

Horváth János

Kilenc szabadságfokú MEMS szenzor gyakorlati alkalmazása

TDK dolgozat

Konzulens: Dr. Pilászy György, Rácz György

2016/17 tanév őszi félév

Tartalomjegyzék

Bevezetés	3
A korábbi rendszer bemutatása	4
Hardver egység.....	4
Működés.....	6
A mérés menete.....	6
Rádiós kommunikáció	6
Paraméterek.....	6
Feladat megfogalmazása.....	6
Az új rendszer bemutatása.....	7
Alapfunkciók.....	7
Működés.....	8
A mérés menete.....	8
Rádiós kommunikáció	8
Protokoll:	9
Paraméterek.....	11
Módosítások.....	11
A rendszer általános paramétereinek, funkcióinak változtatása	11
A szenzorral végzett vizsgálatok.....	12
A rádiós kommunikáció üzenetének feldolgozási idejének gyorsítása és zajszűrése	13
Válasz-időzítések egységesítése	13
Mérések, eredmények.....	15
Adatfeldolgozás.....	15
Adatok transzformálása	15
A szenzor tengelyek hibáinak meghatározása:	17
Használt módszerek.....	19
A mérés eredményeinek kiértékelése.....	28
Gyakorlati alkalmazási lehetőségek	32
Összefoglalás	32
Jövőbeli tervek	33
Források.....	34

Bevezetés

A szenzortechnika fejlődésével megjelentek az igen kicsi gyorsulás, szögsebesség és orientáció mérésére alkalmas mikroelektromechanikai (MEMS) szenzorok. Ezek az eszközök kis méretük és tömegük ellenére nagy felbontással képesek mérni a fizikai paramétereiket. Többek között az orvosi diagnosztika területén a mozgásvizsgálat kiemelt jelentőséggel bír. Számos torna és terápia során mutatkozik igény a foglalkozások eredményességének, a páciens állapotában bekövetkező fejlődések műszeres mérésére, dokumentálására.

A céloom egy olyan segédrendszer kialakítása volt, amely egy optikai mozgásérzékelő rendszerrel szenzor fúziót alkotva, tetszőleges emberi mozgást kis mintavételi idővel jobban tud követni, mint egy optikai mozgásérzékelő rendszer önmagában. Ez a rendszer a továbbiakban orvosi felhasználásra is alkalmas lehet. A munkám során egy korábban kifejlesztett, már meglévő rendszert alakítottam át, egy újabb, másik funkciót betöltő rendszerré.

A meglévő rendszert a BME Irányítástechnika és Informatika tanszékén fejlesztették ki. Ebben a rendszerben a mérés sokáig több napig, hónapokig tartott nagy mintavételi időtartammal (10-15 percenkéntivel). A mért adatok kiolvasása nagyobb időközönként, havonta egyszerre történt.

A munkám során ennek a rendszernek a hardver eszközeit felhasználva, annak működtető szoftvereit alakítottam át úgy, hogy képes legyen valós idejű adatgyűjtésre és megjelenítésre. A megvalósított rendszer alkalmas lett arra, hogy egy precíziós kamerarendszer feldolgozó szoftverével is lehessen szinkronizálni a mintavételezések időpontját.

A fejlesztés során a szenzor működését és beállításait megváltoztattam. Kidolgoztam a valós idejű adatgyűjtéshez, a kommunikációhoz és az összeszinkronizáláshoz egy új kommunikációs protokollt, az érzékelők kalibrálásához egy új eljárást, amely egy, a szakirodalomból ismert, kidolgozott eljárás alapul. Az új eljárás felhasználásával meghatároztam az egyes gyorsulás mérő, mágneses orientáció mérő tengelyeinek hibáit.

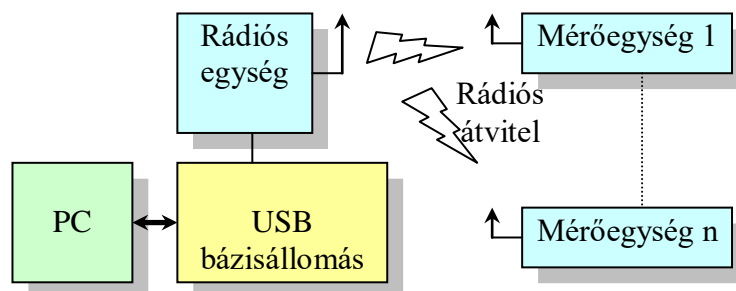
A következőkben röviden bemutatom a korábbi rendszert.

A korábbi rendszer bemutatása

A következőkben a rendelkezésemre álló eredeti rendszer hardverét és működtető programját ismertetem. A rendszernek csak azon funkcióit mutatom be, amelyeket felhasználtam a munkámhoz. Az alábbi pontban röviden ismertetem az általam felhasznált hardver egységek felépítését.

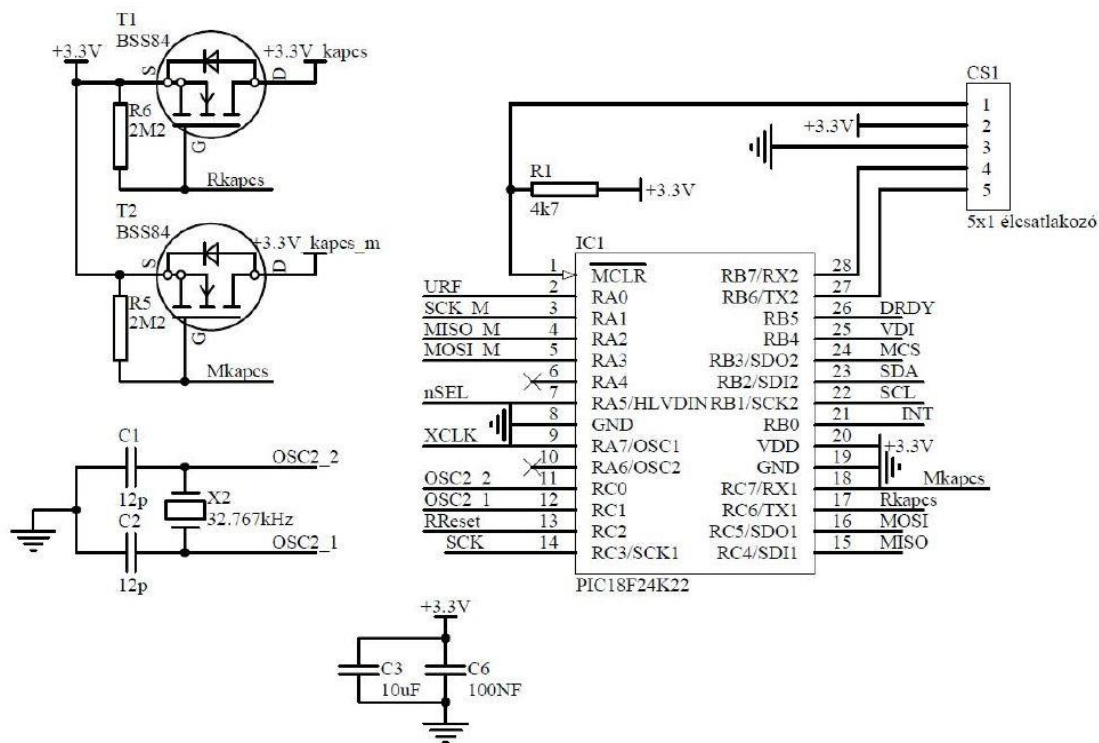
Hardver egység

A teljes rendszer két fő alkotórészt tartalmaz, egy olyan bázisállomást, amely az USB-porton keresztül képes kapcsolódni a számítógéphez, valamint mozgó egységeket, amelyek rádióon keresztül kommunikálnak a bázisállomással. (Lásd 1. ábra)



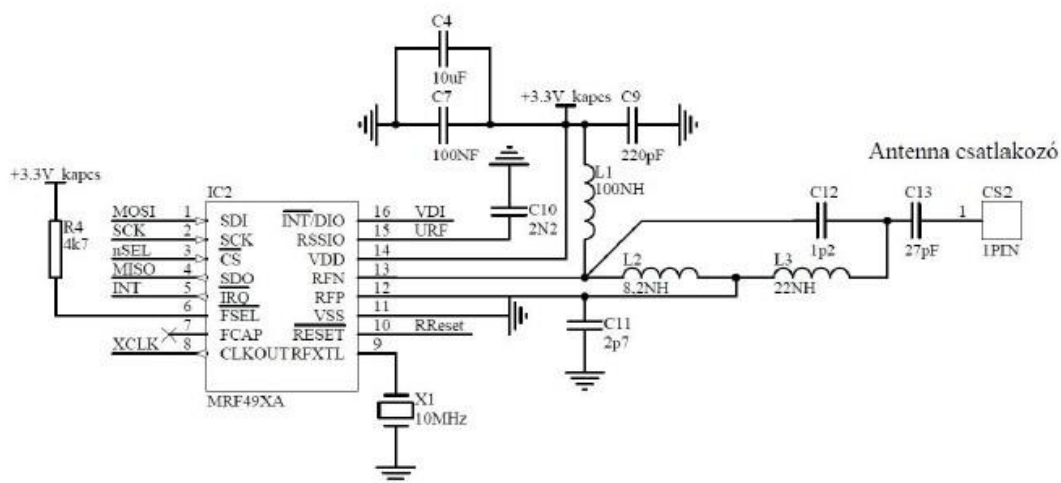
1. ábra, Rendszer sematikus rajza

A mozgóállomások adott frekvencián, három dimenzióban mérik a gyorsulást, mágneses orientációt és a szögsebességet. A mozgóállomás központja egy PIC18F24K22 mikrokontroller (2. ábra), erre csatlakozik a többi periféria. A mikrokontroller számos funkciót lát el. Inicializálja indításkor a többi modult, elkéri a mérő állomásoktól a mért adatokat, az EEPROM memóriát kezeli. A rádiós modulon keresztül jövő adatokat értelmezi, feldolgozza és a küldendő csomagokat elkészíti és a rádiós modulon keresztül továbbítja.



2. ábra, A szenzor mikrokontrollere [2]

A rádiós kommunikációt lebonyolító egység egy MRF49XA modul (3. ábra). Mind a bázisállomáson és a mozgóállomásokon ugyanolyan rádiós modul van. A mozgóegység a mikrokontrollerrel SPI protokollon keresztül kommunikál.



3. ábra, Kommunikációs egység [2]

A gyorsulásmérő, a mágneses orientációmérő és a szögsebességmérő egy LSM9DS0 szenzor. A működtető program a szenzort a mikrokontroller indításánál felinicializálja, majd a szenzor által mért adatokat az SST25VF040B 4 Mbit-es soros flash memóriába tárolja.

A bázisállomásoknak a szenzorok vezérlésében és a hely meghatározásában volt szerepük, ezen funkciókat nem használtam az új rendszerben ezért nem részletezem.

Működés

A teljes korábbi rendszer bemutatása az [1] irodalomban megtalálható, itt csak a feladat szempontjából lényeges részeket emelem ki.

A mérés menete.

Minden szenzort rögzítettek egy mozgóegységre (pl.: nagyobb méretű halra). A szenzor ezután egyfolytában működött (kb. 1 hónap) és 15 perces időközönként végzett méréseket, melyet a saját Flash memóriájába lementett. Az energiatakarékosság miatt a szenzor egy alacsony frekvenciás állapotba kerül, melynél a fogyasztás nagyságrendekkel kisebb, mint normál működésnél. Az alacsony frekvenciás állapotból akkor lép normál működésbe, ha rádió jelet fog vagy mérnie kell. Az adatok kiolvasása két-három hetes időszakonként történt egyszerre, melynél az adatátviteli idő nem számít.

Rádiós kommunikáció

A rádiós kommunikáció egy üzenetének a struktúrája a következő:

1 byte start karakter

1 byte id

1 byte parancs

Több byte, a parancshoz tartozó paraméterek vagy a parancsra jövő válasz

2 byte hiba ellenőrző összeg

1 byte stop karakter

Az üzenet keretes szerkezetű. Használ speciális karaktereket, és ha ilyen speciális karakterű adat jön az üzenet közben, azt úgy kezeli a szenzor, hogy beszúr elé egy speciális karaktert, melyet a feldolgozó fél eldob, de tudja, hogy ez nem speciális karakter, hanem az üzenetben egy adat byte.

Paraméterek

A mérés alaptulajdonságai:

-Mintavételi frekvencia (gyorsulás, mágneses orientáció, szögsebesség):1-2perc

-Memória kiolvasási idő: kb. fél óra

-A mérés ideje: 2-3 hónap

Kommunikáció:

-Adatsebesség: 1,2 kbps

-Sávzélesség: 134kHz

-Vivőfrekvencia: 436.2500mHz

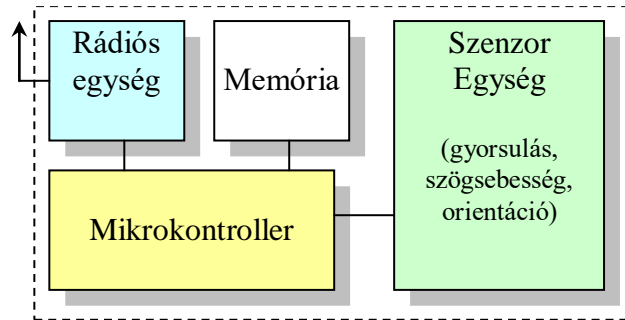
-Frekvencialöklet: 90 kHz

Feladat megfogalmazása

Az előző pontban bemutatott áramkör gyorsabb működésre is képes (nagyobb mérési gyakoriság illetve gyorsabb adatátvitel). A rendszerben rejlő képességek alapján a fent bemutatott rendszert alakítottam át egy olyan rendszeré, mely egy optikai mozgásérzékelő rendszerhez illesztve az egyes képkockák közötti időpontok között is vesz fel mozgási adatokat. A kifejlesztett mérőegységből származó adatok segítségével a mozgás, a képkockák mintavételezése között is követhetővé válik, így pontosabb mérések valósíthatók meg.

Az új rendszer bemutatása

Az új mérőegység blokkvázlatát mutatja az **Error! Reference source not found.** A teljes rendszer több mérőegységet is tartalmazhat. Egy mérőegység tartalmaz 3D-s magneto-, accele-méter és giroszkóp szenzorokat, rádiós egység és egy EEPROM memóriát. A rendszer központja a mérőegységekkel való kommunikációt megvalósító bázisállomás, amely egy PC USB portjára kapcsolódik. A bázisállomás a rádiós egységen keresztül kommunikál - az alkalmazástól függően - egy vagy több szenzorral. Minden pc által küldött üzenetre az összes megcímzett mérőegység válaszol. A mérőegységek fogadják a bázisállomás parancsait, veszik a szenzorok adatait, mentik az adatokat a memóriába és rádión keresztül továbbítják az adatokat a pc-nek.



4. ábra, Az új mérőegység blokkvázlata

Alapfunkciók

A következő parancsokat adhatja a szenzoroknak a pc:

- id lekérdezés
- start parancs (automatikus méréssorozat indítása)
- mérés vége
- a legutóbbi mért adatok lekérdezése
- mérési gyakoriság beállítása (mintavételezési idő)
- válasz küldés késleltetésének megadása
- a beállított paraméterek és memória törlése
- memória írási ciklusszámlálójának lekérdezése
- tárolt rekordok lekérdezése

A mérőegység a következő műveleteket végzi:

- rádión kommunikál
- a szenzoroktól lekéri az adatokat
- adatgyűjtő memória írása/olvasása
- a szenzorok és az adatgyűjtő memória inicializálása bekapcsoláskor vagy külön parancsra

Működés

A mérés menete.

A mérések összehangolását a PC-n futó program végzi. Minden mérés három részből áll. Első részben a szenzorok feltérképezése, inicializálása folyik.

A szenzorok feltérképezésénél a pc ad egy id lekérdezés parancsot, melyre minden szenzor, amely megkapta válaszol. A fogadott válaszokból a pc kilistázza a talált szenzorokat, majd innen ki lehet választani, hogy melyik szenzorokat szeretnénk használni a mérésnél. Ezután az egyes szenzorok paramétereinek beállítása következik (a válasz késleltetése, mérési frekvencia).

A második részben a mérés elkezdése, óra felszinkronizálás és maga a mérés folyik. Minden parancsot a pc küldi. Először a start parancsot küldi periodikusan míg minden szenzor egy start parancsra egyszerre nem válaszol, ekkor a szenzorok az időzítéseiket nullázzák. A pc ezután periodikusan küldi az aktuális adat lekérdezési parancsot, a szenzorok az első ilyen fogadott parancsra a mért adatokat elkezdik a memóriába menteni. A pc a mérés kezdetét attól számítja, amikor minden mérésben részt vevő szenzor legalább egyszer válaszolt egy ilyen parancsra. Ekkor a pc megjegyzi, hogy az egyes szenzorok hányadik mérésüket végezték és lementi azt, ettől az adattól kezdve fogják a szenzor memóriájából kiolvasni az adatokat. A mérés végét egy stop parancssal állítjuk le, amelyet addig küldünk periodikusan míg minden mérésben résztvevő szenzor egyszer nem válaszolt rá.

A harmadik részben a szenzorok memóriájából a mért adatokat kiolvassa a pc és elmenti. Mindig egy szenzortól olvassa ki az adatokat. A bázis üzenetben megmondja, hogy melyik adatokat akarja elkérni. Az egyes üzeneteket addig ismétli, míg nem kap rá hibamentes választ. Addig kéri folyamatosan az adatokat, amíg az összes adatot nem kapta meg. Ezután a következő szenzortól kérhetjük el az adatokat.

Rádiós kommunikáció

A kommunikációnál minden pc-s üzenetre minden címzettől jön válasz. Ha egy szenzor címzett, akkor attól az egytől jön válasz egy megadott késleltetés múlva. A pc-s parancsok közül a start parancs, a mérés vége- és aktuális adat lekérdezés parancsoknál várok több szenzortól választ. Ezen három parancs közül az aktuális adat lekérdezés parancsra való válasz a leghosszabb. Ha ezen válaszokat egymástól el tudom különíteni, akkor a többi parancsra adott válaszokat is el tudom különíteni egymástól.

Aktuális adat lekérdezési parancs válasza:

1 byte start karakter

2 byte üzenet hossz

1 byte id

1 byte parancs karakter

Egy rekord mérete: $3 \cdot 3 \cdot 2 = 18$ byte adat (minden adat 2 byte-os, 3 tengelyen 3 különböző adat típus),

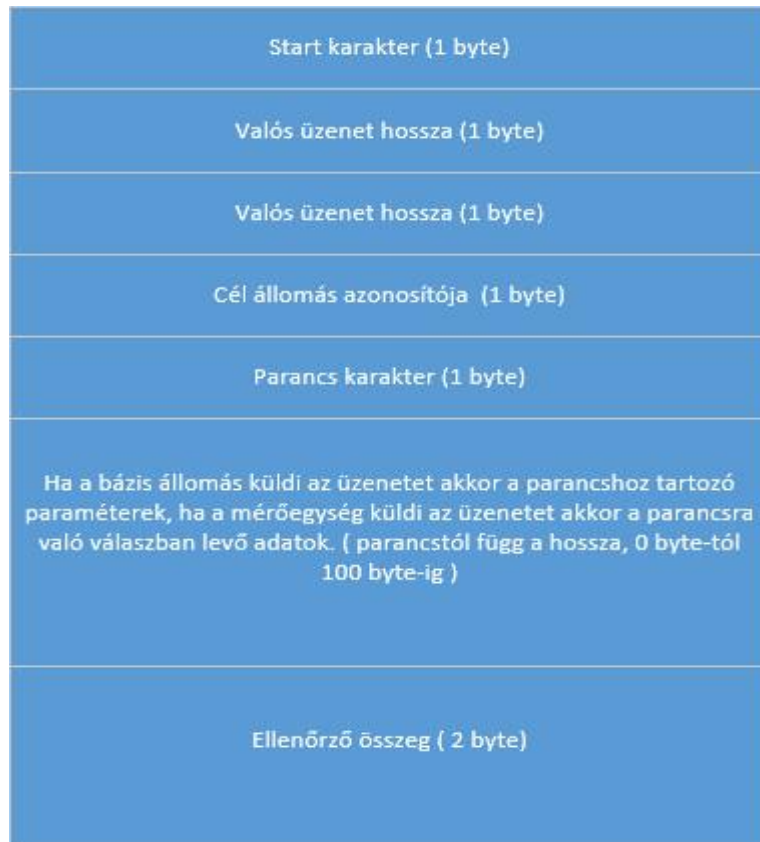
2 byte ellenőrző összeg

A teljes adatcsomag: $1+2+1+1+18+2 = 25$ byte összesen.

A kommunikáció lebonyolítása miatt kell az elején két szinkronizáló byte-ot küldeni, és végén három byte-ot hogy a végső byte-ot biztosan jól elküldje a rádiós modul. Tehát összesen egy aktuális adat lekérdezési parancs válasza 30 byte hosszú. Ezt a rádiós modul legnagyobb baud rate-jével (86kbit/s) $30 \cdot 8 / 86000 \approx 3,5$ ms ideig tart.

Tehát egy időosztásos protokollal, melyben minden szenzornak van egy 4 ms-os időrése, meg lehet oldani a szenzorok válaszáinak szétválasztását.

Protokoll:



5. ábra, Kommunikációs keret felépítése

Az üzenet mindig egy adott formátumú karakter sorozat.

A PC-től kapott lekérdező parancsokra formátuma:

Első karakter egy start szimbólum.

A második és harmadik karakter ugyanaz, még pedig a valós üzenet hossza.

A negyedik karakter a cél állomás azonosítója.

Az ötödik karakter a parancs karakter.

A következő karakterek az adott parancshoz tartozó paraméterek.

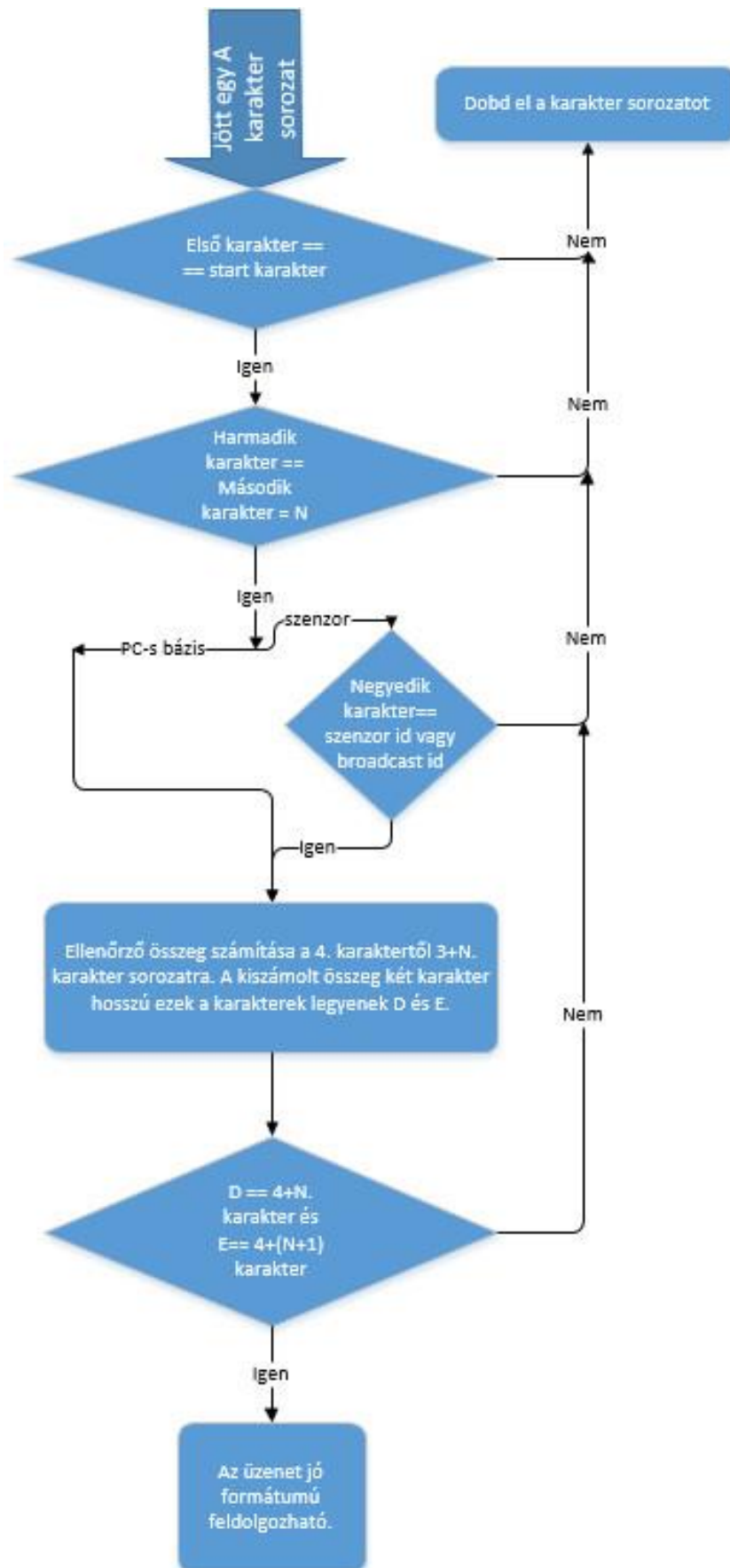
Az utolsó két karakter az üzenet hiba ellenőrző összege.

Az első három byte-ot kivéve az üzenet többi része a valós üzenet.

A szenzor válasza is ilyen formátumú kivéve, ha adatot kérek még a válaszban, ezen adatok az utolsó két karakter előtt helyezkednek el.

A parancsok és a válaszok maximális hossza nem haladhatja meg a maximális üzenethossz tároló kapacitását, ezért használhatom az üzenet hosszát, mint egy protokoll paraméter.

Ha a bázis állomás küldi az üzenetet, akkor a parancshoz tartozó paraméterek, ha a mérőegység küldi az üzenetet, akkor a parancsra való válaszban levő adatok. (parancstól függ a hossza, 0 byte-tól 100 byte-ig) A kommunikációs protokoll állapotgépe:



Paraméterek

A mérés alaptulajdonságai:

- Mintavételi frekvencia (gyorsulás, mágneses orientáció, szögsebesség): 1600Hz, 760Hz, 100Hz
- Memória kiolvasási idő: 2-3 perc
- A mérés ideje: 2-5 perc

Kommunikáció:

- Adatsebesség: 86,2 kbps
- Sávszélesség: 270kHz
- Vivőfrekvencia: 434.05MHz
- Frekvencialöklet: 135kHz

Módosítások

A rendszer általános paramétereinek, funkcióinak változtatása

A bázisállomást egy intelligens soros port-ként üzemeltettem a számítógép és a mozgó egységek között. A számítógép által USB-n adott adatokat a rádión továbbítja, a rádión kapott adatokat szűri majd továbbítja az USB-n a számítógépnek. A protokollt újraterveztem a feladatnak megfelelően.

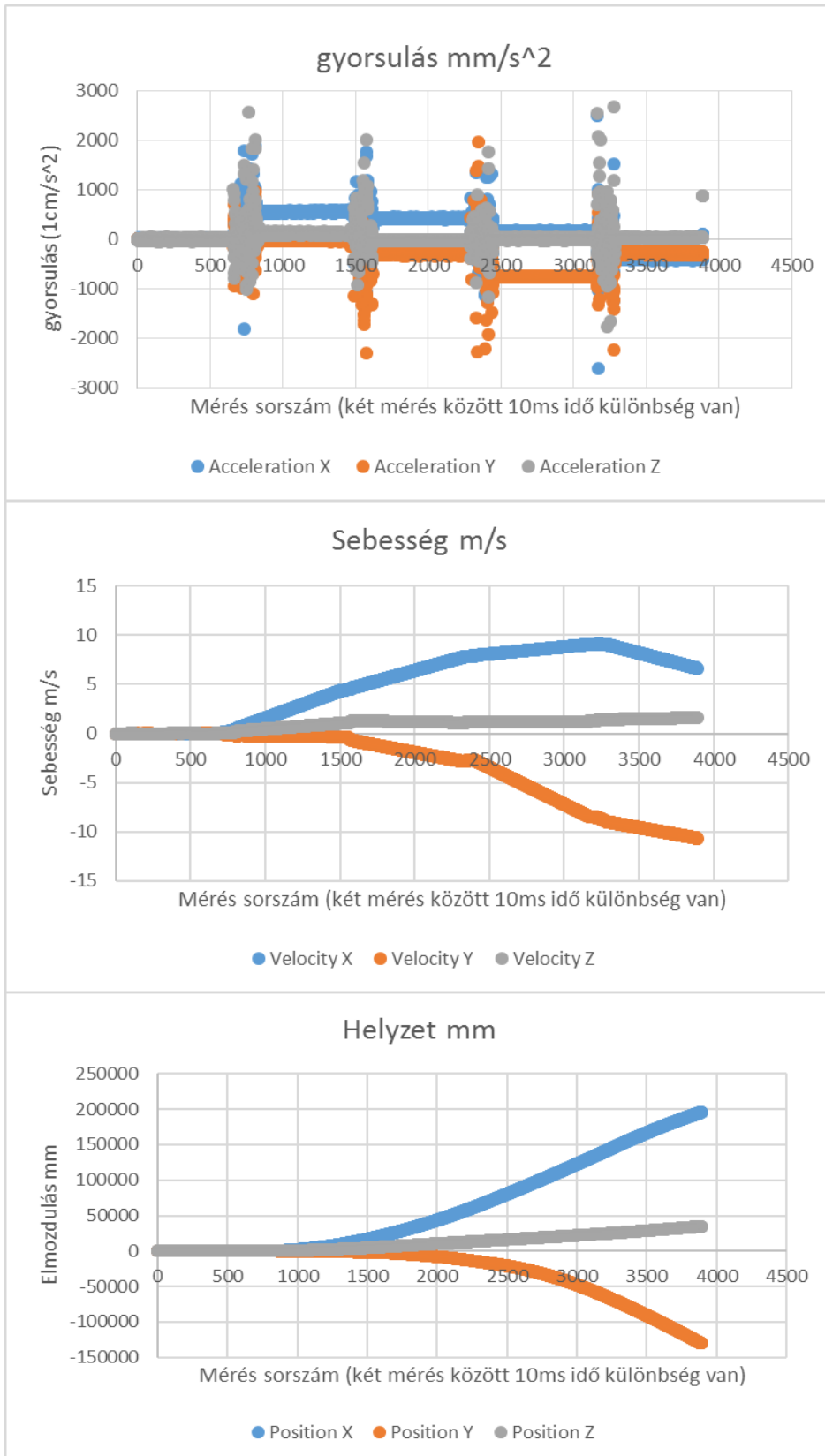
A mozgóállomások beállítható mintavételi időközönként, három dimenzióban mérik a gyorsulást, a mágneses orientációt és a szögsebességet. A mért adatokat a saját EEPROM memóriájában eltárolja, egy részét valós időben továbbítja a rádión, a működés végén pedig az összes memóriában tárolt adatot továbbítja a számítógépnek a bázisállomáson keresztül.

Az összes mérést három dimenzióban végzi és minden adat 16 bites. A szenzort a gyorsulás méréséhez 1600 Hz-es, a mágneses orientáció méréséhez 100 Hz-es, a szögsebesség méréséhez pedig 760 Hz-es mérési frekvenciára állítottam. A mért adatokat egy adott regiszter csoportba teszi, majd az új mérésnél ugyanebbe a regiszter csoportba teszi az adatokat, a régieket felülírja. A PIC18F24K22 mikrokontroller egy parancsként lekéri a szenzortól az összes aktuális adatot.

Ez a megoldás azt jelenti, hogy egy szenzor három különböző paraméterének (gyorsulása, mágneses orientáció, szögsebesség) mérése biztos, hogy nem egy időpillanatban történt. Minden egyes mérés után a regiszterek felülírása előtt egy időintervallumon belül lesz a tényleges adat mérése, 10ms-os mintavételi frekvenciánál a gyorsulás $1/1600\text{Hz} = 0,625\text{ms}$ -os, a mágneses orientáció mérésnél $1/100\text{Hz} = 10\text{ms}$ -os és a szögsebességnél $1/760\text{Hz} = 1,316\text{ms}$ -os időintervallumon belül lesz a mérés. Egy szenzornál az összes adat pontos mérése még könnyen kivitelezhető, de több szenzornál ez nem egy egyszerű feladat, ennek megoldásával nem foglalkoztam. Így egy mérés három paraméterének 10ms-os időpillanaton belül történik meg a mérése. A 10ms-os mintavételi időnél még a mért adatok pont függetlenek egymástól, de nagy a hibájuk. Ekkora hibánál a szenzor önmagában nem használható mozgás leírására, mert ekkor a feldolgozásnál egy parabolához hasonló pályát kapunk (a hiba véletlenszerű, így annak négyzetével arányos integrálása egy parabola)

A szenzorral végzett vizsgálatok

Egy forgó mozgást végzett a szenzor, melynél mindig egy adott helyen állt egy ideig (a gyorsulás ábráján a vízintes részek ezeket a részeket mutatják.) A helyzetnél a közel parabola pályát lehet látni mindhárom tengelyen. Alábbiakban láthatjuk a kapott grafikonokat:



A rádiós kommunikáció üzenetének feldolgozási idejének gyorsítása és zajszűrése

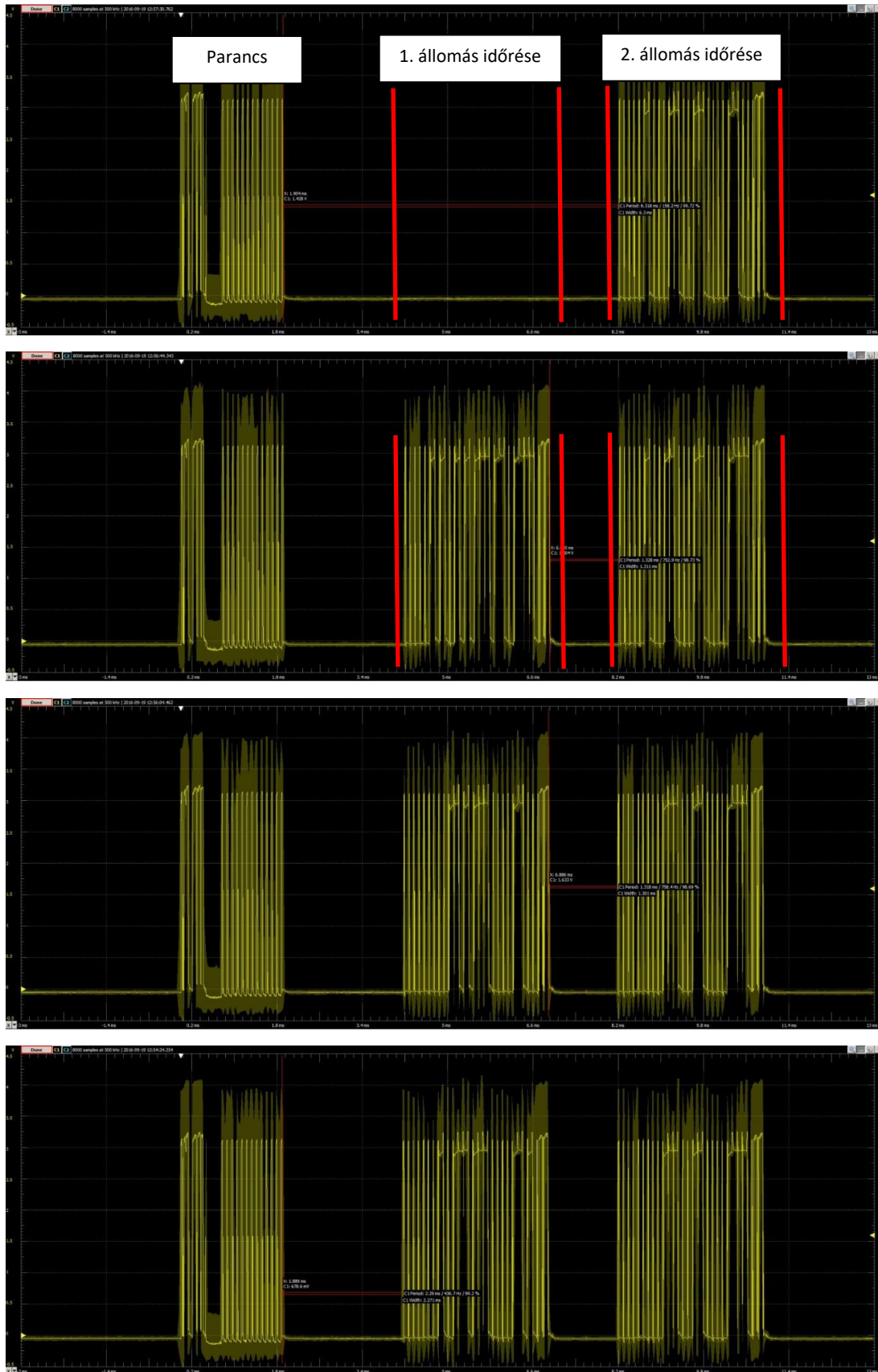
A rádió kommunikáció adatátviteli sebességét növeltem 1,2kbit/s-ról a gyakorlatban elérhető legnagyobb átviteli sebességre 86 kbit/sec –re, ehhez a rádiós modul egyes beállításain és az üzenet szoftveres feldolgozásán kellett változtatnom. A rádiós modul adatsebességét 86kbit/sec-re és a frekvencia modulált jel frekvencia löketét 90kHz-ről 135kHz-re növeltem. Egy rádión kapott karakter feldolgozás diagramja:



Ilyen feldolgozási láncot valósítottak meg abban a projektben, amelyből kiindultam [1]. A struktúrán egyáltalán nem változtattam, de az egyes részek idejét csökkentettem. Az első két rész idején nem lehetett olyan változtatást végezni, mely csökkentette volna a karakter feldolgozási idejét. A végtelen ciklus futási idejét lehetett csökkenteni. A kiindulási projektben sok olyan funkciót építettek be a szenzor szoftverébe, melyet a szenzor minden-egyes ciklusban megvizsgált (például az alacsony fogyasztáson való működést, a reset utáni állapotot vagy éppen az alacsony fogyasztáson való működés utáni inicializálási állapotot) ez sok feltétel megvizsgálását vonta maga után, melyeket már utána nem is használtam, de sok idő elment vele. A kapott karakter feldolgozásánál is egy kétszintű állapotgép volt kiépítve, melyet leegyszerűsítettem, átalakítottam, egyszintű állapotgépet használtam. Így próbáltam csökkenteni a karakter feldolgozási időt, melyet kellett is, mert egyébként azt tapasztaltam, hogy az első karaktert veszi, de az utána levőket már nem megfelelően észleli (két karakter is jött a feldolgozási idő alatt és a második felülírta az elsőt.) Így az üzenetet az ellenőrzésnél hibásnak találta a szenzor és nem foglalkozott vele.

Válasz-időzítések egységesítése

Az üzenet gyorsításához a szenzorok időzítéseit egymáshoz kellett szinkronizálnom. A szenzorok időzítését 1 ms-os timer interrupt-ok adták meg. A fogadott üzenetre a választ egy adott idő múlva küldtem, mert ha csak a szenzorok időzítését veszem figyelembe, hogy mikor menjen a válasz, akkor ± 1 ms-os hibával megy ki a válasz. Tehát ha az üzenet továbbítása 3,5 ms-ig tart és az előző üzenet előtt várnom kell legalább 0,5 ms-ot (a soros port két üzenet észlelése között legalább kell egy ekkora időkeret) akkor a 4 ms-os időkeret helyett egy 6 ms-os időkeretet kell biztosítanom, hogy az üzenetek ne ütközzenek egymásba. Ez egy 50%-kal nagyobb időkeret, amely már nem lett volna elegendő a megfelelő adatmennyiség továbbítására. Ehhez a szenzorok óráját egymáshoz kellett szinkronizálnom. Egy lehetséges megoldás, ha van egy start üzenet a mérés elején, amelynél az eszköz elkezdi az adatokat a szenzorból lekérni és az üzenet szinkronizáló óráját mindig nulla értékre állítja. Addig ismétlődik ez a start üzenet, míg minden szenzor nem nyugtázta a parancs vételét a neki megfelelő időrásben. Ha akárcsak egy mérőegység is elmulasztja a nyugtát, akkor az újra küldött start üzenettel az összes többi egységet is újraindítjuk. Így biztosítható, hogy egy időpillanatban minden szenzor időzítése azonos állapotban lesz (majdnem órajel pontosan).



6. ábra, Oszilloszkóp ábrasorozat. Az oszcilloszkópot a bázis állomás I²C buszára kötöttem és az ezen folyó kommunikációt vizsgáltam, ahol az első nem nulla csomag rész a parancs és a többi nem nulla csomag az erre küldött válasz.

Mint a fenti ábrákból is látszik ez megoldotta a problémát.

Az ábrasorozaton érdemes megfigyelni, hogy az alsó három esetben a kommunikáció mindig ugyanazon időpontok között zajlik, az időzítések bizonytalansága jóval kisebb, mint a keretek között hagyott idő (inter-frame gap). A mérésnél az órajelek hibáját nem vettem figyelembe, de ez egy maximum öt perces mérésnél nem akkora probléma, hogy a mért adatok teljesen szétcsússzanak. Az eszközben használt kvarckristály hibája 10ppm, ez öt perc alatt egy $5 \cdot 60 \cdot 1000 \cdot 10^{-5} = 3\text{ms}$ -os hibát fog eredményezni, ez a többi hibához képest nem nagy hiba. A [3] szakirodalom ismertet egy szinkronizáló eljárást mely az egyes szenzor órajeleit egymáshoz szinkronizálta volna, de ezt nem végeztem el, mert az alkalmazás során nem szükséges.

Mérések, eredmények

A rendszer önmagában nem használható mozgás leírására, mert még nem kapcsoltam össze egy optikai rendszerrel. Készítettem egy segédprogramot, mely a mért adatokat feldolgozza. A feldolgozásnál az adatokat két részre bontottam.

Adatfeldolgozás

Az első rész az első száz adatból állt. Minden mérést úgy végeztem, hogy a szenzorok nyugalomban voltak és elindítás után még legalább egy 1 másodpercig a szenzorokat nyugalmi helyzetükben hagytam. Ezeknek az adatoknak a gyorsulás és mágneses orientáció komponensét kiátlagoltam, így kaptam az eredő gyorsulást (a gravitációs gyorsulást) és az eredő mágneses orientáció vektort, mely észak felé mutat. Ezek a kiátlagolt adatok, referencia adatoknak fognak szolgálni a többi adat feldolgozásánál.

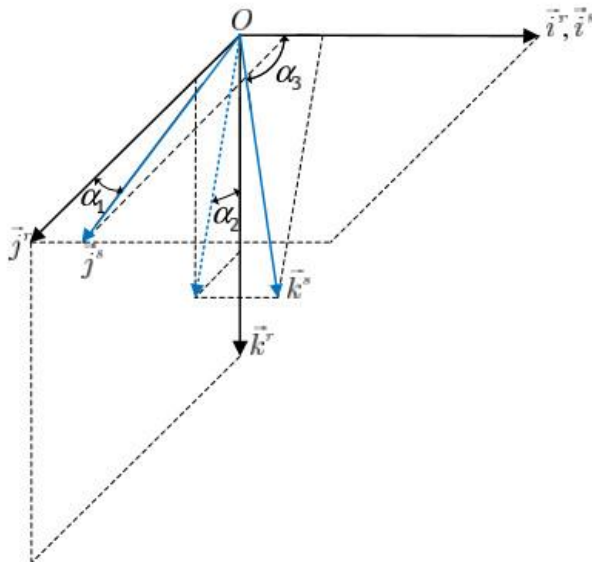
A második résznél az összes többi adatot a referencia adatok segítségével transzformáltam, az egyes adatok koordináta rendszerét egy egységes koordináta rendszerbe és az adatokat (gyorsulás adatokat) olyan formára mely közvetlenül használható a mozgás leírásra.

Adatok transzformálása

A szakirodalomban fellelhető módszer

A [4]-es forrás egy négy lépésből álló transzformálást mutat be mely a szenzor mért adatait a szenzor saját mozgó koordináta rendszeréből áttranszformálja egy földhöz képest fix koordináta rendszerbe. A szenzor esetükben egy három forgás szabadságfokú géphez volt szerelve. Első lépésnél a szenzor nem ortogonális koordináta rendszerét egy szenzorhoz fix ortogonális rendszeré transzformálta át. Ehhez meghatározta az összes mérőműszer tengelyeinek egymáshoz képesti szögeit (3db-ot) melyből létrehozta a transzformációs mátrixot.

$$\mathbf{T}_r^s = \begin{bmatrix} 1 & 0 & 0 \\ \sin(\alpha_1) & \cos(\alpha_1) & 0 \\ \cos(\alpha_3) & \sin(\alpha_2) \sin(\alpha_3) & \cos(\alpha_2) \sin(\alpha_3) \end{bmatrix}$$



7. ábra, A szenzor nem ortogonális rendszerből ortogonális rendszerbe való transzformálása [4]

Transzformációs mátrix: \mathbf{T}_r^s

Az összes többi rendszer mind egy három dimenziós forgatás hisz mindegyik rendszer egy 3D-s ortogonális rendszer.

Második lépésben a szenzor ortogonális rendszerét az őt mozgató szerkezet szenzorral együtt mozgó részénél egy fix koordináta rendszerbe transzformálta. Ezt a forgatást három forgatásra bontották fel, melyet mind a három tengely körül végeztek el.

Transzformációs mátrix: \mathbf{C}_q^r

Harmadik lépésben egy, a szerkezet mozgó részéhez fix koordináta rendszert egy a szobában fix koordináta rendszerbe transzformálta át. Ezt a forgatást is három forgatásra bontották fel melyet a három tengely körül végeztek el.

Transzformációs mátrix: \mathbf{C}_p^q

Negyedik lépésben a szobához fix koordináta rendszert egy észak-kelet-gravitációs gyorsulás által kifeszített ortogonális rendszerbe transzformálta át. Az előző koordináta rendszerben a z tengely a gravitációs gyorsulás irányába mutat, ezért csak egy két dimenziós forgatást kell elvégezni ebben az esetben.

$$\mathbf{C}_{\text{NED}}^p = \begin{bmatrix} \cos \beta & \sin \beta & 0 \\ -\sin \beta & \cos \beta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Transzformációs mátrix: C_{NED}^p

Az ismeretlen paramétereket a Levenberg-Marquardt algoritmus alapján határozták meg.

Ismeretlen paraméterek:

- skálázási hiba a mátrix három komponense
- nem ortogonális koordináta rendszer tengelyeinek egymáshoz képesti szögeit 3db
- az utolsó forgatásnál a forgatási szög (C_{NED}^p - nél a forgatási szög)
- három tengely, rendszer hiba komponense

A mért gyorsulási adatokkal a következő transzformációt végezzük (mátrixokkal szorozzuk).

$\vec{a}_m = (I + S) T_r^s C_q^r C_p^q C_{NED}^p \vec{a}^{NED} + \vec{b} + \vec{n}$. mért adatok ahol a_m a mért adat, a^{NED} a földhöz viszonyított rendszerben a mért adatok, b az ezen felüli rendszer hiba, n pedig a cikk által használt szenzor modell egyik paramétere (szenzor modell $\vec{e}_m = \vec{h}(\vec{e}, \theta) + \vec{n}$), S mátrix pedig skálázási hiba mátrix. Minden mérési adatnál az a_m függvényében az a^{NED} -t meghatározzuk.

A mágneses orientáció mérésnél ugyanezen transzformációs lépéseket hajtjuk végre (gyorsulási adatok feldolgozásánál levő transzformációs lépéseket hajtjuk végre). A szögsebesség adatok ugyanazon módszerrel transzformálhatók át, mint a gyorsulási adatok.

A használt transzformálás

A fent részletezett modellt egyszerűsítettem és az egyszerűsített modellt használtam. Először az összes adat hibáját kompenzáltam (később bővebben kifejtem). A szenzor mért adatait azonnal egy rögzített koordináta rendszerbe transzformáltam ezt pedig a kiinduló állapotban levő szenzor koordináta rendszerébe. A transzformáláshoz meg kell határoznom az aktuális mágneses orientáció vektor és a kezdeti állapot mágneses orientáció vektor szögét α -t (α egy vektor). Majd a szögsebesség és gyorsulás vektorokat elforgatom α -val. Ezután az áttranszformált eredő gyorsulásból (vektoriálisan) kivontam az elsőként meghatározott gravitációs gyorsulást. Az adatok hibáira lineáris modellt használtam. (Az egyes tengelyeknek van ofszet és erősítési hibájuk.) Először az ofszet majd az erősítési hibát kompenzáltam.

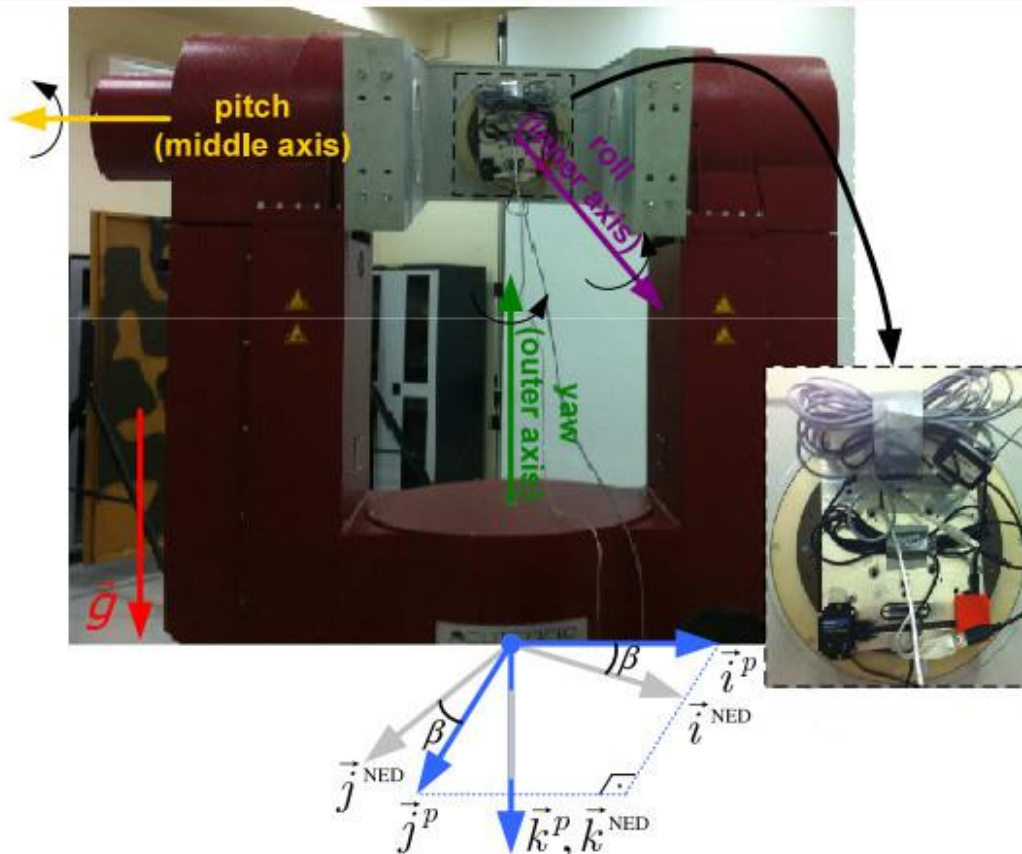
A szenzor tengelyek hibáinak meghatározása:

A szakirodalom [4] egy hét lépéses mérést mutat be, a nyert adatokból és a Levenberg-Marquardt algoritmus alapján a keresett adatokat meghatározza.

A mérést egy ortogonális három forgás szabadságfokú eszközzel végezték. A középső forgás tengelye vízszintes, a külső függőleges, a belső pedig a középső tengely aktuális helyzetéhez képest sugár irányú. A szenzor a belső tengelyen úgy van rögzítve, hogy a szenzor legnagyobb oldalának, síkjának normális egyenese merőleges legyen a belső forgástengellyel.

A mérés hét lépése:

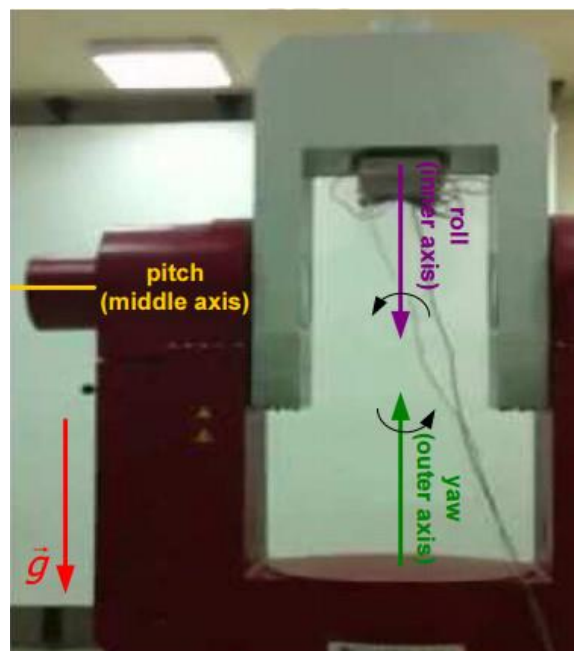
1. A szenzort elhelyezzük az egyik függőleges állapotába.



8. ábra, Szenzor kezdeti helyzete[4]

2. A szenzort a belső tengelye körül elforgatjuk 270°-kal 22,5°-os lépésenként minden helyzetben 12,5 másodpercet várunk.

3. A szenzor a következő helyzetben lesz



9. ábra, A szenzor 3. lépésbeli helyzete[4]

4. Ezután a középső tengelye körül elforgatjuk 180° -kal $22,5^\circ$ -os lépésenként minden helyzetben 12,5 másodpercet várunk.

5. A 3-as résznél levő pozícióba forgatjuk vissza a szenzort

6. A belső tengely körül elforgatjuk a szenzort 90° -kal

7. A 4. lépést megismételjük.

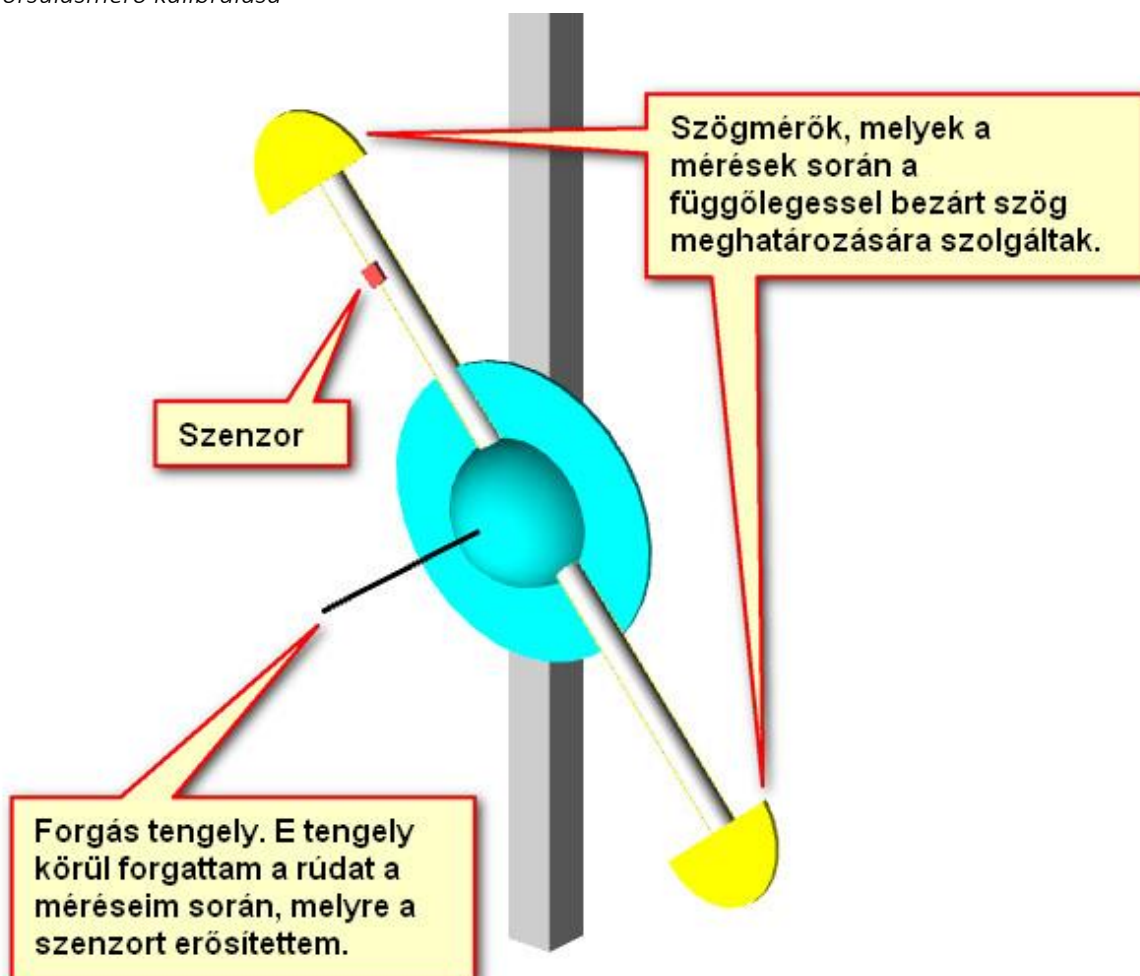
A Levenberg-Marquardt algoritmus egy nemlineáris, legkisebb négyzetek módszerét felhasználó görbe illesztő eljárás. Részletesebb leírást az algoritmusról a [4]-es, [8]-as forrásban lehet találni. Miután a szenzor hibáira egy differenciálisan lineáris modellt használtam (offset és erősítési hiba) ezért a Levenberg-Marquardt algoritmust nem használtam.

Használt módszerek

A használt eszköz egy forgás szabadságfokú eszköz volt melyre viszont a szenzort bármilyen 3D-es helyzetben tudtam rögzíteni. A forgás tengelye jó közelítéssel vízszintes volt.

Először a gyorsulás mérőt próbáltam kalibrálni, mert van egy konstans statikus gerjesztése (a gravitációs gyorsulás) melynek értékét is tudom ($9,81 \text{ m/s}^2$). Így egy abszolút hibát tudok mérni az adott tengelyekre nem pedig egymáshoz képesti relatív hibát.

Gyorsulásmérő kalibrálása



10. ábra, A mérési elrendezés

Első variáció: Egy egyszerű eljárással próbáltam meghatározni az egyes tengelyek hibáit. Először megmértem az egyik szélső helyzetben az eredő gyorsulást (legyen ez A, hiba nélkül ideális esetben A_{id}) majd a másik szélső helyzetben (legyen ez B, hiba nélkül ideális esetben B_{id}) $A_{id}=B_{id}$. Az adódó ofszet hiba $(A+B)/2/A_{id}$ az erősítési hiba pedig $1-(A/2+B/2)/A_{id}$.

A mérés során mindig az adott pozícióban mérek legalább 5000 adatot, folyamatos mintavételezéssel (bármely két adat független, és miután 5000 adatot átlagolok az átlag eloszlása normális eloszlású). Az egyik tengely, a gyorsulás tengely ofszet hibájának meghatározására elvégeztem a mérést háromszor és a következő adatokat kaptam.

Mérés	Mértékegység ($0,61\text{mm/s}^2$)	Átlag	Szórás	Ofszet	Ofszet valószínűségi hiba (egyenlő érzékenységgű a két adat)
1	min	-16627	286	-135.5	509
	max	16356	422		
2	min	-16970	165	-346	231
	max	16278	163		
3	min	-16617	289	-109.5	424
	max	16398	311		

A mérést akkor tartottam elfogadhatónak, ha a kapott ofszet hiba nincs benne a 0 saját ofszet hibájának/szórásának környezetében. Ez csak a második mérésnél teljesült és ott is ($346/231 \approx 1,5$ szórásra van a 0-tól) csak $\sim 86\%$ -os konfidenciával mondható, hogy ofszetet mértem. Az ofszet hiba meghatározása további vizsgálatokat igényel.

Hibák a módszernél:

- a szenzort nem tudtam pontosan olyan helyzetbe hozni, amely a szélsőértéket venné fel.
- a rögzítés adott helyzetében a mérési elrendezésből adódóan legalább $\pm 1^\circ$ elmozdulás volt a kezdeti helyzettől

Második variáció: Az első próbálkozás egy javított verziója. A szenzort a forgástengely közelébe tettem majd lassan ($T = 1$ perc periódus idővel) elforgattam a tengely körül egyszer teljesen, majd a mért értékek közül a legnagyobbat és legkisebbet vettem, mint feldolgozandó adatot. Így próbáltam az egyes adatok hibáját az ofszet számolásánál eltüntetni, mert ekkor a legkisebb adatnál negatívan, míg a legnagyobb adatnál pozitívan adom hozzá a hibát, és így az átlagnál pont kiejtik egymást.

Öt mérést végeztem, a mérési eredmények:

Mérés	Mértékegység (0,61mm/s ²)	x		y		z	
		mért érték	ofszet	mért érték	ofszet	mért érték	ofszet
1	max	17128	-288	18113	-183	9908	4487
	min	-17416		-18296		-5421	
2	max	16893	-304	15416	-830	9552	2735
	min	-17197		-16246		-6817	
3	max	16913	-111	15727	-1462	9795	-9840
	min	-17024		-17189		-19635	
4	max	17251	668	16624	-232	14768	5013
	min	-16583		-16856		-9755	
5	max	17221	58	16784	-149	8283	1931
	min	-17163		-16933		-6352	

A mérésből adódóan a minimum és maximum abszolút értékek egy közös értéktől legfeljebb 10%-ban térhetnek el, mert ha nagyobb mértékben térnek el, akkor a mérést nem egy síkban végeztem, vagy valamilyen kisebb rezgésből eredő hatás miatt, esetleg a forgatásban levő megugrás okozott ilyen változást. Emiatt a z tengely ofszet hibájának meghatározása lehetetlenné vált (a kapott ofszet egy nagyságrendben van a mért értékekkel (melyek nem nulla közeli értékek) és ez irreális). A másik két tengelyen számolt ofszet hibák átlagával ugyanaz a probléma merül fel, mint az első módszernél. (A kapott ofszet hiba benne van a 0 saját ofszet hibájának környezetében.)

Mértékegység (0,61mm/s ²)	x	y
ofszet átlag	3.833333	-476
ofszet szórás	356.982	543.2326

Az ofszet hiba meghatározása további vizsgálatokat igényel.

Hibák a módszernél:

- A szenzor forgási síkjától legalább 2 fokkal eltért
- forgás során érték olyan rezgések, amelyek a szélső értékek meghatározásánál nagy hibát okoztak.
- nagyon érzékeny egy-egy kiugró mérési hiba értékre (emiatt nem vonható le következtetés a z tengely ofszet hibájára)

Harmadik variáció: A szakirodalombeli mérést [4] egyszerűsítettem, nem használtam komplex modellt a hibákra és némely szakirodalomból említett hibát el is hanyagoltam (például a szenzor tengelyeit ortogonálisnak tekintettem). A szenzort úgy helyeztem el ($\pm 1^\circ$ fokos pontossággal) hogy függőleges helyzetében csak egy tengelyen érzékelje a gravitációs gyorsulást és ezt várjuk el vízszintes helyzetben is. Majd 15° -os lépésként egyszer teljesen körbe fordul, ahol minden egyes

álló helyzetben legalább annyi ideig marad (10 másodperc), amennyi idő alatt 1000 adatot tud mérni. Majd ugyanezt a mozgást elvégeztem úgy is, hogy az eredeti helyzetéhez képest egy olyan merőleges helyzetben legyen, hogy az előző fordulatnál kimaradt tengelyen, a mérés során legyen olyan állapot, amikor rajta mérek gyorsulást, míg a többi tengelyen nem.

Az egyes tengelyeken mért értékeket a következő kifejezéssel modelleztem:

$$f(x) = a * x + b$$

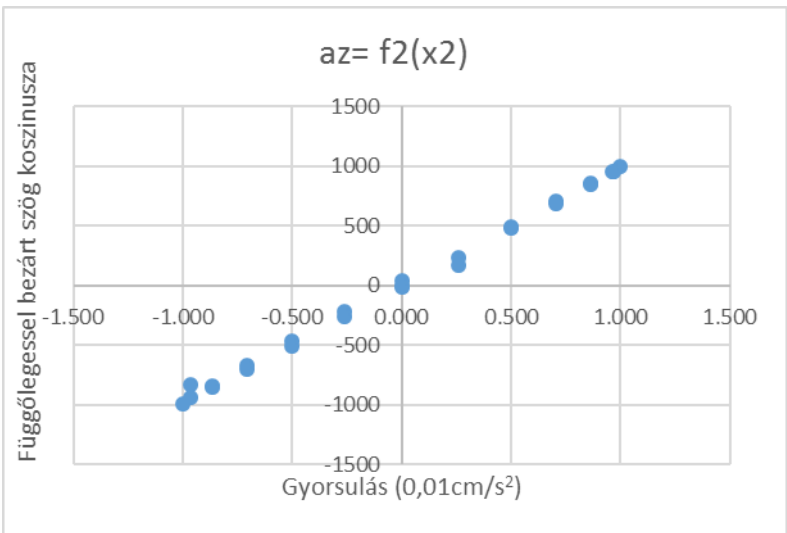
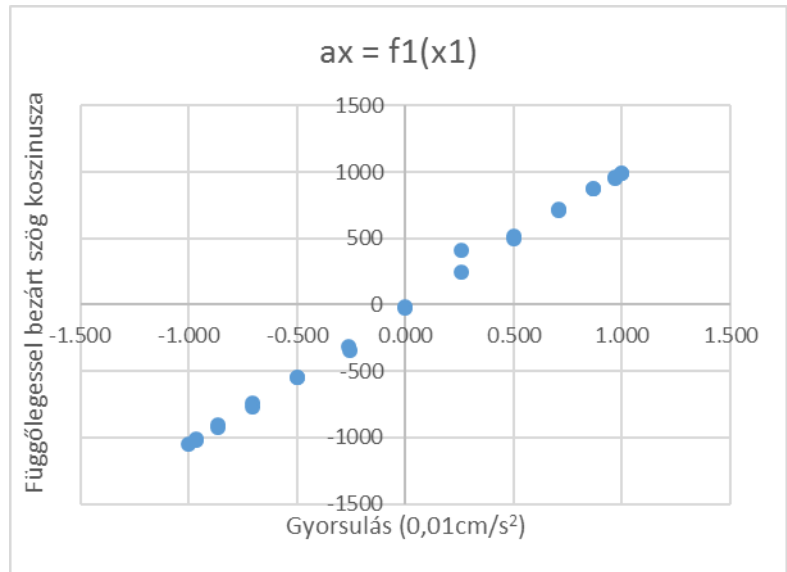
A mért adatok alapján a legkisebb négyzetek módszerét alkalmazva, az a és b paraméterek a következőknek adódnak: (levezetés a [9]-es irodalomban)

$$a = \frac{N * \sum(x \cdot y) - \sum(x) * \sum(y)}{N * \sum[(x)^2] - \sum(y) * \sum(y)} \quad b = \frac{\sum y - a * \sum x}{N}$$

A mért adatok nagyságrendje a gravitációs gyorsulás ezredrésze (9.81mm/s^2)

1.Mérés adatai és grafikonjai:

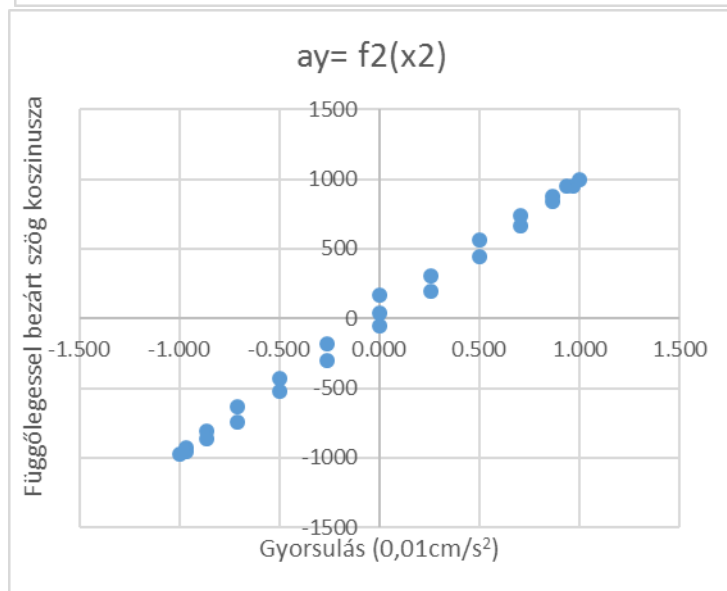
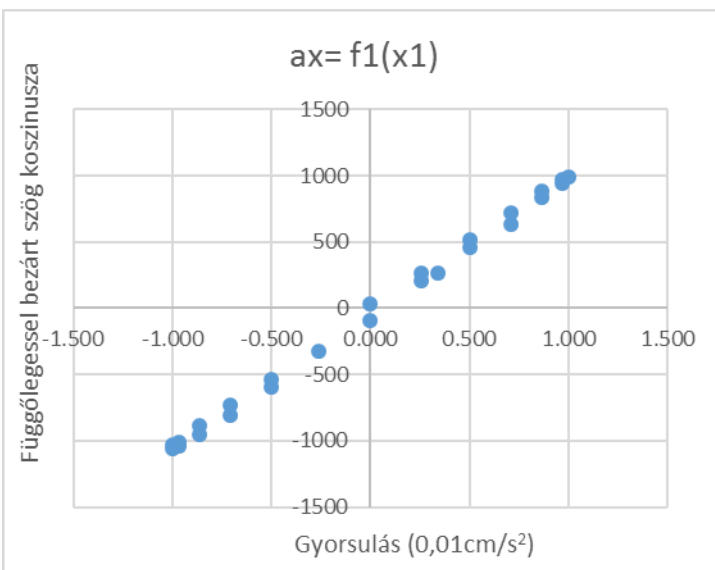
x_1	$a_x = f_1(x_1)$	x_2	$a_z = f_2(x_2)$
1.000	990	0.000	43
0.966	965	-0.259	-222
0.866	871	-0.500	-465
0.707	716	-0.707	-674
0.500	496	-0.866	-845
0.259	408	-0.966	-829
0.000	-17	-1.000	-990
-0.259	-344	-0.966	-938
-0.500	-550	-0.866	-853
-0.707	-745	-0.707	-697
-0.866	-903	-0.500	-506
-0.966	-1010	-0.259	-263
-1.000	-1046	0.000	-4
-0.966	-1023	0.259	237
-0.866	-922	0.500	485
-0.707	-767	0.707	686
-0.500	-547	0.866	856
-0.259	-317	0.966	955
0.000	-22	1.000	995
0.259	246	0.966	959
0.500	512	0.866	850
0.707	707	0.707	705
0.866	870	0.500	495
0.966	952	0.259	165
1.000	994	0.000	-10



N=25	a_x	a_z
Ex^2	12.99965	12.00035
Exy	13403.16	11645.42
Ex	0.999296	0.000185
Ey	514.824	135.5568
a	1032	970
b	-20	5

2.Mérés adatai és grafikonjai

x_1	$a_x = f_1(x_1)$	x_2	$a_y = f_2(x_2)$
-1.000	-1063	0.000	38
-0.966	-1041	-0.259	-181
-0.866	-951	-0.500	-428
-0.707	-807	-0.707	-633
-0.500	-595	-0.866	-812
-0.259	-321	-0.966	-931
0.000	-92	-1.000	-977
0.259	209	-0.966	-951
0.500	457	-0.866	-864
0.707	635	-0.707	-747
0.866	836	-0.500	-525
0.966	946	-0.259	-298
1.000	991	0.000	-54
0.966	969	0.259	191
0.866	880	0.500	445
0.707	724	0.707	667
0.500	518	0.866	835
0.259	266	0.966	949
0.000	38	1.000	992
0.342	268	0.940	950
-0.500	-533	0.866	875
-0.707	-727	0.707	741
-0.866	-883	0.500	561
-0.966	-1009	0.259	307
-1.000	-1032	0.000	167



N=25	a_x	a_y
E_{x^2}	13.04985	11.94999
E_{xy}	13386.09	-11733.7
E_x	-0.39848	0.026149
E_y	-1320.34	316.1949
a	1024	-982
b	-36	14

Végleges eredmény:

	a_x	a_z	a_z
a	1029	982	970
b	-29	-14	5

A mérés végeredményeként a szenzor gyorsulás mérő három tengelyére kaptam egy ofszet és egy erősítési értéket (elméletben az erősítés 1000, míg az ofszet 0 mindegyik tengelyre). A kiszámolt egyenesre vett szórás, mindegyik értékre legalább 30 (1 mérés szórásai a_x 36 a_z 32, a 2 mérés szórásai a_x 35 a_z 51). Ha végértékre nézem ezt a hibát, akkor az összes kiszámolt (erősítés és ofszet) elméleti érték szórás körüli környezetén belül van. Tehát ha csak ezt veszem figyelembe, akkor ezek a mérési eredmények is rosszak.

A mért eredmények a tényleges erősítésekhez és ofszetekhez közel állnak, a melyet a következőkben indokolok meg. Először is később mutatok egy olyan mozgást, amelyet mértem és először az elméletbeli hibamentességgel dolgoztam fel, majd a korrigálandó értékekkel javítottam és dolgoztam fel. Így a feldolgozott adatok jobban leírták a mozgást. Tehát a korrigálással jobb kiértékelést értem el, mint anélkül. Mindhárom tengelyen a mért értékek azt mutatják, hogy a becsült hibák a tényleges hiba értékhez közelebb vannak, mint a hibamentes állapotban. Másodszor a gyorsulás mérő x tengelyének hibáját mindkét mérésnél kiszámoltam és a becsült hibák mindkét esetben ugyanolyan irányban tértek el és közel ugyanolyan értékkel.

Megjegyzem, hogy az adatlapon a hibákat párszázalékosnak mondják, és ezen belül vannak a számolt értékek.

Hibák a módszernél:

-A szenzor forgási síkjától legalább 2 fokkal eltért (ez látszik is a grafikonokon, hogy nem egy egyenesen vannak a pontok, hanem inkább egy ellipszisen)

Mágneses orientáció mérő kalibrálása

A mágneses orientáció mérésnél nem volt egy olyan konstans gerjesztésem, mint a gyorsulás mérőnél a gravitációs gyorsulás. (Elméletben a Föld mágneses tere a felszínen 0,3-0,4 Gauss, méréseim során átlagosan rendre 0,5-0,6 Gausst mértem.) Abszolút hibát így nem tudtam mérni az eszközökre. De abszolút hibát nem is volt szükséges meghatároznom, hiszen számításaim során csak a mágneses orientáció irányát használtam (ennek meghatározásához csak a tengelyeken lévő nagyságok arányai számítanak). Ezért a tengelyek egymáshoz viszonyított hiba meghatározása is elég volt az irány korrigálásához.

A mérést az is nehezítette, hogy az eredő mágneses tér irányát nem tudtam pontosan meghatározni. Így nem tudtam olyan mérést végezni, hogy csak két tengelyen legyenek hasznos adatok, azt sem sikerült megállapítani, hogy a két tengelyen levő mért adatok milyen arányban vannak egymással (csak az adatok és a tengelyeik hibáinak, kompozíciójának arányát tudtam megállapítani.)

A gyorsulás mérő második mérési variációját alakítottam át. Ugyanúgy egy helyzetben egyszer teljesen körbe forgattam a szenzort és az egyik tengely minimum, maximum értékét lejegyeztem. Három egymásra merőleges helyzetben végeztem el a forgatást az egyes szenzor helyzetekben mindig csak egy tengely adatait használtam fel a vizsgálataim során. Három helyzetből adódóan mindhárom tengelyen egy ugyanolyan görbét kapunk (legalább is a minimum és maximum szempontból), ezt a három pár (minimum és maximum) adatot használok fel a vizsgálataim során.

Az egyes minimum és maximum adatokból meghatároztam az egyes tengelyek ofszetjét és meredekségét. Majd egymáshoz viszonyított erősítésüket határoztam meg és ezután az erősítésekkel arányosan két ofszet értéket korrigáltam.

A kapott eredmény:

x/y erősítés arány	y/z erősítés arány
1.44	0.95

x ofszet y-hoz képest	y ofszet	z ofszet y-hoz képest
-83.2812	148	-324.115

Amint említettem a szélsőérték közelítésnek megvan a hibája ezért ilyen nagy az ofszet és egymáshoz képesti erősítések hányadosa. De még mindig jobb modell volt ilyen nagy eltérésekkel számolni, mint hibamentességet feltételezni.

Jövőbeli fejlesztésnél a mágneses orientáció mérő hibájának meghatározására pontosabb mérést kell keresni. Például egy olyat, mint a gyorsulás mérőnek a harmadik variációja.

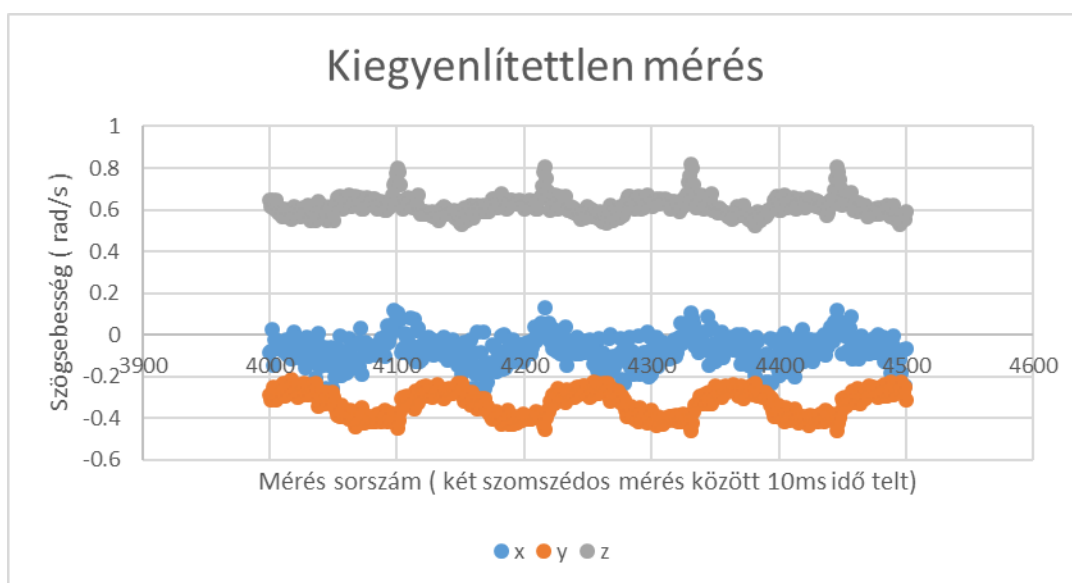
Giroszkóp kalibrálása

A [4]-es szakirodalom ajánlotta, hogy ha pontos műszerünk van, akkor a Föld forgásából számítsuk ki a giroszkóp egyes tengelyeinek hibáit. A Föld körülbelül 24 óra (23 óra 56 perc) alatt fordul meg a tengelye körül, átlagosan $360^\circ / (24 \cdot 60 \cdot 60) = 4,167 \cdot 10^{-4} \text{ }^\circ/\text{s} = 15 \text{ }''/\text{s}$ (szögmásodperc/ másodperc). A giroszkóp legkisebb kvantálási lépcsője $245 \text{ }''/\text{s}$. Tehát a Föld forgását a szenzor nem fogja érzékelni.

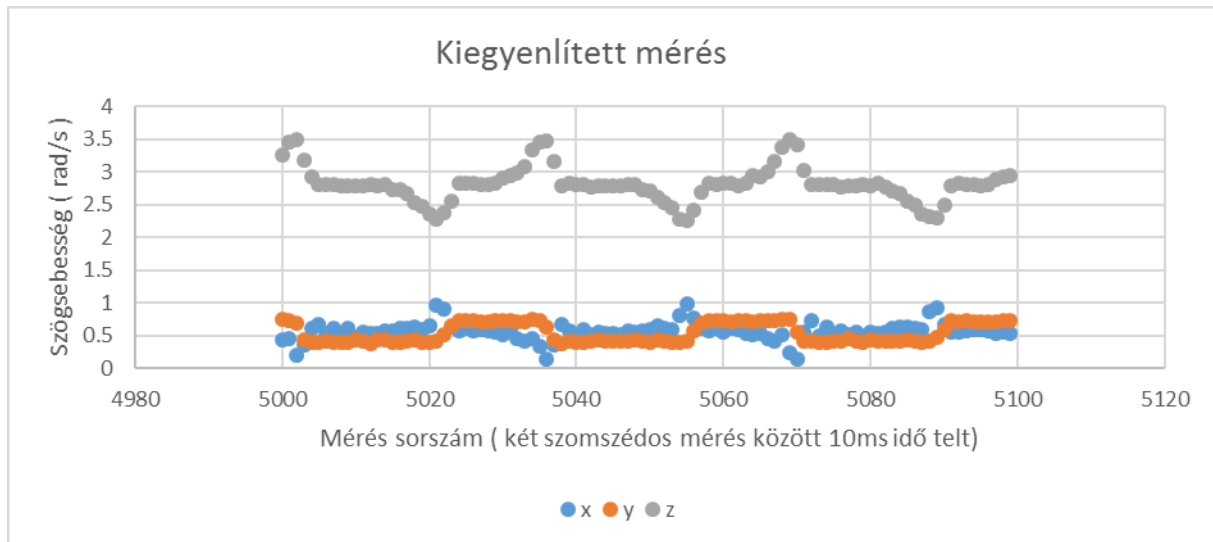
A következő mérésnél egyenletes sebességgel forgattam a szenzort (ez megvalósítható például egy csavarozó vagy fúró gép segítségével) és az így létrehozott szögsebesség segítségével próbáltam a giroszkóp hibáit megmérni.

A következő probléma merült fel ezzel kapcsolatban. Az eszközön a szenzor és tápja (két ceruzaelem) súlya miatt a forgásban egyenletlenség jött létre, melyet az eszköz kompenzálni próbált a szögsebesség változás függvényében.

Az így elvégzett mérés egy része:



Ezután a szenzort és tápját ellensúlyoztam és úgy végeztem el a mérést. A legjobban sikerült mérésnél sem tudtam jól kiegyenlíteni a forgó részt. Amint látszik a következő grafikonon.



Ezután megpróbáltam az egyes vízszintes szakaszokat figyelembe venni a kiértékelés során. Az egyes lineáris szakaszok átlagukat tekintve nem egyenlők (minden második viszont igen) ez az x és y görbénél jól megfigyelhető. Az így kapott átlagokból a nyert adatokra a differenciálisan lineáris modellt alkalmazva $f(x) = a \cdot x + b$ a legkisebb négyzetek módszerével elemeztem az adatokat. (dps = szögperc/ másodperc) x mértékegysége Hz, y-nok nagyságrendje 70dps

1. Mérés			
x	y1(w _x)	y2(w _y)	y3(w _z)
2.116667	846	6330	6486
0.75	334	2192	3178
1.5	627	4877	5719
0.45	143	1233	1605
N=4			
Ex ²	7.495	7.495	7.495
Exy	3047.087	22912.90	25412.98
Ex	4.817	4.817	4.817
E y	1950.857	14631.33	16987.41
a	412	3123	2924
b	-8	-103	725

2. Mérés			
x	y1(w _x)	y2(w _y)	y3(w _z)
0.466667	488	410	947
1	977	493	4906
1.85	1743	1283	8312
1.05	971	1109	4936
1.633333	1446	1742	7597
N=5			
Ex ²	8.411	8.411	8.411
Exy	7809.84	7068.63	38316.46
Ex	6.00	6.00	6.00
E y	5624.90	5037.54	26697.87
a	876	846	5187
b	74	-7	-885

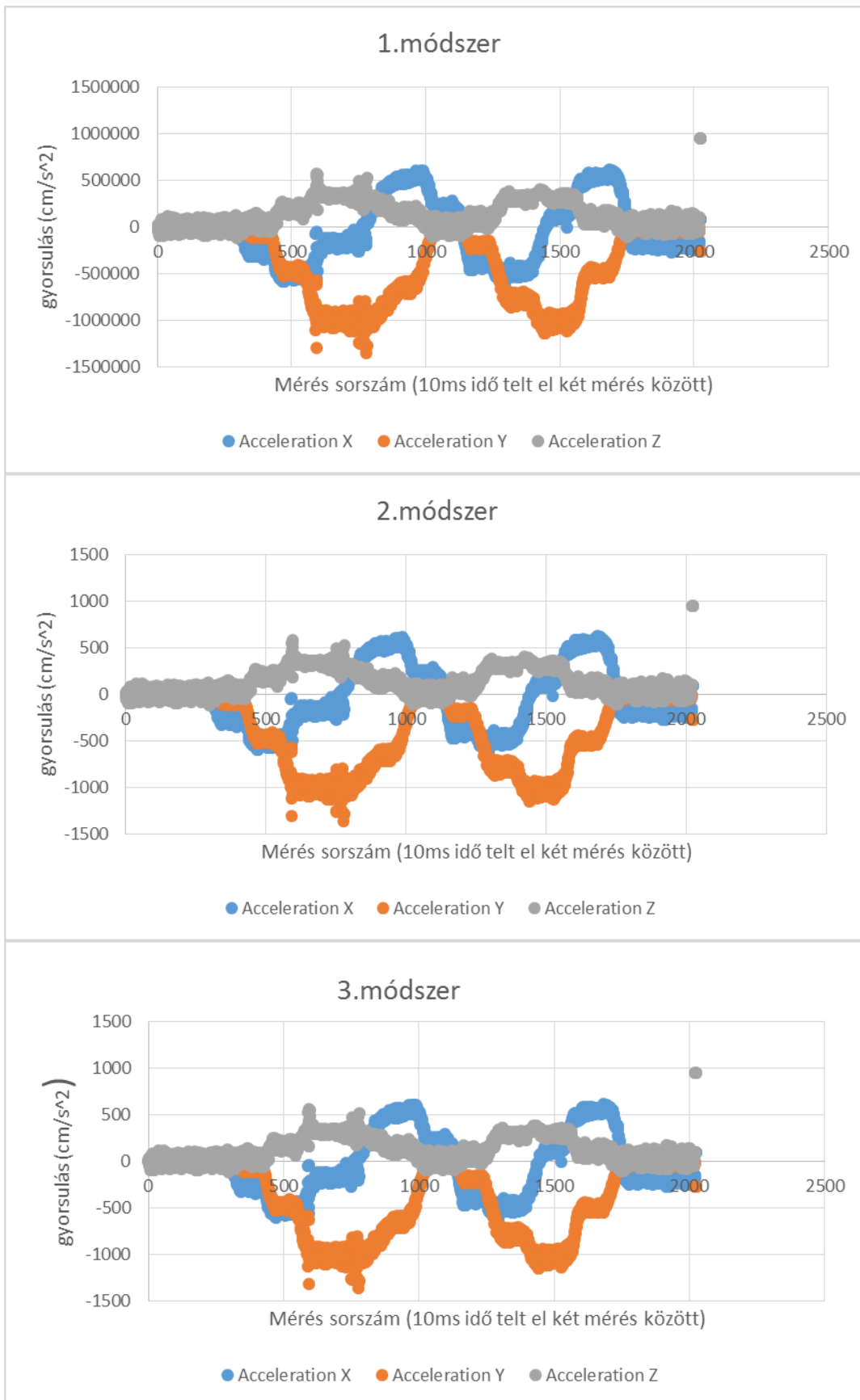
A gerjesztés (az eredő gerjesztés egyes tengelyekre vett vetülete) a két mérés során más volt és arányukat nem határoztam meg, ezért az a paraméterre nem vonható le semmilyen összefüggés, viszont a b paraméter kiszámolásához szükséges lenne az a paraméter. A b paraméternek pedig mindig egyenlőnek kell lennie az összes mérésnél. A z tengely ofszet hibáját vizsgálva az 1. mérésnél egy nagy pozitív értéket, míg a második mérésnél egy nagy negatív értéket kaptam. Ez túl nagy eltérés ahhoz, hogy következtetést vonjak le az ofszet hibákkal kapcsolatban.

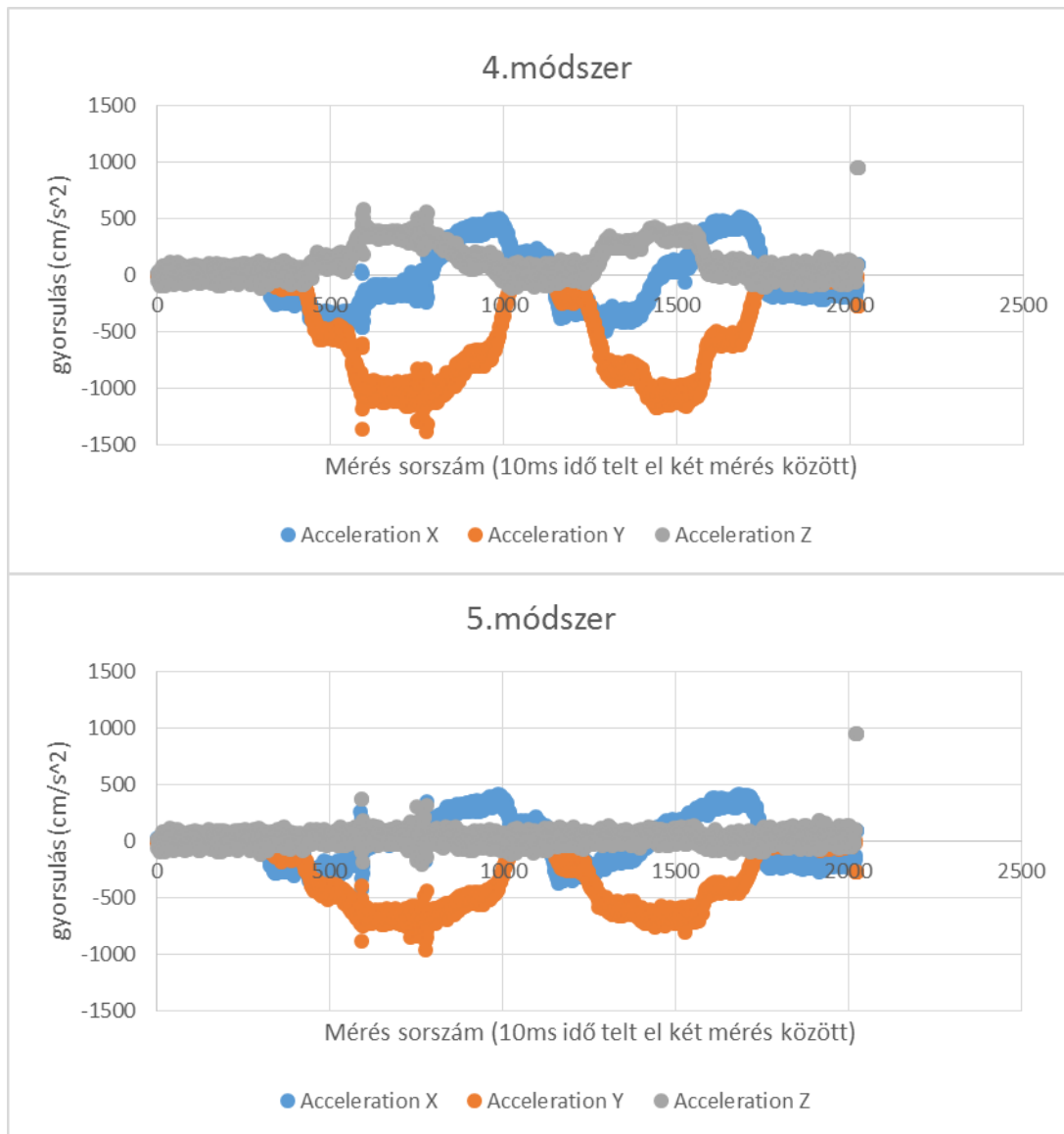
A giroszkóp hibájának meghatározásához további méréseket kell elvégezni.

A mérés eredményeinek kiértékelése

A mért adatok kiértékelésénél/feldolgozásánál öt módszert használtam. Ezek csak abban különböznek, hogy milyen hibákat vettem figyelembe. Az első módszernél az összes tengelyt hibátlanak feltételeztem. A másodiknál a gyorsulás mérő tengelyeinél erősítési hibával számoltam. A harmadik módszernél a gyorsulás mérő tengelyeinél ofszet és erősítési hibát is feltételeztem. A negyedik módszernél a gyorsulás mérő tengelyeinél ofszet és erősítési hibát, míg a mágneses orientáció mérőnél erősítési hibát feltételeztem. Az ötödik módszernél mind a gyorsulás mérő és mágneses orientáció mérőnél ofszet és erősítési hibát is feltételeztem.

A mérésnél a szenzort egy forgó székre helyeztem. Megforgattam párszor, majd nyugalmi helyzetében hagytam. A kiértékelés annál jobb minél kevesebb nullától eltérő vízszintes szakasz van a gyorsulásban, de ha van, az minél közelebb legyen a nullához, mert ezen vízszintes részeknél a szenzor egyhelyben állt. A következő grafikonkon az adatok különböző módszerekkel való feldolgozását látjuk időtartományban:



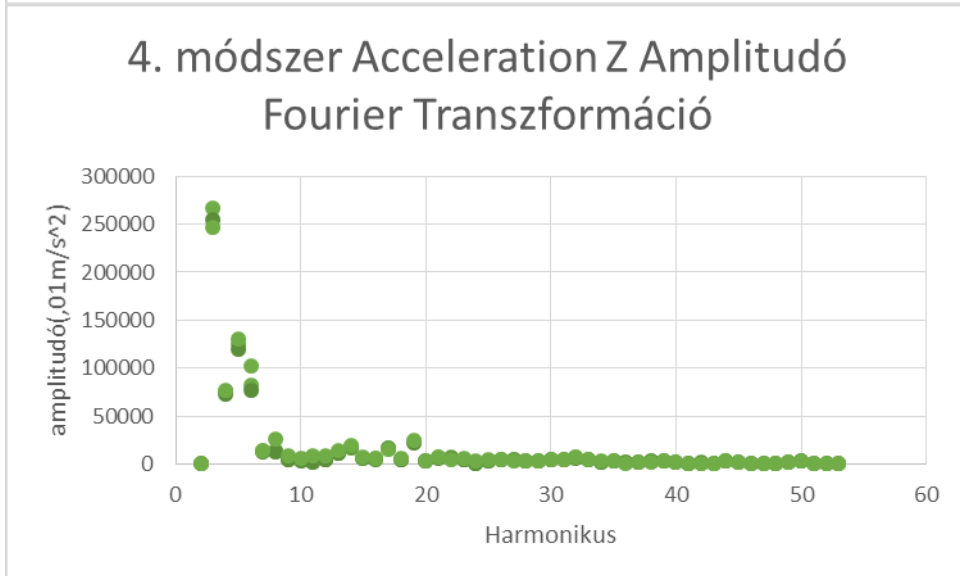
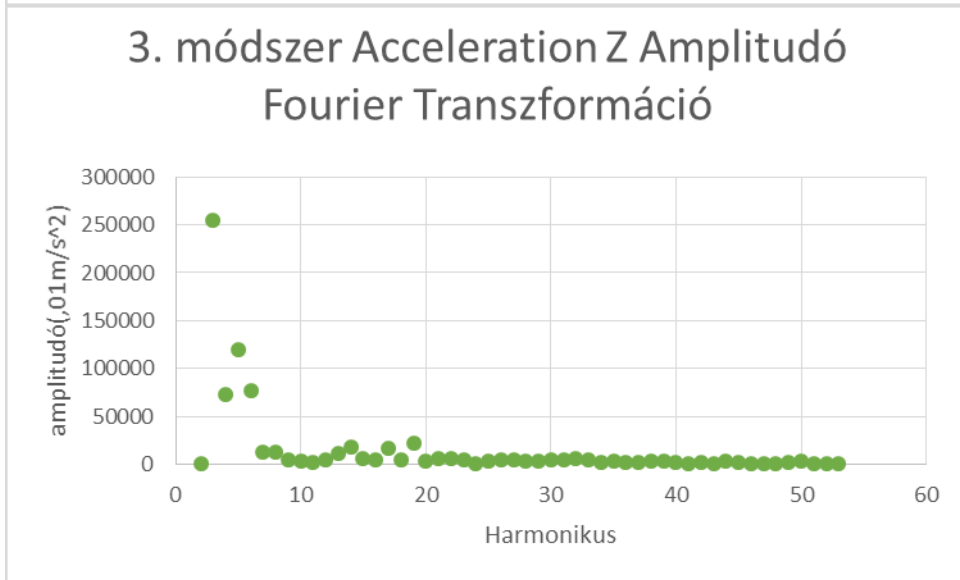
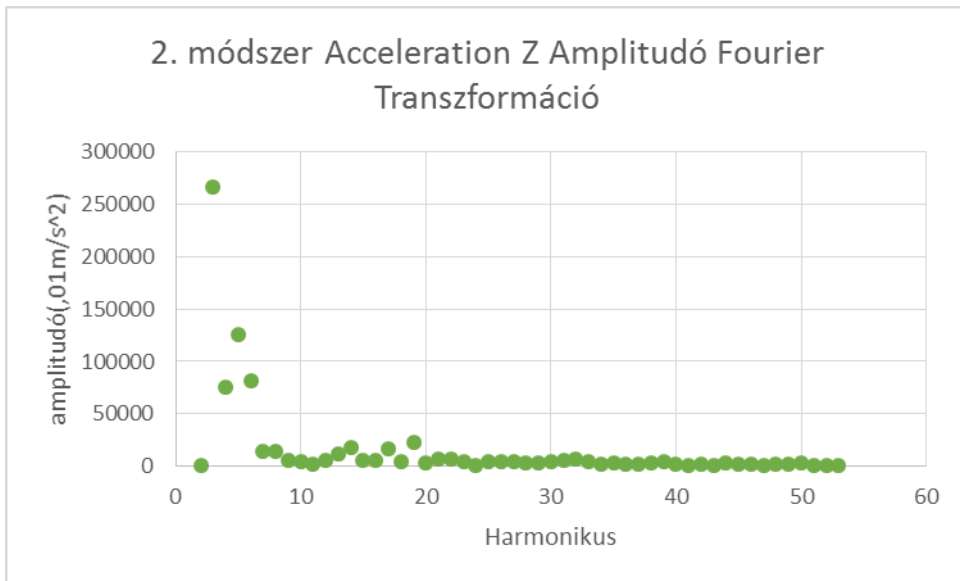


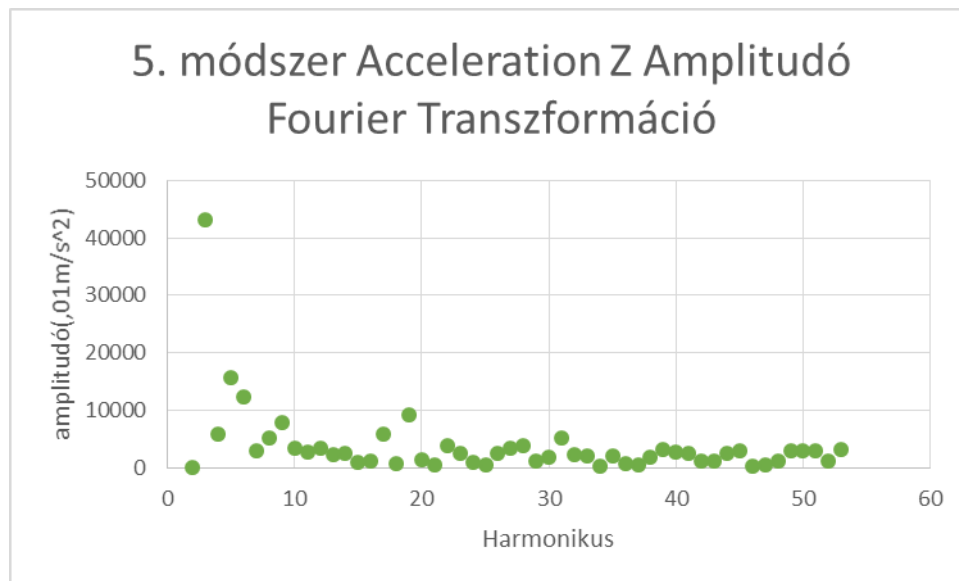
A kapott grafikonokból a következő következtetéseket vontam le.

Az első módszerrel történő feldolgozás nem vezet eredményre, irreális gyorsulást eredményezett.

Minél több tengely hibát veszünk figyelembe, egyre inkább közelít a valós gyorsulási görbéhez. A rendszert önmagában nem lehet helymeghatározásra használni, mert még mindig nagy hibája van. Viszont kis időtartamra, referencia adatokkal megsegítve már valóshoz közeli eredményt ad.

Ugyan ezen feldolgozott adatokat nem idő tartományban, hanem frekvencia tartományba is megvizsgáltam. (A z tengelyre vett vizsgálat). A grafikonokból látszik hogy a jel harmonikusainak amplitúdójai több hiba figyelembevételével csökken. Az 5. módszernél a z tengelyen vett értékek már egy kis korrigálással valós mozgás leírást eredményeznek. A következő grafikonon az adatok különböző módszerekkel való feldolgozását látjuk frekvencia tartományban:





A szenzor gravitáció mérő kalibrációs mérésnél említettem, hogy a legjobb mérésem szórása is elég nagy ahhoz az elméleti érték a mért értéknek a szórás környezetén belül legyen. Az előbbi mérés feldolgozásának eredményéből viszont arra lehet következtetni, hogy a mért tengely hibákkal való számolás pontosabb eredményre vezet, mintha a szenzort hibátlanak venném.

Gyakorlati alkalmazási lehetőségek

A [10] forrásban egy szenzort (mágneses orientáció, szögsebesség, gyorsulás mérő szenzor) önmagában használtak mozgás érzékelésre. Orvosi területeken és sportban levő felhasználást említettek. A testtartás és járás vizsgálatát, epilepsziás roham és Parkinson kór kivizsgálását és még más egyéb orvosi felhasználásokat. A sporton belül (például tenisz, golf) a sportolók teljesítményét lehetne növelni, ha sportolás közben a mozgásukat megfigyelik és mérik, majd a kapott eredményekből következtetéseket levonva optimalizálják a sportolók mozgását.

A bevezetőben említettem, hogy a céloom egy olyan segédrendszer kialakítása volt, amely egy optikai mozgásérzékelő rendszerrel szenzor fúziót alkotva, tetszőleges emberi mozgást kis mintavételi idővel jobban tud követni, mint egy optikai mozgásérzékelő rendszer önmagában. Ha a rendszert segéd rendszerként használom, a következő problémákat oldja meg, ami a felhasználás szempontjából előnyös. Minden x (maximum 10) adat után lesz egy referencia adat (optikai mozgásérzékelő rendszerből) amihez a szenzor adatait lehet igazítani. (Ha véletlenül az optikai mozgásérzékelő rendszerből egy kis időre erről a helyről nem jön adat, akkor is fog a szenzorból adat jönni. A szenzoros segédrendszer helyettesítheti az optikai rendszert az adott helyen nagy hibával, ami annál nagyobb minél tovább tart az az időtartam, amely alatt nem jön adat az optikai rendszerből. Ilyen elszakadás is maximum 200 darab mérésig tarthat.) Így minden olyan pontatlanság mely a hibák integrálódásából jön, nem lesz jelentős a mozgás leírásában. Tehát a szenzornak egyes tengelyein lehetnek kis hibák, mert ez kis időtávra a mozgás leírásánál nem fog nagy hibát okozni.

Összefoglalás

A munkám során először rendszerbeli átalakításokat végeztem az új rendszer elérése érdekében. Három fő változtatást tettem. Elsőként a programok struktúráját az egyes eszközök funkcióját és a szenzor kommunikációs eszközök paramétereit változtattam meg. Másodszer egy új kommunikációs protokollt dolgoztam ki mely a valós egy vagy több kapcsolatos master-slave zavar jeles modellben hatékonyabb (egységnyi idő alatt az elküldött és fogadott üzenetek száma nőtt). Harmadszor a kommunikációs időzítésekre egy protokollt vezettem be.

A rendszer elkészülte után a mérési adatok transzformálásával foglalkoztam majd a feldolgozott adatokból kiderült, hogy a szenzor tengelyeinek akkora hibája van, hogy feldolgozható mérési eredményt csak ezek kompenzálásával lehet elérni.

A szenzor gyorsulás és mágneses orientáció tengelyeinek hibájának meghatározására eljárást találtam a szakirodalomban, amelyet módosítva elvégeztem és meghatároztam a gyorsulás és mágneses orientáció tengelyeinek hibáját. Egy ellenőrző mérést is végeztem ezután, mely eredményét különbözőféleképpen dogoztam fel. Arra a következtetésre jutottam, hogy a kimért hibákkal alkotott szenzor modell a valósághoz közelebbi modellt alkot, mint anélkül. Majd egy alkalmazási lehetőséget és annak előnyeit mutattam be.

Jövőbeli tervek

A szenzor giroszkóp tengelyeinek hibájának meghatározására egy módszert szükséges kidolgozni. Az elkészült segédrendszer és a fő optikai mozgásérzékelő rendszer közötti kommunikációt is ki kell dolgozni. Eddig mindig a mérés után végeztem el az adatfeldolgozást, valós idejű adatfeldolgozás kidolgozására is szükség lesz majd.

A szenzorok időzítésének egymáshoz szinkronizálása (nem csak a kezdeteknél kell, hogy legyen és a nem csak a válaszok időzítése során) a mérés hibáját tovább csökkentené.

Források

- [1] György Rác, György Pilászy, Tamás Horváth(2015)
Tracking moving objects underwater in shallow lakes
- [2] Borszuk Judit(2014)
RÁDIÓS POZICIONÁLÓRENDSZER MOZGÓ EGYSÉGÉNEK FEJLESZTÉSE
- [3] Emanuele Garone a , Andrea Gasparri b , Francesco Lamonaca(2015)
Clock synchronization protocol for wireless sensor networks with bounded communication delays
- [4] Görkem Secer, Billur Barshan(2016)
Improvements in Deterministic Error Modeling and Calibration of Inertial Sensors and Magnetometers
- [5]LSM9DS0 Datasheet (3D accelerometer, 3D gyroscope, 3D magnetometer)
- [6]MRF49XA Data Sheet (ISM Band Sub-GHz RF Transceiver)
- [7] PIC18(L)F2X/4XK22 Data Sheet
- [8] H. P. Gavin(2016)
The Levenberg-Marquardt method for nonlinear least squares curve-fitting problems
- [9] <http://bme.videotorium.hu/hu/recordings/5316/merestechnika-5-eloadas>
- [10] A. Olivares, G. Olivares, F. Mula, J.M. Górriz, J. Ramírez (2011)
Wagyromag: Wireless sensor network for monitoring and processing human body movement in healthcare applications