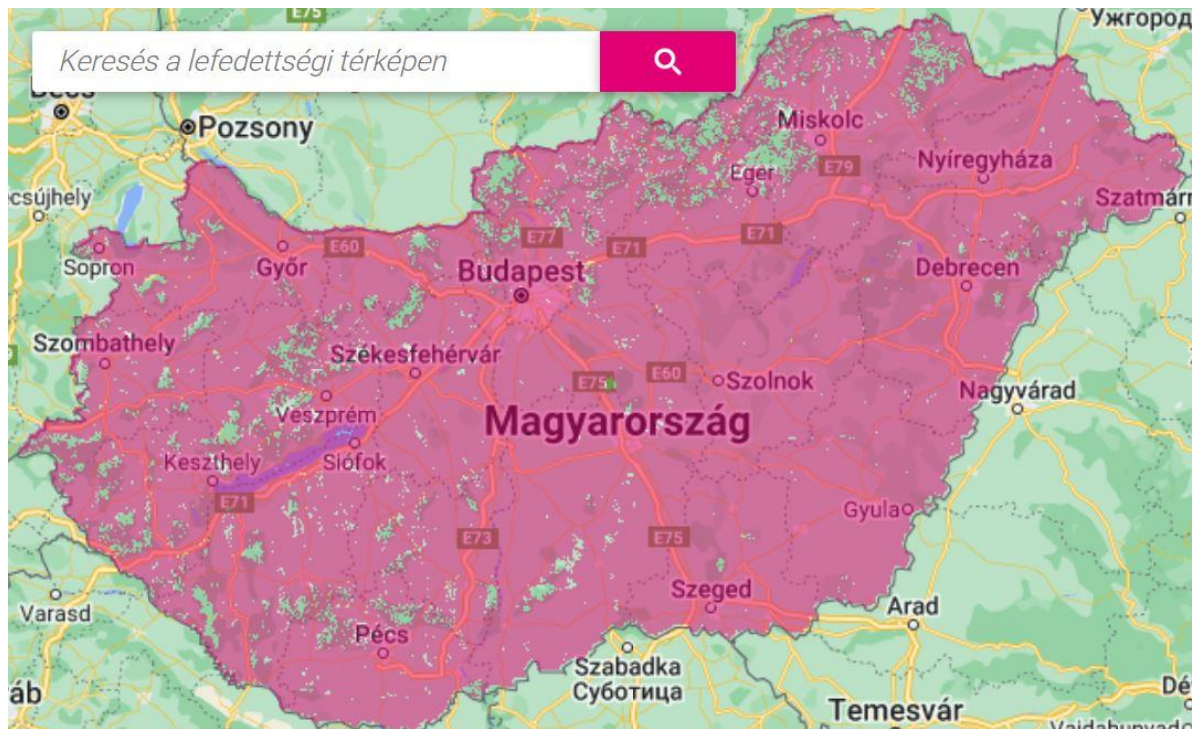
esetenként használja a mobilhálózatot, így a minimális adattovábbításra a felsorolt technológiák mind megfelelnek.

A következő szempont a technológia kiválasztása során a lefedettség és a jelterjedés vizsgálata. Magyarországon jelenleg a legnagyobb lefedettsége a GSM-nek van, ami közel 99%-os. A Telekom lefedettségét a következő ábrán láthatjuk [1].



1. ábra Magyar Telekom GSM lefedettsége

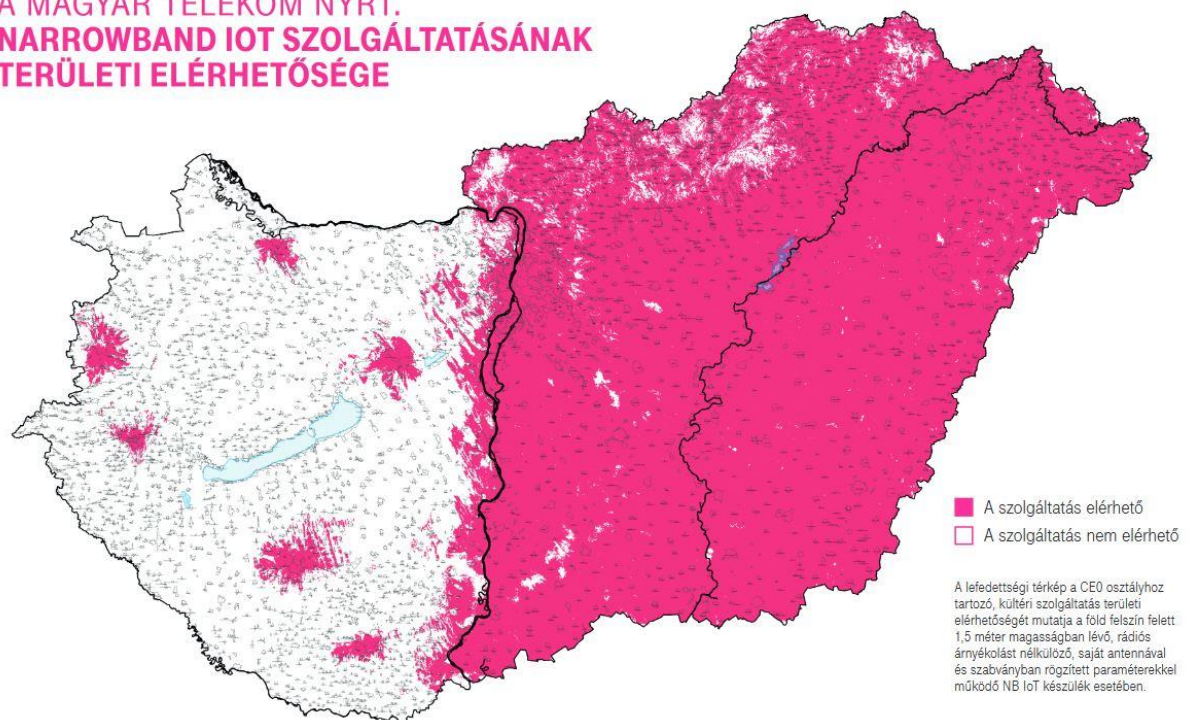
Látható, hogy néhány erdős, völgyes régiót eltekintve elfogadható a lefedettség, így ezzel a modemmel felszerelt eszköz szinte mindenhol használható. Az LTE lefedettség szintén kiváló Magyarországon, közel 98-99%-os lefedettséggel. Ezt az ábrát szintén megtekinthetjük a Telekom honlapján (ahol a szolgáltatásuk elérhető) [1].



2. ábra Magyar Telekom LTE lefedettsége

Itt azért látható, hogy nagyobb vakfoltok vannak a térképen, így azokon a részekén problémába ütközhet a működés, vagy egyáltalán nem is működne a modem. Természetesen, ez a sima LTE lefedettség, maga az NB-IoT szolgáltatást elérhetőségét egy külön térképen szokták jelölni, amit a következő ábrán láthatunk [2].

A MAGYAR TELEKOM NYRT. NARROWBAND IOT SZOLGÁLTATÁSÁNAK TERÜLETI ELÉRHETŐSÉGE



3. ábra Magyar Telekom NB-IoT szolgáltatásának lefedettsége

Látható, hogy ez a szolgáltatás olyan 50-60%-os lefedettséggel rendelkezik, így ahhoz, hogy az országon belül mindenhol tudjuk használni az eszközt vagy több SIM-es megoldásban kellene gondolkoznunk, hogy egyszerre több szolgáltató hálózatát is kellene használnunk, vagy olyan előfizetéssel, ami engedi a szolgáltatók közötti átjárást.

A jelterjedés szempontjából az a határ, ahol a modem eldobja a kapcsolatot általában mínusz 100-120dBm környékén jelentkezik. A GSM nagyjából -105-109dBm-nél kezd el megbízhatatlan lenni, míg az LTE alapú modemek (Cat-M, NB-IoT) egészen -110-120dBm-ig többnyire megbízható kapcsolatot biztosítanak. Nem véletlen, hogy ezeket akár pincékben és vízaknákban is megbízhatóan lehet alkalmazni különféle szenzorok modemjeként. Ezeket a szempontokat figyelembe véve a legideálisabb választás a Cat-M lenne az esetünkben főleg az IP alapú kommunikáció miatt, utána az NB-IoT és ha ezek nem beszerezhetőek akkor meg a hagyományos GPRS képességű GSM modem.

Az elemzés során nem esett szó az 5G-s lehetőségekről, illetve a 3G-ről, ami nem véletlen. A 3G hálózatot a Telekom 2022-ben leállította és a többi szolgáltató is követte ezt a példát, így ez a szolgáltatás megszűnt Magyarországon. Az 5G hálózat telepítése meg jelenleg is folyamatban van és csak limitált helyeken elérhető. Jelenleg ezekért a modemekért a különféle alkatrész árusító honlapokon (és gyártóknál) prémium árat kell fizetni, így a projekt jelentősen megdrágulna és egy ideig korlátozott lenne a használhatósága is. Természetesen az újabb modem chip-ek visszafele kompatibilisek sok esetben, viszont továbbra is fennállna az, hogy egy 5G-s modem esetén kifizetjük a prémium felárat és nem használjuk ki ezt a képességet.

3 TERVEZÉS ÉS ÖSSZESZERELÉS

3.1 Rádiókommunikációs modul(ok) kiválasztása

A modemek terén sok gyártó különböző termékei közül választhatunk, az elsődleges választásom a SIM7070-re [3] esett, mivel ez a modul támogatja a Cat-M, NB-IoT és a GSM technológiákat egy chip-ben és olcsón beszerezhető, illetve, könnyen alkalmazható. Ebben a modulban nincs GNSS modul, így azt külön kell elhelyezzük az áramkörünkben. Erre szintén egy megfizethető és beszerezhető modult választottam, ami egy Quectel L80-as [4]. A tervezés során jelentős segítséget nyújt, hogy beépített antennával rendelkezik, így nem kell külön antenna és RF csatlakozó a GNSS-nek. Az okosotthon-hoz való csatlakozásra szükség van egy olyan chip-re, ami WiFi képességekkel rendelkezik, erre a legoptimálisabb választás az ESP32-S2 [5] mikrokontroller, ami az egész projektünk vezérlése mellett WiFi és Bluetooth-os rádiós interfésszel is rendelkezik

A továbbra is fennálló chip-hiány miatt, a kiválasztott SIM7070 gyorsabban elfogyott (a limitált mennyiség miatt) mint ahogy be tudtam volna fejezni a tervezést. Ez a probléma második újra tervezés után is előfordult, így a harmadik verzióban egy olcsó GSM modult választottam, ami egy Ai-Thinker A9-es [6]. Ennek van egy kompatibilis verziója az A9G, ami GNSS modullal is el van látva viszont nincs belső antennája, ebben az esetben nincs szükség a Quectel L80-ra viszont szükség van egy RF csatlakozóra, egy külső antennára és a megfelelő impedanciájú microstrip-re a panelen. A NB-IoT képességről továbbra sem mondtam le és az Ai-Thinker kínálatában van olyan modul, ami minimális átkötéssel, de láb kompatibilis az A9/A9G-vel, ennek a megvalósításnak a kivitelezését a továbbiakban jobban ki fogom fejteni a 3.4-es fejezetben.

3.2 Alkatrészek kiválasztása

Az alkatrészek kiválasztása során főleg arra törekedtem, hogy a készenléti áram a legkisebb legyen, így jelentősen megnőhet a beépített akkumulátor üzemideje töltés nélkül. A mikrokontroller a már előbb is említett ESP32-S2, ami egy WROOVER-I [7] nevezetű részben kiépített modulban helyezkedik el beépített NYÁK (nyomtatott áramkör) antennával, illetve szükség esetén külső antennával is használható. Ennek a mikrokontrollernek az előnye, hogy USB-n keresztül is programozhatjuk nem csak a megadott UART [8] portokon, ezért elhagyhatjuk a programozó IC-t az áramkörtől, ezzel is csökkentve a fogyasztást és a

költségeket. USB-C típusú csatlakozókat választottam mind a modem port-jára mind az ESP port-jára, így megfelel a modern elvárásoknak. Az akkumulátor feszültsége jellemzően 4.2V feltöltve, de az eszközeink főleg 3.3V-os feszültséggel működnek így egy feszültség szabályzó IC-re is szükségünk lesz, kivéve a modemnek, ami 3.5-4.2V-os tartományban működik. Az akkumulátor töltésére egy szabályozható (EN, azaz enable lábbal, ki-be kapcsolható) egy cellás töltő IC a legjobb választás. A többi alkatrésznek lehetőleg a legkisebbnek kell lennie, amivel még kényelmesen (és otthon is) dolgozhatunk. Az én esetben azt jelenti, hogy a kondenzátorok és ellenállások SMD 0603-as tokozásúak lesznek. Az áramkörben el lesznek helyezve LED-ek is, mivel ezek növelik a fogyasztást esetenként így ezek nem lesznek beforrasztva, csak az első prototípuson (kivéve az ESP32-re direktbe rákötött LED, amit akár hibakód kijelzésre is használhatunk). A kompakt elrendezés szempontjából érdemes nem rendes méretű SIM kártya foglalatot elhelyezni, hanem micro vagy nano SIM foglalatokat, mint amilyeneket manapság az okostelefonokban is használunk.

3.3 Áramkör tervezése

Az áramkör tervezéséhez az EasyEDA nevű online tervező programot használtam [9], ami integrált alkatrész könyvtárral is rendelkezik, így azonnal nyomon követhető, hogy az adott alkatrészből mennyi van éppen az LCSC [10] kínálatában. A tervezés során előnyös, ha blokkokra osztva tervezünk, így átláthatóbb és a későbbiekben más projektekhez is felhasználható egy-egy modul újra tervezés nélkül. A megtervezett áramkör az előbb felsorolt követelmények szerint a következő ábrán látható.

4. ábra A megtervezett áramkör kapcsolásrajza

Az ESP32 vezérli a töltést egy FET-en keresztül, mivel a chip-hiány következtében elfogytak azok a töltő IC-k, amik enable (EN) lábbal rendelkeznek és kellően nagy bemeneti feszültség tolerancia határral rendelkeznek. Ezért előtte található egy feszültség szabályzó, ami a CBATT-on keresztül érkező feszültséget 5V-os kimeneti feszültséggé alakítja és egészen 20V-ig működik. A jármű akkumulátorát a CBATT csatlakozóra kell rákapcsolni, ami után találhatunk egy hagyományos 5*20mm-es üvegbiztosíték tartót (hogy a felhasználó is ki tudja cserélni, ha szükség van rá) és egy egyszerű FET-es polaritásvédelmet, ami sokkal kevesebb hőt termel, mint egy egyszerű diódás védelem. A SIM kártyatartónak egy SMD micro SIM-et választottam, így egy átalakítóval a nano SIM is használható. Maga a SIM modul áramköre egy referencia áramkör, ami a legtöbb modem datasheet-ben megtalálható úgy, mint a GSM power regulator. A modem magáról az akkumulátorról működik, így csak a regulator modul szükséges az akkumulátor után, hogy kellően stabil legyen az ellátása, viszont az ESP32-nek és az SHT31 hőmérséklet és páratartalom szenzornak 3.3V-ra van szüksége. Ezeknek az IC-knek egy külön feszültség szabályzója van amire rá van kötve az akkumulátor egy egyszerű diódás védelemmel, hogy az USB 5V-os feszültsége ne okozzon kárt az akkumulátorban. Az ESP32-re rá van kötve az akkumulátor egy feszültség osztón keresztül így azt is felügyelni tudjuk, illetve ebből a firmware meg tudja határozni, hogy mikor kell a töltést bekapcsolni.

Az ESP32 az adatlap szerint 2db 13bit-es felbontású SAR ADC-vel rendelkezik (Successive-approximation analog-digital converter). Felbontás szempontjából ez annyit jelent, hogy $2^{13}=8192$, a maximális bementi feszültség meg 3.3V lehet az IO portokon, ebből adódik, hogy bizonytalanság a következőképpen alakul:

$$\frac{3.3V}{8192} = 4.028 * 10^{-4}V$$

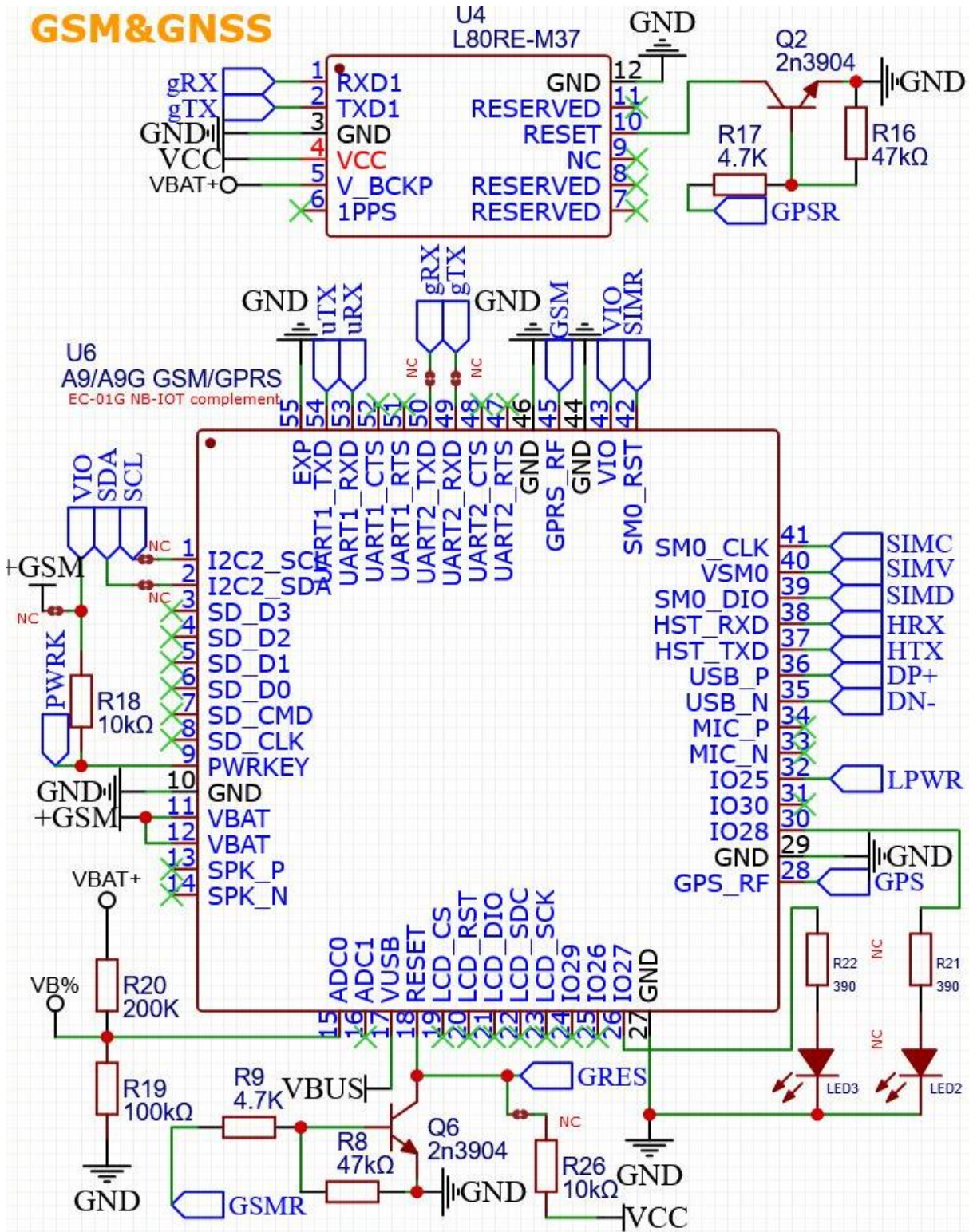
Azaz $\pm 0.4028mV$ bizonytalansággal számolhatunk mind az akkumulátor feszültségének megállapításakor (VB% feszültség busz), mind a jármű akkumulátor feszültsége esetén (VBOUT feszültség busz). Természetesen ehhez még hozzáadódik a feszültség osztóban található ellenállások valós értéke, amik $\pm 1\%$ -os eltéréssel rendelkeznek a névleges ellenálláshoz képest, így ezeket a beforrasztás után le kell mérnünk és azzal kell számolnunk a programon belül.

Ezen kívül az áramkörben található még 3db EXT csatlakozó, hogy megkönnyítse a programozást és az esetleges bővítést. Az egyik egy I2C port, ami rá van kötve az SHT31-re [11] a megfelelő felhúzóellenállásokkal, így több I2C-n [12] kommunikáló eszközt is ráköthetünk a panelunkra szükség esetén. A másik egy UART port, ami az ESP32

programozásához kell, igaz lehet USB-n keresztül is programozni ezt a verziót, de nem árt meghagyni az alternatív útvonalakat. Az utolsó csatlakozó meg a HX_UART, ez a port a modem programozására szolgál és az esetleges debug-olásra.

3.4 Kompatibilitás kivitelezése a chip hiány tekintetében

Korábban már említettem, hogy az NB-IoT vagy a Cat-M modem megoldás lenne a legjobb viszont alkatrész hiány következtében ez nem megoldható. A számunkra megfelelő és a GSM modemmel megegyező (vagy ahhoz közelálló) lábkiosztás esetén viszont megoldható a kompatibilitás. Szerencsére az Ai-Thinker kínálatában található egy ilyen modem, ami NB-IoT alapú kommunikációt támogat és beépített GNSS-el is rendelkezik. Ez nem más, mint az EC-01G [13] modem. A tápfeszültség tűréshatárai megegyeznek az A9/A9G-vel, az RF lábak ugyanott vannak és a SIM kártya is ugyanoda csatlakozik, így csak az UART csatlakozást kell megoldani mert ez az egy eltérés található rajta. Mivel az A9 és az A9G között is vannak eltérések, így ezeket 0603-as beforrasztható jumperekkel oldottam meg, szám szerint 8db-al, amit a kapcsolásrajzon láthatunk (az elnevezésük el lett rejtve, hogy elférjen minden az adott modul keretein belül, de a NYÁK-on láthatóak lesznek). A kapcsolásrajzban megtalálható még egy jumper NC8-néven, az arra szolgál, hogy a töltő áramkör előtt álló FET-et kikerülje szükség esetén.



5. ábra Nagyított kapcsolásrajz a modem környékéről

Látható, hogy a piros NC felirat (not connected) megtalálható pár ellenállás és LED mellett is, mivel az energiatakarékosság miatt ezeket nem kell beforrasztani, csak a prototípuson. A jumperek konfigurálása és a kompatibilitás megoldására a következő táblázat ad magyarázatot.

	NC1	NC2	NC3	NC4	NC5	NC6	NC7	NC9
A9+L80	-	-	I2C EN	I2C EN	-	-	-	-
A9G	X	X	I2C EN	I2C EN	-	-	-	-
EC-01G	X	X	I2C EN	I2C EN	*	*	*	X

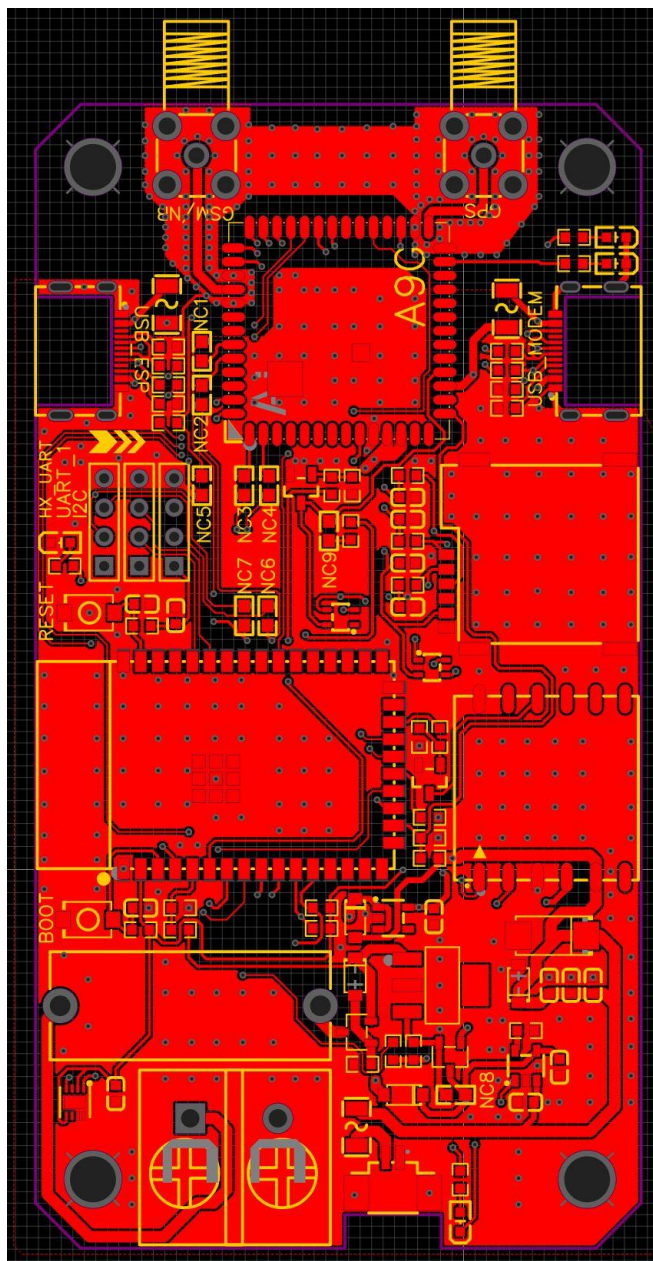
1. táblázat: Jumper konfigurációk

A táblázatból látható, hogy az NC3-4 jumper arra szolgál, hogy az I2C kommunikációt engedélyezzük, természetesen ehhez az is kell, hogy a firmware támogassa ezt a funkciót (főleg az A9/A9G esetében). Az NC5 esetében akkor szükséges a beforrasztás, ha az EC-01G nehezen reagál a hard reset-re, amit amúgy a tranzisztoros áramkörön keresztül vezérel az ESP32. Az NC6-7 esetében először fel kell tölteni a firmware-t az EC-01G-re a HX_UART porton keresztül, utána be lehet forrasztani a jumpert, hogy a GNSS része működjön. Firmware upgrade esetén meg ki kell forrasztani, ennyi kellemetlenséggel jár az áramkör rugalmassága.

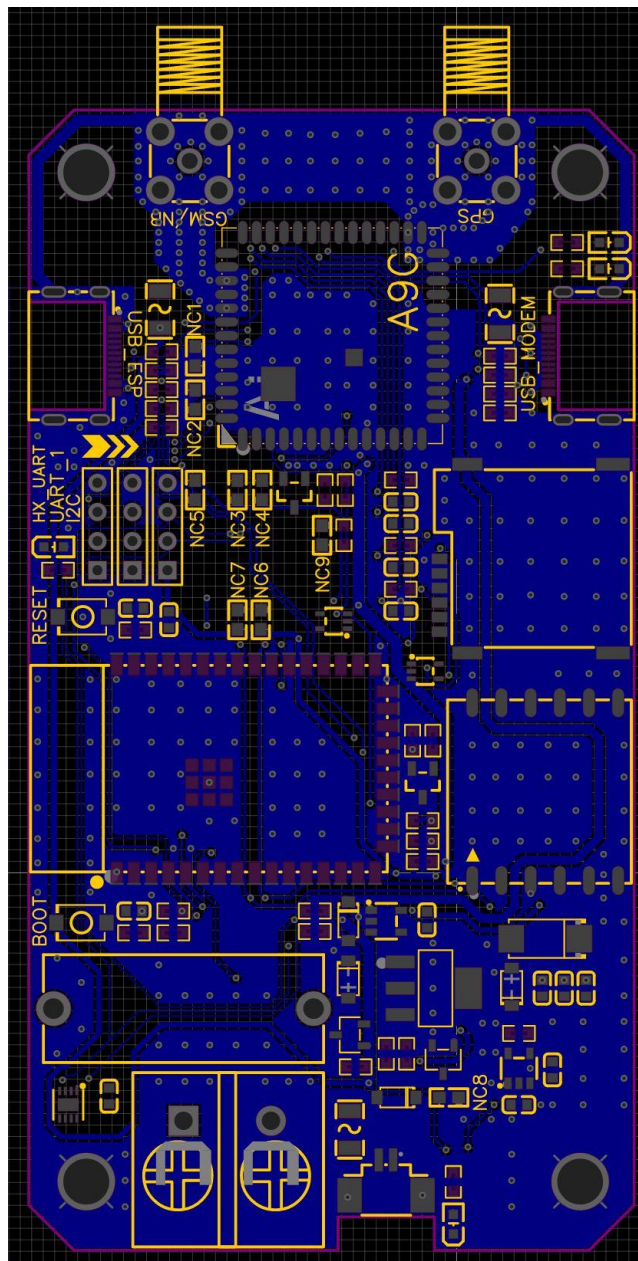
3.5 NYÁK tervezése

A NYÁK azaz nyomtatott áramkör tervezése során figyelniünk kell arra, hogy az RF (rádió frekvenciás) mikrostrip-ek a lehető legrövidebbek és a legjobb szigeteléssel rendelkezzenek. Az RF sávok átmérőjének kiszámításához egy online kalkulátort használtam [14], hogy megfelelő impedanciával illeszkedjen a modem az antennához (ebben az esetben 50Ω). A mikrostrip-ek kivitelezése esetén fontos szempont, hogy nem lehet nagyobb átmérője, mint az alkatrész lába, ellenkező esetben több útvonalon kell elvezetni az RF-et, hogy meglegyen a megfelelő impedancia. Éppen ezért a modem és a hozzá tartó alkatrészek a panel egyik oldalán fognak elhelyezkedni és az ESP32, töltés és feszültség osztók a panel ellentétes oldalán lesznek, hogy ne okozzanak interferenciát. A tápfeszültségeket szállító vezetősávoknak a nagyobb áramok miatt vastagabbnak kell lenniük, illetve minimális túlméretezés is szükséges, hogy túláram esetén ne azonnal égjenek el. Mivel a modem akár 1-2W-ot (30-33dBm) is képes kisugározni GSM esetén, így a vezetősávot 1mm-esre választottam. Mivel ez a sáv rácsatlakozik a 3.3V-os feszültség szabályzóra is, illetve a töltő áramkörre is, így nem árt a minimális túlméretezés. A többi tápfeszültség vezető sávját 0.5mm-esre választottam mivel azokon már nem léphet fel amperes nagyságrendű áram, maximum 3-500mA. A többi vezetősáv, amin vezérlőjelek mennek vagy esetlegesen adat, azokat 0.2-0.25mm-esre választottam, több nem is szükséges ezekben az esetekben.

A modemen kívül az SHT31-es alkatrész szintén érzékeny, mert a NYÁK-on található komponensek melegszenek, így meghamisítják a mérést, ezért a szenzort közel helyezem el az egyik sarokhoz, ahol többnyire el van szigetelve a többi komponenstől. A tápellátást biztosító és szabályzó komponenseket meg közel az akkumulátor csatlakozókhoz, így rövidebb vezetősávokat kell húzni, amivel a melegedést is csökkenteni lehet nagyobb terhelés esetén. Sok szó esett arról, hogy a pozitív vezetősávok milyen elrendezésben legyenek, és milyen szélességgel rendelkezzenek, de a földelés, azaz a GND iránya is rendkívül fontos. Éppen ezért ahol üres helyek vannak a panelon, ott GND kitöltést kell alkalmazni és fémes összeköttetéssel (a továbbiakban Via) kell a lehető legjobban összekötni az alsó és a felső réteget minél több helyen (főleg az RF sávok közelében). A leírt elvárások után a megtervezett NYÁK a következő ábrákon látható.

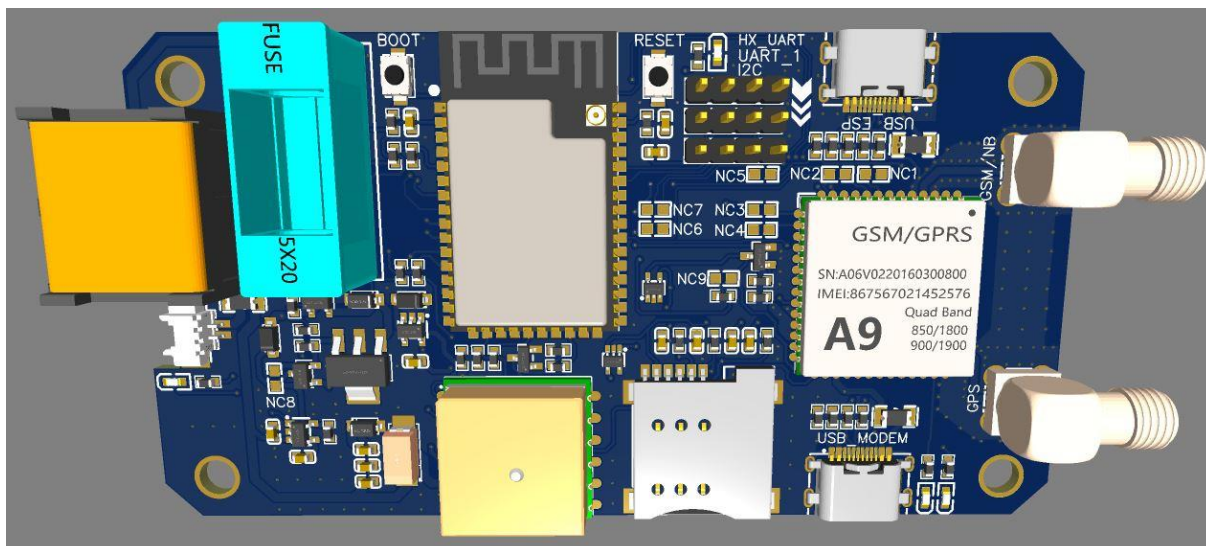


6. ábra A NYÁK felső rétege



7. ábra A NYÁK alsó rétege

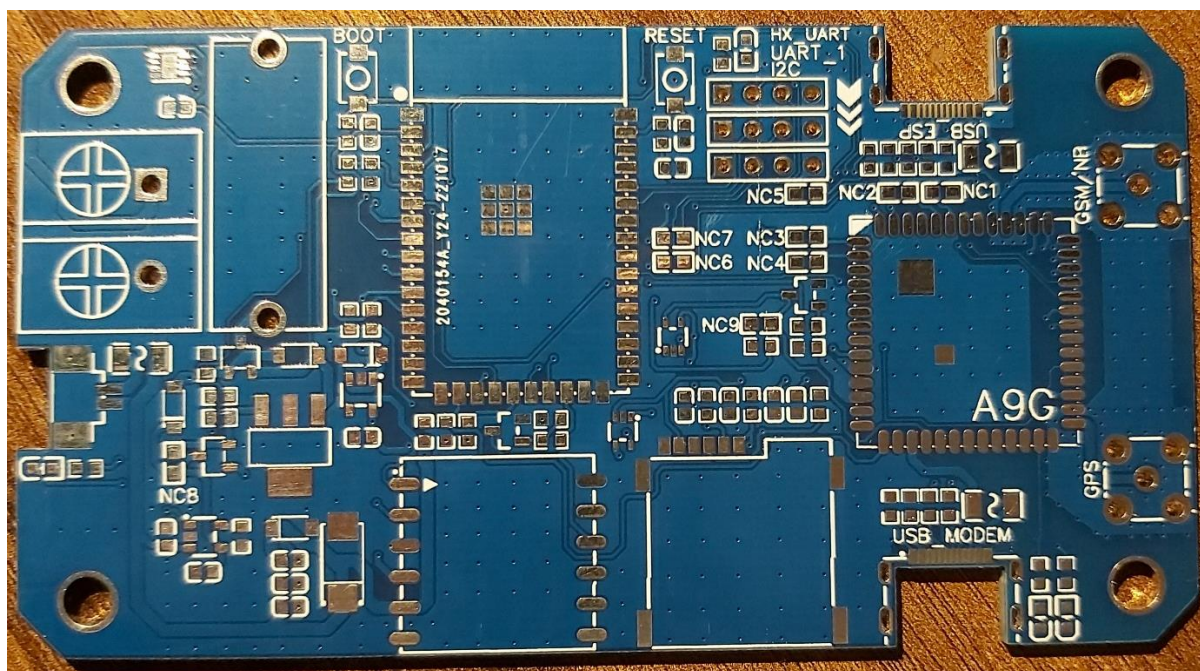
A panelen elhelyeztem 4db 4mm-es furatot is, hogy könnyebben rögzíthető legyen egy műszerdobozba vagy akár közvetlenül a járműbe. A megtervezett panel 99*52mm-esre sikerült, ezen lehetett volna csökkenteni, ha az ESP32 WROOVER modul helyett magát a chip-et használtam volna, de akkor ki kellett volna alakítani egy WiFi/ Bluetooth RF vezetősávot is. Továbbá, ha micro SIM helyett nano SIM-et használtam volna. A panelen láthatóan nagy GND régiók helyezkednek el és ezeket rendkívül alaposan össze is kötöttem egymással, szám szerint 475 Via található meg a NYÁK-on összesen.



8. ábra 3D-s előnézet a kész eszközről

3.6 A kész NYÁK átvizsgálása

A gyártást a JLCPCB [15] végezte el, a standard gyártási folyamat 3 napot vesz igénybe a maradék várakozást meg a szállítás foglalja magában, így mondhatjuk, hogy gyorsan kézhez kaphatjuk a megtervezett paneleket. A fizikai méretek mellett 1.6mm-es vastagságú FR4-es anyagból készült és kék színű felületi lakk-ot választottam rá, illetve HASL (hot air solder leveling) felületi kezeléssel rendelkezik, az eljárás során a kész NYÁK-on szabadon lévő réz érintkezőket ónnal látják el, úgy, hogy rövid ideig bemártják a paneleket egy forrasztókádba.



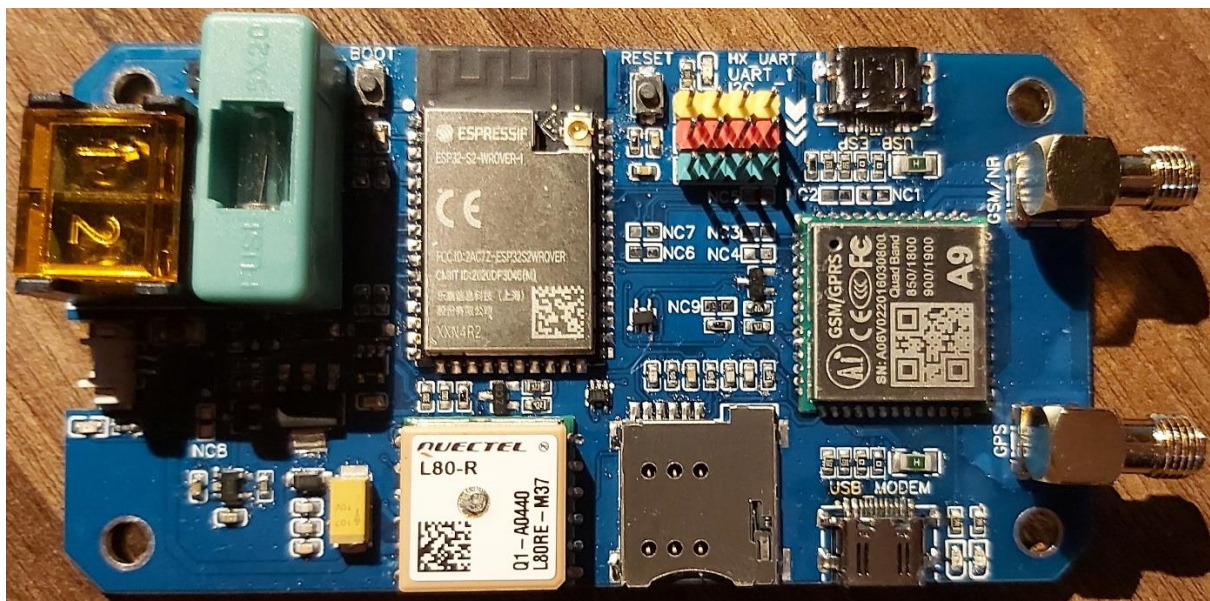
9. ábra Az elkészült NYÁK

Miután megérkezett a panel fontos, hogy átvizsgáljuk és ellenőrizzük a vezetősávokat és az alkatrészek érintkezőit, ezzel megspórolhatjuk azt a kellemetlen helyzetet, hogy az alkatrészek beültetése után valami nem szeretne működni, majd egy hosszú hibakeresés után kiderül, hogy az egyik vezetősávot vagy érintkezőt elrontották a gyárban. Az eszemben az átvizsgálás során nem talákoztam hibával, így elkezdhettem az összeszerelést.

3.7 Forrasztás és hiba keresés

A tervezés során sokféle alkatrészt használtam, de a beépítés során érdemes a kisebb SMD alkatrészekkel kezdeni, azon belül az ellenállásokkal és kondenzátorokkal, ezután következhetnek a tranzisztorok, FET-ek, LED-ek és a beforrasztható biztosítékok. Következnek a mikrokontroller, a modem, a sim kártya foglalat, nyomógombok és az akkumulátor csatlakozó forrasztható be, és az utolsó, amit el kell végeznünk az RF csatlakozók, USB-C és a THT alkatrészek beforrasztása.

A forrasztást kétfajta módszerrel végeztem el, az egyik esetben, ónpasztával kentem be az érintkezők felületét majd hőlégfúvóval olvasztottam meg. Ez a módszer az SMD alkatrészeknél előnyös, ahol sokszor a forrasztópákával már nem tudunk dolgozni. A másik módszer meg a hagyományos vékony laposfejjel rendelkező forrasztópákával történő forrasztás, főleg a THT alkatrészekhez szükséges, de bizonyos esetekben az SMD alkatrészeket forrasztásait is lehet korrigálni velük (esetleg, ha az ónpaszta kicsit összefolyik a mellette lévővel és zárlatot okozna).



10. ábra Az elkészült eszköz

Az ábrán látható, hogy az elkészült eszköz szinte megegyezik a 10-ik ábrán látott 3Ds előnézettel. Az ESP32, A9 modem és az L80-as GNSS modul esetén az érintkezőket a pasztás forrasztás mellett, még a rendes forrasztópákával is rámelegítettem, így biztosított, hogy az összes érintkező megfelelő fémes összeköttetéssel rendelkezik a panellel, illetve pár ellenállás esetén kellett a forrasztások újra melegítése, de ezt leszámítva gyorsan és sikeresen zártam ezt a folyamatot.

Az első tesztelés során rákötöttem egy USB-C-s hálózati adapterre és ellenőriztem a feszültség szabályzó komponensek kimeneti feszültségeit, illetve zárlatokat kerestem a panelen. Szerencsére az összes alkatrész megfelelően működött eddig és nem találtam forrasztásból vagy alkatrészhibából eredő zárlatot.

A korábban már említett névleges ellenállás értékektől való eltérés kiszámításához, először meg kell mérnünk az ellenállások valós értékét, ehhez egy UNI-T UT70a multimétert használtam. A mért értékek a következő táblázatban szerepelnek. Ezeket az ellenállásokat 1%-os tűréshatárúnak választottam, kifejezetten azért, hogy a lehető legpontosabb értéket kapjuk a számításokor, főleg a beépített akkumulátor esetében, ahol ebből az értékből határozza meg az ESP32, hogy mikor szükséges aktiválni a töltést.

Ellenállás prefixe	Névleges ellenállás	Mért valós ellenállás	Eltérés a névleges értéktől
R1	30k Ω	~30k Ω	<1%
R2	7.5k Ω	~7.49k Ω	1.3%
R19	100k Ω	~100k Ω	<1%
R20	200k Ω	~200k Ω	<1%

2. táblázat: Ellenállás mérés

A 2. táblázatból látható, hogy az ellenállások az elvárásoknak megfelelően az 1%-os tűréshatáron belül maradtak nagyjából. A mérés olyan szempontból pontatlan, hogy ezt a multimétert nem professzionális felhasználásra tervezték. A továbbiakban a SAR ADC-nél kiszámolt pontatlanság mellett hozzászámolhatjuk +-1%-os (+-33mV) pontatlanságot az ellenállások miatt, amikor kiszámoljuk a feszültségeket.

A modem-et nem kell programoznunk, mivel a gyártó biztosít hozzá firmware-t, amit a HX_UART porton csak rá kell töltenünk az első használat előtt. A modem USB-C csatlakozó a firmware aktiválását követően már használható lesz debug-olásra, vagy vezérléshez.

4 PROGRAMOZÁS

4.1 Az elvárt funkcionalitások és feladatok

Korábban már említettem, hogy az egyik legfontosabb része ennek az áramkörnek, hogy energiatakarékos legyen. Ezt az ESP32 átgondolt vezérlésével érhetjük el, mivel rendkívül alacsony fogyasztása van deep sleep üzemmódban (μA -s nagyságrendű), és az összes többi modult meg ki-be kapcsolhatjuk igény szerint. Időnként kell csak felébreszteni a mikrokontrollert deep sleep-ből, ebben az esetben a fogyasztás is megnő (10-100mA-ig) és bekapcsol a WiFi modul, de ez csak pár másodpercet venne igénybe pár óránként, és csak abban az esetben kapcsolná be a többi modult, ha az okosotthon szerverén keresztül kap egy wake up ping-et és lekérjük az információkat, vagy ha a fix időközönként beállított ellenőrzéskor érzékeli az akkumulátor merülését (elérte a kritikus szintet). Fontos, ha nem tud csatlakozni az otthoni WiFi-re pár próbálkozás után, akkor kapcsolja ki ezt a modult és a modemén keresztül kommunikáljon. Fontos, hogy a modem hosszabb boot időt igényel, ezért csak akkor kezdjen el adatot küldeni az ESP32 amikor már válaszol a modem. A hőmérséklet és páratartalom szenzor az akkumulátor feszültségének ellenőrzéséhez hasonló módszerrel működne, annyi különbséggel, hogy ezt az információt csak WiFi-n keresztül továbbítja az ESP32, a modemnek nem küldi el. Azért alakítottam ki így a működését, mivel a koordináták és akkumulátor feszültség infó mellé nem szükséges ebben az üzemmódban. Hiba esetén nem csak soft reset üzenetet lehet küldeni a modulnak, hanem hard resetet is, egy-egy tranzisztoros áramkörön keresztül, a GNSS modul és a modem rendelkezik ezzel a kiegészítővel. A programozást az Arduino IDE programozói környezetében végeztem el [16], és az így megalkotott program blokkvázlata a 4.2-es fejezetben található.

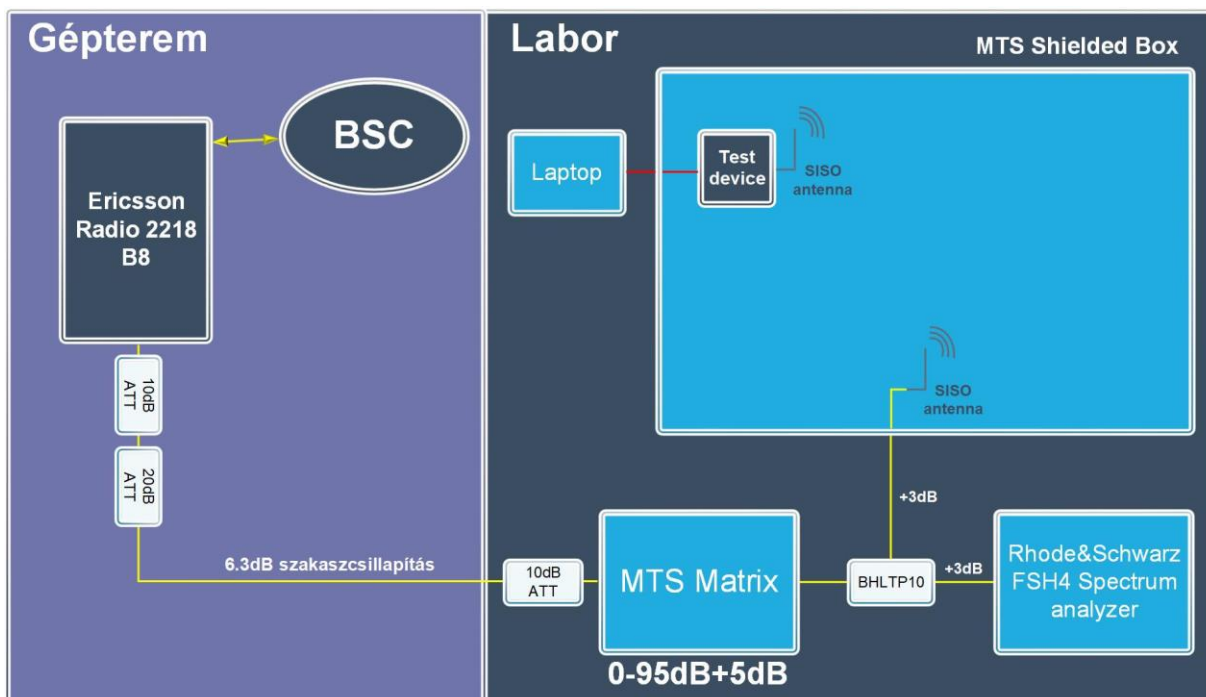
5 TESZTELÉS

5.1 Tesztelés fázisai

A tesztelés két külön részre lesz osztva. Az egyikben az okosotthon környezetében fogom vizsgálni az eszközt, hogy jelzi-e az akkumulátor feszültségeket, illetve a hőmérséklet és páratartalmat a Thingspeak [17] szerverén keresztül. A másik esetben meg a mobilhálózati képességeket fogom vizsgálni, hogy a GSM modem valóban képes-e elvégezni a feladatát különböző körülmények között.

5.2 Mobilhálózati mérés teszt körülményei

A mobilos mérésekhez a Magyar Telekom TDI (Technology Delivery International) SSC laboratóriumában található eszközöket használtam, illetve az ott dolgozó kollégák segítségével végeztem el a méréseket. A mérés során készített fotókra és az eszközök használatára írásbeli engedélyt kaptam, így elhelyezhetem a TDK dolgozatban a fotókat és az eredményeket.



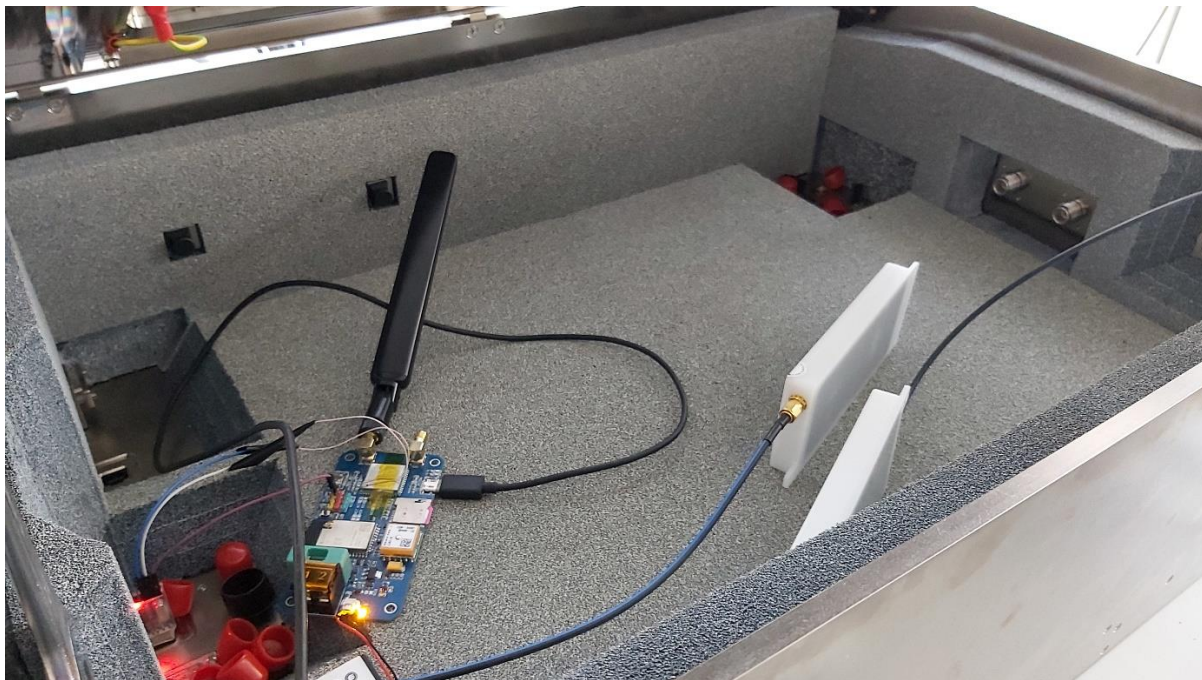
12. ábra Mérési elrendezés a laboratóriumban

A mérési elrendezésben látható, hogy az Ericsson 2218 Band 8 (900MHz) RRU (Remote Radio Unit) a gépteremben helyezkedik el és egészen nagy csillapítókkal van ellátva. A mérés alatt az RRU 34dBm-es maximális teljesítménnyel ment, így kizárhatjuk azokat az eseteket

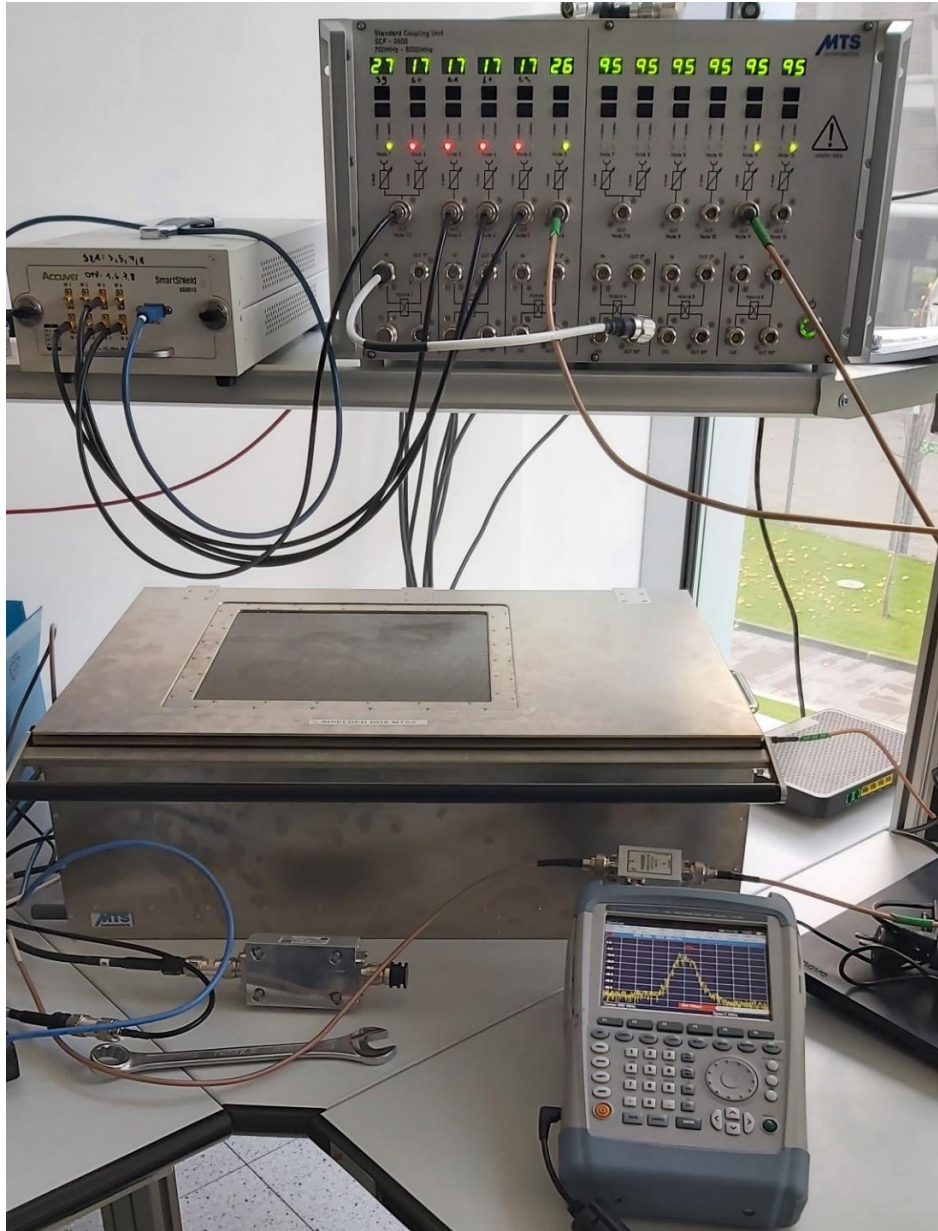
amikor a gyengébb teljesítmény miatt fellépne valamilyen korlátozó tényező vagy elvesztés. A teljes csillapítás, amivel számolnunk kell a mérés során az ATT (Attenuator, azaz csillapító) nevű csillapítók, a szakasz csillapítás (ami a gépteremből a laborig lép fel), az MTS mátrix belső csillapítása (5dB 900Mhz környékén) és a BHE Bonn BHLT10 elágazásban fellépő csillapítások összege. Ezt egy offset (eltolás)-ként kell beírunk a Rhode&Schwarz FSH4 spektrum analizátorba, hogy korrekt értékeket jelenítsen meg. Az MTS shielded box egy Faraday kalitka, ami 80dB-90dB-el csillapítja a kintről érkező jeleket, így biztos lehetünk abban, hogy a benne lévő eszköz csak azt a teszt cellát használja, amit bevezetünk neki. A dobozon belül az RF kábel bemenő csatlakozójára egy GSM frekvenciákkal kompatibilis (a jelen esetben 900MHz környékén) antennát csatlakoztatunk, az eszközre meg egy ugyan olyan (vagy hasonló képességű) antennát csatlakoztatunk, ezzel egy SISO (single-input, single-output) antenna elrendezés valósítunk meg ami GSM esetében elég. A teszt eszköz meg egy USB portra van bekötve a dobozon belül, amire kívülről a doboz oldalán csatlakoztatunk egy USB kábelt és UART-on keresztül kommunikálunk vele. A tápellátását az eszköznek az akkumulátora és egy külön USB-C-s hálózati adapter biztosítja. A hálózati feszültség és az USB portok zajszűrését, egy aktív szűrő végzi el, ezért ezt be kell kapcsolnunk a mérés megkezdése előtt. A számítógépen egy AT command tester [18] nevű programot használtam, ezzel a standard AT parancsok [19] kiadása mellett előre beprogramozott funkciókkal is gyorsíthatjuk a mérést, vagy script-eket is írhatunk egy komplett mérés automatizálásához.

5.3 Mobilhálózati mérés eredmények

Az első mérés során az MTS mátrix állítható csillapításával beállítottam, egy elfogadható és valós körülmények között is előforduló RSRP [20] szintet (az eszköz mérése alapján a -70dBm-es szintet céloztam meg kezdőértéknek), majd ettől a szinttől kezdve, növeltem a csillapítást a mátrixon egészen addig a pontig amikor az eszköz leszakad a hálózatról, a GSM esetén ez olyan -108dBm környékén várható. Az eszköz elhelyezkedése a mérődobozon belül a következő ábrán látható.



13. ábra Mérődoboz belseje



14. ábra Mérési elrendezés a laborban

Mért RSRP az eszközön	MTS mátrix beállított csillapítása
-69dBm	26
-75dBm	30
-81dBm	36
-91dBm	46
-95dBm	51
-103dBm	57
-107dBm	62
-109dBm	64

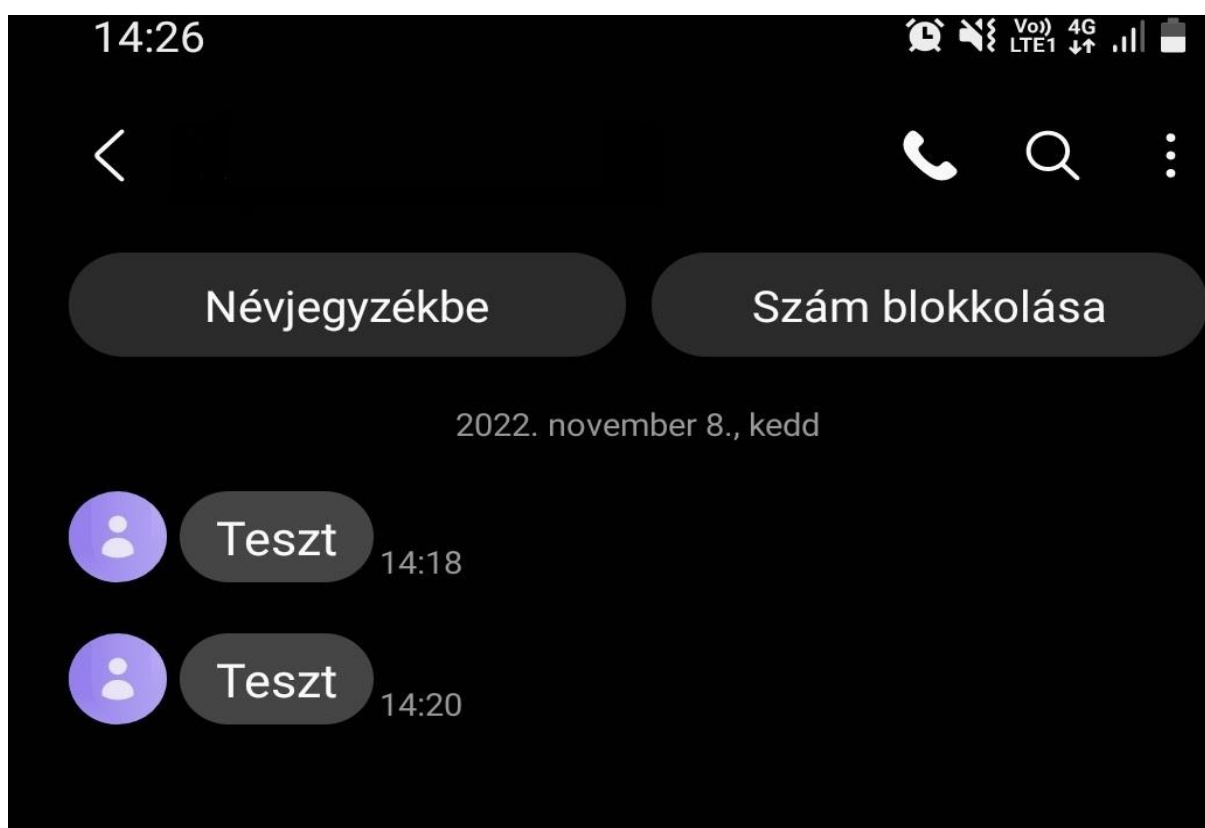
3. táblázat: RSRP mérés eredményei

A mérési eredményekről elmondható, hogy jól teljesített az eszköz és -109dBm-es RSRP szint mellett még éppen, hogy működött majd eldobta a hálózatot az elvártaknak megfelelően.

A második teszt során a GSM funkcionalitását fogom vizsgálni különböző RSRP szintek mellett, a teszt során három darab SMS-t szeretnék küldeni a saját telefonomnak.

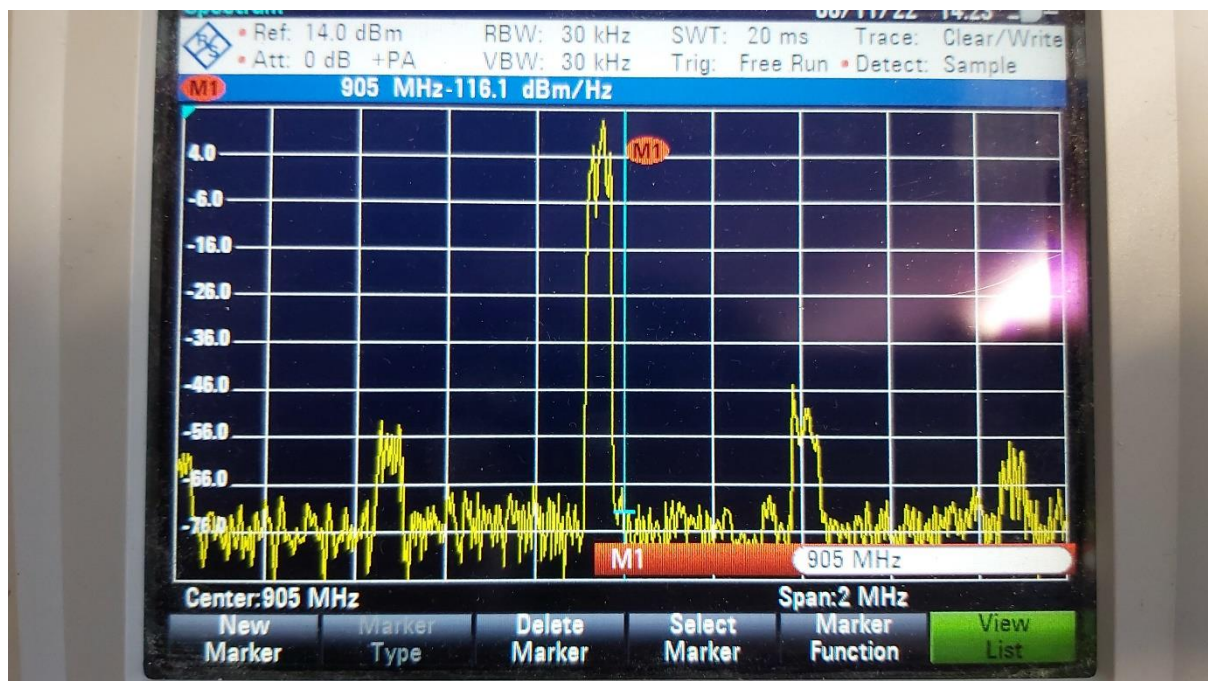
Beállított RSRP	Sikeres	Nem sikeres
-71dBm		-
-105dBm		-
-115dBm	-	

4. táblázat: SMS funkció vizsgálata különböző jelszinteken



15. ábra SMS teszt

Látható, hogy ahol az előbbi mérés során már ledobta a hálózatot a modem ott, illetve azon túl már az SMS-sem fog működni, vagyis az elvárásoknak megfelelően működött az eszköz. Az RSRP növelésével együtt viszont megnőtt az SMS küldési ideje is, az első tesztben nagyjából 10 másodpercet vett igénybe, a második esetén ez már 15-17 másodperc környékén volt. Az FSH4 spektrum analízátoron az uplink irányú (905MHz-en) az SMS küldése közben megfigyelhető voltak a burst-ök, a mérés során ez az eszköz a megfelelő jelszint beállításához volt szükséges.



16. ábra Az uplink irányú forgalom vizsgálata

5.4 Okosotthon környezetében végzett vizsgálatok

A méréseket a Thingspeak segítségével végeztem el, ennek a honlapnak a lényege, hogy egy szerverként üzemel, amikor mikrokontroller feljelentkezik az internetre akkor ennek a szervernek küldi el a szenzor információkat. Ezt különféle grafikonokká alakíthatjuk és integrálhatjuk az okosotthon környezetében, mint egy külső információ forrás.

A szerveren hozzáadtam a jármű akkumulátor feszültségét és a belső akkumulátor feszültségét jelző grafikonokat, illetve egy trigger eseményt amikor eléri a például 11.5V-ot a járműben (ezt a felhasználó saját maga konfigurálná a jövőben) illetve a beépített akkumulátor esetén a 3.5V-ot (ez egy fix nem változtatható érték lenne) akkor küld egy e-mailt a szerver. A kódot legenerálja a honlap, onnantól kezdve már csak be kell másolni a programozó környezetbe a kódot és a hitelesítő kulcsot. A teszt során a jármű akkumulátor kapocsra egy labortápot kötöttem (0-15V feszültségtartományban állítható) és elkezdtem csökkenteni a feszültséget. Amint elérte a trigger pontot, pár másodperccel később már meg is érkezett az előre konfigurált teszt e-mail (benne a Teszt szöveggel). Ezt megismételtem az akkumulátor esetében is, ekkor USB-n kapta a tápfeszültséget az áramkör, ebben az esetben is hibátlanul működött az áramkörünk, illetve a szolgáltatás.

6 ÖSSZEGZÉS

A projekt elején kitűzött célokat elértem, és sikeresen megalkottam az eszközt. A mérések bizonyították, hogy ez a konstrukció működőképes és érdemes továbbfejleszteni. A műszerdoboz tervezése folyamatban van, mivel a megfelelő kialakítás és anyagválasztás fontos a járművek környezetében, viszont mindenképpen 3D nyomtatással szeretném ezt megoldani. Továbbfejlesztés szempontjából, lehetőség van arra, hogy több szenzort integráljunk a panelra, illetve a jelenleg használt WROOVER-I modult kicseréljem a szimpla chip verzióra, és kialakítsak egy megfelelő antenna vezetést a WiFi-nek és a Bluetooth-nak. A chip hiány következtében kialakult nehézségek lecsengése után, a töltőáramkört és a modem részét újra lehet tervezni, hogy az eredetileg tervezett modernebb mobilhálózati technológiákat használjon a GSM helyett.

A következő mérés, amit a műszerdoboz kivitelezése után el kell végezni az eszközzel az, hogy valós körülmények között is megállja-e a helyét. A laboratóriumi mérések jók arra, hogy kontrollált környezetben megvizsgáljuk a funkciókat és a viselkedését az eszköznek, de a valóság nem ideális, ezért kell tovább tesztelni.

7 IRODALOMJEGYZÉK

- [1]: <https://www.telekom.hu/lakossagi/szolgalatasok/mobil/lefedettseg#hasznosinfok> (letöltve: 2022.11.05)
- [2]: https://www.telekom.hu/static-la/sw/file/Nb_IoT_Lefedettseg.pdf (letöltve: 2022.11.05)
- [3]: <https://www.simcom.com/product/SIM7070G.html> (letöltve: 2022.11.09)
- [4]: https://www.quectel.com/wp-content/uploads/pdfupload/Quectel_L80_GPS_Specification_V1.3.pdf
(letöltve: 2022.11.09)
- [5]: https://www.espressif.com/sites/default/files/documentation/esp32-s2_datasheet_en.pdf (letöltve: 2022.11.09)
- [6]: https://docs.ai-thinker.com/_media/b100ps01a3_a9_product_specification.pdf (letöltve: 2022.11.09)
- [7]: https://www.espressif.com/sites/default/files/documentation/esp32-s2-wrover_esp32-s2-wrover-i_datasheet_en.pdf (letöltve: 2022.11.09)
- [8]: <https://www.circuitbasics.com/basics-uart-communication/> (letöltve: 2022.11.09)
- [9]: <https://easyeda.com/> (letöltve: 2022.11.09)
- [10]: <https://lcsc.com/> (letöltve: 2022.11.09)
- [11]: https://www.mouser.com/datasheet/2/682/Sensirion_Humidity_Sensors_SHT3x_Datasheet_digital-971521.pdf (letöltve: 2022.11.09)
- [12]: <https://www.circuitbasics.com/basics-of-the-i2c-communication-protocol/> (letöltve: 2022.11.09)
- [13]: https://docs.ai-thinker.com/_media/nb-iot/ec-01g_specification_english_-zw20210602.pdf (letöltve: 2022.11.09)
- [14]: <https://www.eeweb.com/tools/microstrip/> (letöltve: 2022.11.09)
- [15]: <https://jlcpcb.com/> (letöltve: 2022.11.09)
- [16]: <https://www.arduino.cc/en/software> (letöltve: 2022.11.09)
- [17]: <https://thingspeak.com/> (letöltve: 2022.11.09)
- [18]: <https://m2msupport.net/m2msupport/download-at-command-tester/> (letöltve: 2022.11.09)
- [19]: https://docs.ai-thinker.com/_media/b_and_t/nb-iot/n92/kaifazhidaowendang/rda8908a_at_commandmanual9.0.pdf (letöltve: 2022.11.09)
- [20]: <https://www.cablefree.net/wirelesstechnology/4glte/rsrp-rsrq-measurement-lte/> (letöltve: 2022.11.09)

8 ÁBRAJEGYZÉK

1. ábra Magyar Telekom GSM lefedettsége.....	4
2. ábra Magyar Telekom LTE lefedettsége	5
3. ábra Magyar Telekom NB-IoT szolgáltatásának lefedettsége	5
4. ábra A megtervezett áramkör kapcsolásrajza	9
5. ábra Nagyított kapcsolásrajz a modem környékéről	12
6. ábra A NYÁK felső rétege	15
7. ábra A NYÁK alsó rétege	15
8. ábra 3D-s előnézet a kész eszközről.....	16
9. ábra Az elkészült NYÁK.....	16
10. ábra Az elkészült eszköz	17
11. ábra A program blokkvázlata	20
12. ábra Mérési elrendezés a laboratóriumban.....	21
13. ábra Mérődoboz belseje	23
14. ábra Mérési elrendezés a laborban	24
15. ábra SMS teszt.....	25
16. ábra Az uplink irányú forgalom vizsgálata	26