

GPS adatok mérése és feldolgozása NI, LabVIEW szoftverrel

Balogh Ildikó Julianna
Mechatronikai Tanszék
Debreceni Egyetem, Műszaki Kar
Debrecen, Magyarország
baloghildiko55@gmail.com

Sarvajcz Kornél
Mechatronikai Tanszék
Debreceni Egyetem, Műszaki Kar
Debrecen, Magyarország
sarvajcz@eng.unideb.hu

Dr. habil Husi Géza
Mechatronikai Tanszék
Debreceni Egyetem, Műszaki Kar
Debrecen, Magyarország
husigeza@eng.unideb.hu

Absztrakt—Ezen fejlesztés keretein belül egy GPS adatok mérésére, feldolgozására képes program készült el LabVIEW-ban, amely képes a feldolgozás után az adatokat olyan formátumú fájlba kimenteni, amely alkalmas térkép megjelenítő szolgáltatások általi betöltésre, mint például a Google Map vagy Google Earth.

Kulcsszavak—gps; gnss; nmea; feldolgozás; LabVIEW; gpx; (kulcsszavak)

I. BEVEZETŐ

A kutatómunka témája a GPS adatok feldolgozása LabVIEW programozási nyelvben, amellyel az volt a cél, hogy a GPS vevőkészülék által közvetített adatokat a LabVIEW képes legyen fogadni ezeket az adatokat mérés közben fel tudja dolgozni, megjelenítésre alkalmas formátumba átalakítsa őket, majd megjelenítse, ezek után pedig egy olyan fájl formátumba mentése az adatoknak, amely megjeleníthető térképmegjelenítő alkalmazásokban, így a mérésünket, tehát a megtett útvonalunkat a keletkező fájl feltöltése után megtudjuk grafikusán is jeleníteni.

II. MŰHOLDAS HELYMEGHATÁROZÁS

A. GNSS rendszerek

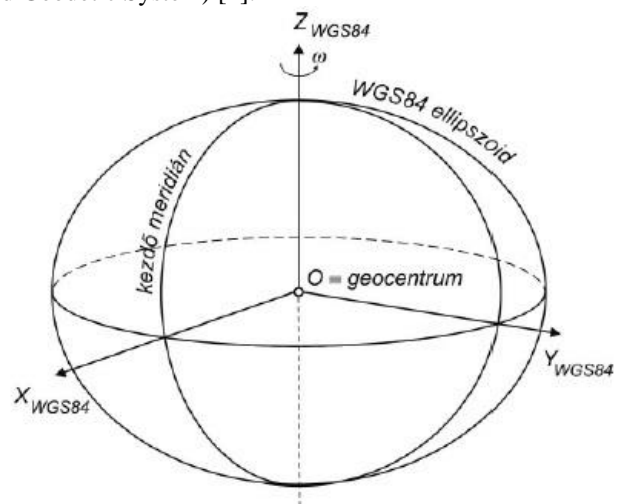
A globális navigációs műholdrendszer (Global Navigational Satellite System – GNSS) a helymeghatározás, a navigáció és az időmeghatározás feladatainak megoldását hivatott szolgálni mesterséges holdak segítségével. A GNSS technológiával végzett műholdas helymeghatározás azon alapszik, hogy a mesterséges holdak helyzete ismert, azok – geometriai értelemben – egy adott időpillanatban ismert pontoknak tekinthetők egy definiált vonatkoztatási rendszerben. A műholdak pályadatait ugyanis ismert helyzetű földi pontokon észlelő pályakövető állomások meghatározzák és ezeket az ún. navigációs adatokat a mesterséges holdak saját maguk sugározzák. A felhasználó (a GNSS vevő) közvetett módon távolságot határoz meg a vevő és néhány, egyidőben (szimultán módon) észlelt műhold között, majd ezen távolságok és a műhold-pozíciók ismeretében a vevő helyzete egy megadott vonatkoztatási rendszerben kiszámítható. A vevő sebessége is számítható, mivel ismert azon időtartam,

ami a vevő két helyzet-meghatározása között eltelik. A GNSS a globális műholdas helymeghatározó rendszerek (GPS NAVSTAR, GLONASSZ, Galileo, COMPASS, GINSS) közös elnevezése. Ezek közül jelenleg a GPS (Global Positioning System) és GLONASSZ (Globalnaja Nawigazionnaja Sputnikowaja Sistema) működik [1].

A GPS rendszerben az egyenlítő mentén 30° hosszúságkülönbséggel elosztott hat pályán 24 mesterséges hold kering. A holdak pályasíkjai 55° -os szöget zárnak be az egyenlítő síkjával. A holdpályák száma, helyzete a mesterséges holdak száma és elhelyezkedése a pályákon azt a célt szolgálja, hogy teljes kiépítésben a Föld bármely pontján, bármely időpontban egyszerre négy mesterséges hold észlelésre alkalmas helyzetben, azaz legalább 15° -ra a látóhatár síkja felett legyen [2].

B. A műholdas helymeghatározás vonatkoztatási rendszerei

A GPS műholdak által sugárzott fedélzeti pályaadatok vonatkoztatási rendszere a WGS84 (Geodéziai Világrendszer - World Geodetic System) [1].

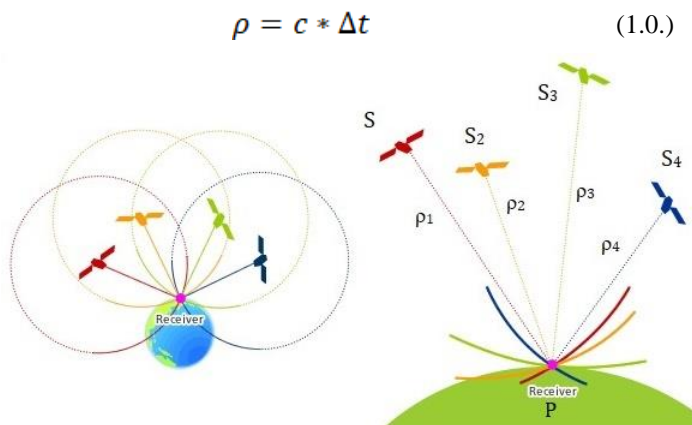


1. ábra A WGS84 tengelyei és vonatkoztatási ellipszoidja [1]

Ez a rendszer a GPS alaprendszer fenntartójának földi vonatkoztatási rendszere, amelyet a követőállomások koordinátái valósítanak meg. A rendszer fenntartói politikai szempontból az Egyesült Államok Nemzetvédelmi Minisztériuma (Department of Defense -DoD) szakmai szempontból az amerikai térképészeti szolgálat (National Geospatial Intelligence Agency - NGA). A WGS84 kezdeti 1984. évi meghatározása doppleres műholdmegfigyeléseken alapszik, ahol pályameghatározásban résztvevő állomások száma változott, kezdetben 5 volt, ami később 15-re bővült [1].

C. Műholdas helymeghatározás elve

Ha feltételezzük, hogy az S-el jelölt műholdon és az ismeretlen helyzetű P jelű vevőkészüléken is egy-egy tökéletesen szinkronizált és pontos óra van elhelyezve, akkor a műhold által a GPS időskálán t_0 időpillanatban kibocsátott kódolt jel a vevőhöz Δt idő múlva érkezik. Ha ezt a futási időt sikerül megmérni, akkor a futási időből egy a c fénysebesség ismeretében a vevő és a műhold közötti ρ geometriai távolság meghatározható [1].



2. ábra A műholdas helymeghatározás geometriai elve [3]

Ha a t_0 időpillanatban a ismert helyzetű S_2 és S_3 műholdakra is végzünk távolságmérést és a holdak geometriai szempontból megfelelő elhelyezkedésűek, akkor a vevő helyzete térbeli ívmetszéssel meghatározható. Abban az esetben, ha három távolságot mérünk, akkor három egyenlet írható fel három ismeretlen összetevő kiszámítására [1] [4].

Mivel a gyakorlatban a vevőben nem atomóra van elhelyezve, hanem kvarc óra, ami nem olyan pontos, mint az atomórák, így az nem olyan pontos, mint a műholdakon elhelyezett atomóra, így mindig van hiba a mesterséges holdak órájához képest. Ezt az órahibát δ jelöljük. A futási idő mérése ezen óra hiba miatt $c * \delta = \Delta \rho$ értékkel hibás lesz. A vevő órahibája miatt mért közelítő távolságot pszeudó távolságnak nevezzük. Szakkönyvekben R-rel jelölik [1].

$$R = \rho + \Delta \rho = \rho + c * \delta \quad (1.1.)$$

A vevő órahibájával mindig számolni kell, így összesen négy meghatározandó paraméterünk van. Ezek meghatározásához négy darab mért távolságra az az négy műholdra van szükség [1] [4].

III. UBLOX NEO-6 GPS VEVŐKÉSZÜLÉK

Az eltervezett feladat teljesítéséhez szükség volt egy külső GPS vevő készülék. Ilyen készülékből többféle is van. A Debreceni Egyetem, Mechatronikai Tanszékén a drónokhoz az U-blox NEO-6 modul sorozatának M-es modelljét használják, amely az u-blox6 pozíció motorjának nagy teljesítményét hozza a miniatűr NEO formához. Az u-blox 6-ot az alacsony energiafogyasztást és alacsony költségeket észben tartva tervezték. Ezek a rugalmas és költséghatékony vevők számos csatlakoztatási lehetőséget kínálnak egy miniatűr 16 x 12,2 x 2,4 mm-es csomagban. A kutatócsoport által használt GPS vevő készülék képes adatokat továbbítani, USB, SPI, UART és DDC-n keresztül[3]. Ezek közül a projekt során az UART kapcsolatot használtuk. Továbbá ez a vevő készülék az adatokat három különböző szabvány szerint tudja továbbítani mind USB, UART és DDC interfészen keresztül Ezek az NMEA, UBX és RTCM, melyek közül az NMEA protokollt használtuk.[5][6].

A GPS konfigurálása során az adatátviteli sebességet 9600 bit/s -re választottuk, ez volt megadva alapértelmezettként is a GPS vevőkészüléknel. Az alapbeállításoknak megfelelően a vevőkészülék az NMEA szabvány üzeneteit közvetítette, de mivel számunkra nem volt szükséges az összes adatsort használni, mivel nem feltétlenül szükséges a GPX fájl megvalósításához, ezért az RMC és GGA-s sorokon kívül a többi sort letiltottuk. A konfigurálást az u-center nevezetű programban hajtottuk végre.



NEO-6:
12,2 x 16,0 x 2,4 mm
3. ábra ublox NEO 6

IV. NMEA PROTOKOLL

Az NMEA üzenetek, amelyeket a GPS vevő továbbít az NMEA 0183 2.3 verziójára vannak alapozva [6]. Az NMEA 0183 kombinált villamos és adat specifikáció a tengeri elektronika, például a visszhangjelző, a hangjelzők, a szélmérők, az iránytűk, a robotpilóta, a GPS vevőkészülékek és számos más típusú eszköz kommunikációjához. Az NMEA 0183 szabvány egy egyszerű ASCII, soros kommunikációs protokollt használ, amely meghatározza, hogy miként továbbítják az adatokat egy "mondat" -ban egy "beszélő" -től több „hallgató” -hoz. Középső bővítők használatával a „beszélő” egyirányú beszélgetést szinte korlátlan számú hallgatóval folytathat és multiplexerek használatával több szenzor beszélhet egyetlen számítógéppel [7].

Az NMEA üzenetek mindig „\$” jellel kezdődnek utána, következik a címzett, ami, hogy ha GPS vevőkészülékről van szó akkor GP-vel, ha pedig saját üzenetről P-vel kezdődik. A

következő 3 karakter határozza meg az adat formátumát, utána következik az értékeket tartalmazó „mező”, amelyek vesszőkkel vannak elválasztva, majd az ellenőrző rész mindig „*” -val kezdődik, és két karaktert tartalmaz, amelyek egy hexadecimális számot képviselnek. Azok az adat mezők, ahol nem érkezett információ üresen maradnak [7][8].

Az NMEA üzenetek közül a választott GPS vevő készülék 14 megjelenítésére alkalmas (DTM, GBS, GGA, GLL, GPQ, GRS, GSA, GST, GSV, RMC, THS, VTG, ZDA) de ezek közül csak kettőt használtunk, tekintettel, hogy azok tartalmazzák a legfontosabb (szélesség, hosszúság, tengerszint feletti magasság, idő, és dátum) adatokat. Ez a két üzenet az RMC (Recommended minimum data) illetve GGA (Global positioning system fix data) üzenetek.

A. RMC (Recommended minimum data)

Példa:

\$GPRMC,083559.00,A,4717.11437,N,00833.91522,E,0.004,7,7.52,091202,,A*57

Mező száma	Példa	Formátum	Leírás
0	\$GPRMC	string	Üzenet ID
1	083559.00	hhmmss.ss	UTC idő, jelenlegi idő
2	A	ddmm.mmmm	Állapot A=Érvényes, V=Érvénytelen
3	4717.11437	ddmm.mmmm	szélességi fok, fok+szögperc
4	N	character	N/S, Észak vagy Dél
5	00833.91522	ddmm.mmmm	hosszúsági fok, fok+szögperc
6	E	character	E/W, Kelet vagy Nyugat
7	0.004	numeric	A földhöz viszonyított sebesség knots
8	77.52	numeric	Pálya a föld felett (fok)
9	091202	ddmmyy	Dátum nap/hó/év
10	-	numeric	Mágneses variációs érték (fok)
11	-	character	Mágneses variáció E/W jelző
12	-	character	Mód jelző
13	*57	hexadecimal	Checksum
14	-	character	Carriage Return és Line Feed

1. táblázat [7]

B. GGA (Global positioning system fix data)

Példa:

\$GPGGA,092725.00,4717.11399,N,00833.91590,E,1,8,1.01,4,99.6,M,48.0,M,,0*5B

Mező száma	Példa	Formátum	Leírás
0	\$GPGGA	string	Üzenet ID
1	092725.00	hhmmss.ss	UTC idő, jelenlegi idő
2	4717.11399	ddmm.mmmm	szélességi fok, fok+szögperc
3	N	character	N/S, Észak vagy Dél
4	00833.91590	ddmm.mmmm	hosszúsági fok, fok+szögperc
5	E	character	E/W, Kelet vagy Nyugat
6	1	digit	pozíció javítás állapotjelző
7	8	numeric	Használt műholdak száma 0-12 -ig
8	1.01	numeric	HDOP, Horizontal Delution of Precision
9	499.6	numeric	Tengerszintfeletti magasság
10	M	character	mértékegység, méter
11	48.0	numeric	Geoid elválás
12	M	character	mértékegység, méter
13	-	numeric	Differenciálkorrekció kora, üres, ha DGPS nem használt
14	0	numeric	Diff. referencia állomás azonosítója
15	*5B	hexadecimal	checksum
16	-	character	Carriage return és Line Feed

2. táblázat [7]

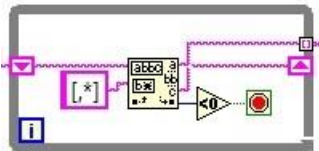
V. A PROGRAM MEGVALÓSÍTÁSA

A programot úgy terveztük, hogy az képes legyen külső GPS egység által küldött adatok beolvasására, feldolgozására, illetve szerettük volna, ha képesek vagyunk ezeket az adatokat egy olyan formátumban menteni, amelyet képes megjeleníteni a Google Maps vagy más hasonló alkalmazás. Ehhez a GPX (GPS Exchange Format) formátumot választottuk, ami egy nyílt XML séma, amelyet a GPS alkalmazások közös adatformátumaként terveztek. Használható, pontok (waypoint), nyomvonalak (track), útvonalak (route) leírására. Az útpontok egymástól különálló pontokat jelentenek, amelyek nincsenek kapcsolatban egymással. A nyomvonal azokat a pontokat jeleníti meg, ahol már voltunk, tehát valaki rögzítette, hogy ott járt, míg az útvonal olyan pontokat is tartalmaz, ahova el szeretnénk jutni [9]. Ezek alapján arra a döntésre jutottunk, hogy nyomvonalként fogjuk majd menteni

a beolvasott adatokat. A programot úgy szerettük volna elkészíteni, hogy az adatok fogadása közben, tehát a mérés közben a különböző adatok megjelenjenek különböző kijelzőkön is, illetve, hogy az eredeti GPS által küldött NMEA adatsorokat is el lehessen menteni. Első lépésként a program konfigurálja a portot a GPS beállításainak megfelelően, a különböző értékeket a Front Panelen megadhatók.

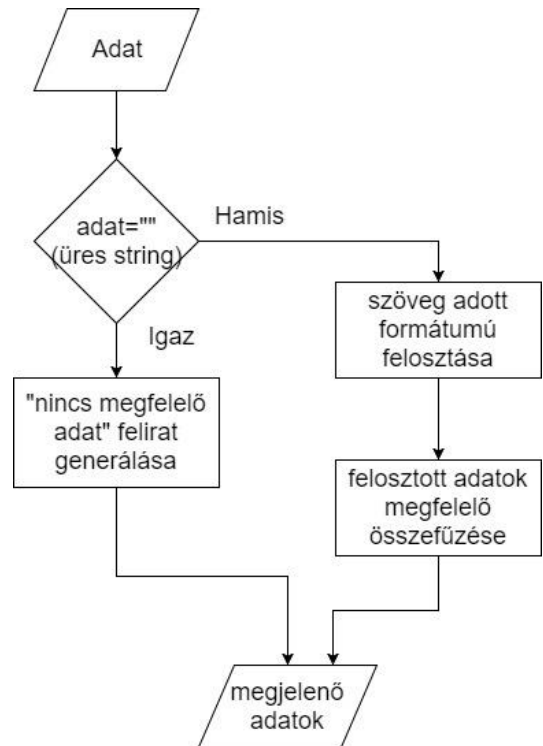
Ezután következik maga a mérés, amelyet mindaddig végzünk, amíg meg nem nyomjuk a STOP gombot. A mérés alatt a kijelzéshez először megvizsgáljuk, hogy érkezik-e adat, majd azt, hogy ez az érkező adat „\$” -val kezdődik-e. Abban az esetben, ha az előző feltételek teljesülnek, akkor megvizsgáljuk, hogy több mint 30 karaktert tartalmaz-e az adatsor. Abban az esetben, ha ez nem teljesül akkor a kijelzőkön a „Nincs megfelelő mennyiségű adat” jelenik meg. Abban az esetben, ha több mint 30 karakter tartalmaz az érkező adatsor akkor megkezdődik az adatok átalakítása, majd kijelzése illetve a megfelelő adatok tömbökbe gyűjtése a mentéshez. A 30 karakternél kevesebb adat azt jelenti, hogy a GPS modul nem kapcsolódik megfelelő számú műholdhoz, és nem tudja megjeleníteni a paramétereket.

Az adatok kiolvasása úgy történik, hogy sorban először a RMC típusú adatsorozat érkezik, majd a GGA adatsorozat. Az adatok átalakításához először is a kapott NMEA üzenetsort elemekre kell bontani. Ezt úgy oldottuk meg, hogy addig keresi a program „,”-t vagy „*”-ot a szövegben, amíg már nem volt több találat; a „*” keresésére azért, van szükség mert az adatsor utolsó eleme után már nincs vessző hanem rögtön az ellenőrző rész következik, ami „*”-gal kezdődik és utána egy hexadecimális szám van.



4. ábra „,” és „*” keresés

Az aktuálisan talált „,” előtti szöveget sorszámozzuk (Tunel Mode: Indexing), így a ciklusból való kilépés után egy adattömbünk lesz, amelynek mindkét esetben 0. eleme az NMEA kódolás típusát határozza meg tehát, hogy RMC vagy GGA típusú adat érkezett. A többi elem pedig a kódolásoknak megfelelő adatok értékeit tartalmazza megfelelő sorrendben. Ezután az adatokat lehetett külön-külön formázni, amit az alábbi folyamatábra alapján tettem meg.

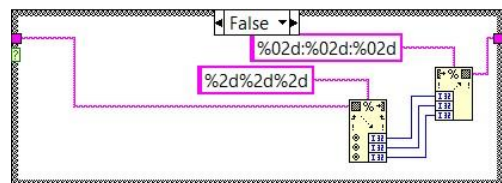


5. ábra Az adatok átalakításának folyamatábrája

Az átalakítandó adatok a földrajzi szélesség, földrajzi hosszúság, dátum, idő, sebesség és tengerszint feletti magasság. Ezeket a megjelenítés, illetve a mentés szempontjából is más alakban mentettem el.

Az 5. ábrán is látható, hogy az átalakításhoz először a kapott elemet megvizsgáljuk, hogy tartalmaz e adatot vagy nem; ha nem érkezett adat abban az esetben az adott kijelzőn a „nincs megfelelő adat” felirat jelenik meg. Abban az esetben, ha érkezett adat akkor ezt a szöveget bizonyos szempontok alapján felosztjuk, majd, ha szükséges matematikai átalakítást végzünk az egyes elemeken és utána újra összefűzzük a megfelelő sorrendben, megfelelő kiegészítésekkel, majd megjelenítjük, vagy a megfelelő tömbbe gyűjtjük, ha mentéshez szükséges.

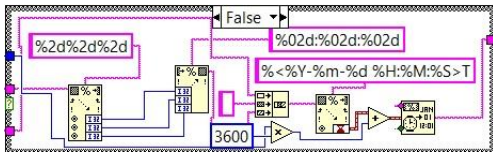
Ez alapján az idő átalakítása úgy történik, hogy az idő estében a 9 karaktert, amit kapunk (pl.: 091526.00) az első hat karaktert levágjuk, majd felosztjuk 3db 2 számjegyből álló számra, tehát 09, 15 és 26-ra, majd ezeket a karaktereket a „:” kiegészítésekkel újra összefűzzük, így a kapott a szöveg, amely az időt jelképezi a következő lesz: „09:15:26”.



6. ábra Az idő átalakítása

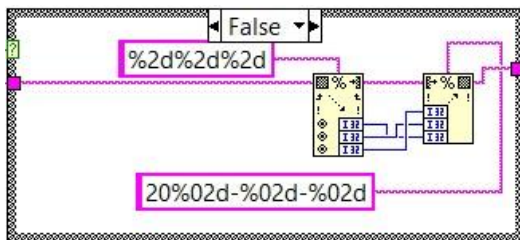
A kijelzőn megjelenő értéket lokális időnek szántuk és ahhoz, hogy ez működőképes legyen bármely időzónában

bármely időpillanatban lokális dátum és időt egyszerre kell megjelenítenem, ehhez a fenti ábrán ismertetett módon összefűztük az időt a megfelelő formátumba, majd elé fűztük a dátum már formázott értékét, ennek átalakítását később ismertetjük. Ebben az esetben a következő értéket kapjuk: pl.: 2017-11-09 09:15:26 szövegként, amit átalakítottunk „Time Stamp” formátumba, ehhez már hozzá tudjuk adni az időzónának megfelelő értéket, majd átalakítjuk a kapott értéket szöveg formátumba.



7. ábra Lokális dátum és idő átalakítás

A dátum esetében 6 karakter érkezik az alábbi formában: „091117”. Az első két karakter a napot a következő kettő a hónapot, míg az utolsó két karakter az évszámot jelképezi. Ezt a 6 karaktert az előbb említett felosztásban felosztjuk, majd a „-” -kel, illetve a „20” kifejezéssel kiegészítve újra összefűzzük a következő képpen: „2017-11-09”. Ez a karakterlánc kerül a mentésre és kijelzésre is.

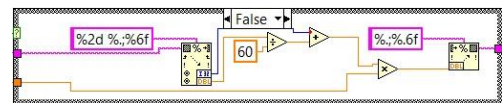


8. ábra Dátum átalakítása

A földrajzi szélesség esetében az adatok pl:4717.11437 formátumban érkeznek, amelyből az első két szám a fokot míg az összes többi karakter a szögperceket jelöli. Ezt a karakterláncot a mentéshez tizedesfokra kellett átalakítani, illetve a megjelenítésnél fok, szögperc, szögmásodperc alakban jelenítettük meg az értékeket.

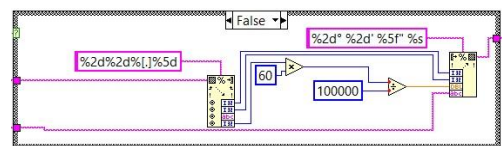
Az előbbihez az adatról leválasztottuk az első két karaktert, aztán a többi 8 karakterből álló számot tehát a „17.11437” -et elosztottam 60-nal, majd hozzáadtuk az első két karakterből álló számhoz, tehát a „47” -hez, így végül megkaptuk az értéket tizedes fokban. A fentebb említett példa alapján a mentésre kerülő érték: „47.2852395” lesz. A földrajzi szélességet -90° - 0 - 90° között határozták meg. A negatív értékek az egyenlítőtől délre vannak, míg a pozitív értékek az egyenlítőtől északra. Ezt az információt a GPS a földrajzi szélesség után érkező egy karakterből álló (N(Észak) vagy S (Dél)) üzenettel fejezi ki.

Ez alapján abban az esetben, ha a déli féltekén történne a mérés tehát, a GPS S értéket közvetítene felénk, akkor a példánkban lévő értéket meg kell szorozni -1-gyel, hogy megkapjuk a megfelelő értéket.



9. ábra A földrajzi szélesség átalakítása mentéshez

A megjelenítésnél az átalakítás úgy történik, hogy a karakterláncot felosztottuk 2db 2 jegyű számra a pontra és a maradék számjegyet egy számként kezeltük. Az első kétjegyű szám a fokot jelenti a második a szögpercet a pont utáni karakterláncból álló számot megszoroztuk 60-nal és elosztottuk 100000-el, így megkapva a szögmásodperceket. Ezek alapján a fenti példán bemutatva a kijelzőn megjelenő érték: „47° 17' 6,8622””. Ebben az esetben a kijelzésnél ezen felirat után fűztük, hogy melyik féltekéről is van szó. Értelem szerűen, ha a GPS N-et küld, akkor a fenti példa úgy néz ki, hogy: „47° 17' 6,8622” Észak”. S érkezése esetében az Észak felirat Délre módosul.



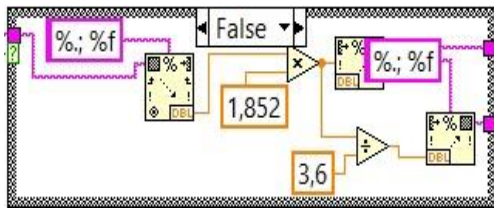
10. ábra A szélesség átalakítása a megjelenítéshez

A földrajzi hosszúság átalakítása ugyan ezen az elven alapszik, annyi különbség van csak, hogy mivel a földrajzi szélesség 0 - 180° -ig megy a Greenwich-től keleti, illetve nyugati irányban is ezért az első 3 karakter tartalmazza az értéket fokban. Ebben az esetben, ha Greenwich-től nyugatra vagyunk a GPS által küldött következő üzenet a W karaktert fogja tartalmazni, míg, ha keletre akkor az E karakter fog érkezni.

A mentésnél a számértéket -1-gyel szorozni a W karakter esetén szükséges. Az előbbi minta alapján, ha a beérkező adat: 00833.91522,E, akkor az átalakítás után a kijelzőn megjelenő

érték: „8° 33' 54,9132 Kelet”, míg a mentésre kerülő érték: „8,565254°”.

A sebesség esetében azért van szükség átalakításra, mert az adatsorban érkező adat csomómban érkezik. A kijelzőn a megjelenítés km/h-ban történik, míg a mentéshez m/s-ra van szükség.



11. ábra A sebesség átalakítása

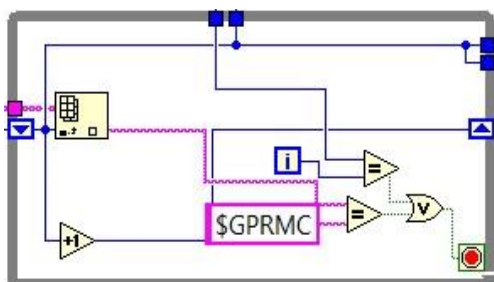
1 csomó = 1,852 km/h 1 m/s = 3,6 km/h

A tengerszint feletti magasságot nem kell átalakítani sem kijelzéshez, sem mentéshez, mivel m-ben kapjuk az értéket.

Ezen a fentebb említett értékeken kívül kijelzésre kerül még az állapotot tartalmazó adat, amely azt mutatja, hogy érvényes üzenet érkezik e a GPS-től. Ez az adat RMC-s adatsor esetén A (érvényes) vagy V (érvénytelen) értéket vehet fel, GGA-s adat esetén 0,1,2 és 6-ot., ahol az 1,2 esetén érvényes az érkező adat.

A kijelzőn még megjelenik a műholdak száma, ez az érték nem kerül mentésre a GPX fájlban, ahogy az állapot sem. Ezen kívül még megjelenik a kijelzőn a föld feletti irány is, amelyet fokban kapunk meg az RMC típusú adatban. A mentéshez meg kell adni, hogy hova szeretnénk menteni a fájlt. Abban az esetben, ha ez a fájl már létezik, akkor felül lesz írva, ha pedig nem akkor a program létrehozza azt. Az összes megjelenő üzenetet is elmenti a program egy szöveg fájlba, ennek az elérési útvonalát is meg kell adni, figyelni kell a kiterjesztés megadására.

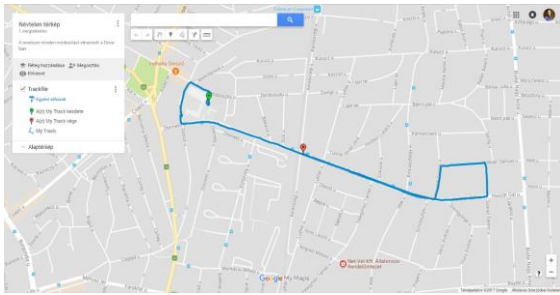
A GPX fájlba mentésre kerülő adatok kivezetésénél a csatorna módot (Tunel Mode) sorszámozásra állítottuk (Indexing), így a mérés végén a kivezetéseken 7 adattömb lesz (idő, szélesség, hosszúság, dátum, magasság, sebesség, jel típusa), amelyek szükségesek a GPX fájlhoz. Az üzenet típusa tömböt, azért kell létrehozni, hogy megtalálhassuk belőle az első elemet, ahol üzenet érkezett, mivel a mérés elején vannak üres üzenetek, így az első pár tömbem üres. Ahhoz, hogy a mentést elkezdhessük meg kell találni az első RMC-s adatokat.



12. ábra Kezdő tömbelemek keresése

Ehhez a jel típusánál keressük a „\$GPRMC tartalom addig, amíg meg nem találjuk, ha megtaláltuk akkor ennek a while ciklusnak a ciklus számát használjuk fel az RMC-s adatok tömbjeinek kezdőindexeként, hogy velük kezdődjön meg az adatok összefűzése nyomvonalpontokká. Mivel a GGA-s adatok a RMC-s adatok után érkezik, de ahhoz kiegészítő információt tartalmaznak, így a GGA-s adat tömbök indexe 1-gyel nagyobb. A következő lefutás alkalmával mindkét tömbindex kettővel nagyobb lesz. A ciklus feleannyiszor fut le mint, ahányszor lefutott a mérés while ciklusa, illetve ahány adatot tartalmaznak az adott tömbök. Az összefűzés csak akkor történik meg, hogy ha a szélesség, dátum, idő, hosszúság és magasság értékeket is kaptunk a mérés során.

Miután az összes nyomvonalpont összefűződött, azokat ahhoz, hogy GPX fájl legyen belőle össze kell fűzni még a GPX fájlnak megfelelő „fejléccsel” illetve „lábléccsel”. Azért döntöttünk a szöveggént történő összefűzés mellett, mivel a LabVIEW csak a saját XML sémáját képes kezelni, illetve bármilyen szöveg szerkesztőben lehet XML fájlt írni, így ez tűnt kézenfekvőnek. Ezzel a lépéssel a programnak vége szakad. Ezek után megnyitható mind az NMEA üzenetek tárolására alkalmazott szövegfájl illetve a GPX fájl. A GPX fájl fel lehet tölteni annak megjelenítésére alkalmas térkép megtekintő alkalmazásokban pl.: Google Maps, Google Earth.



13. ábra A mentett adatok megjelenítés Google Maps-ben

- [9] GPS Exchange Format [Online] Elérhető:
<http://www.topografix.com/gpx.asp> [Hozzáférés dátuma: 2017.10.16]

VI. ÖSSZEFOGLALÁS

Összességben elmondható, hogy kutatómunkában készítettünk egy komplex helymeghatározó programot, amely képes bármilyen típusú GPS modul adatait fogadni soros kommunikáción, értelmezni és feldolgozni az érkező adatokat, illetve szöveges formátumban kimeneti a transzformálatlan adatokat a későbbi feldolgozás érdekében, illetve szabványos GPX formátumú útvonalat készíteni az adatokból. A GPS adatok feldolgozása egy komplex gépjárműdiagnosztikai rendszer részét képezi, amely valós időben képes a gépjárműdiagnosztikai csatlakozóján keresztül menetdinamikai adatokat olvasni és vele azonos időben az útvonal adatokat is menteni.

KÖSZÖNETNYILVÁNÍTÁS

A publikáció/prezentáció/poszter elkészítését az EFOP-3.6.1-16-2016-00022 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

HIVATKOZÁSOK

- [1] Dr. Busics György, "Műholdas helymeghatározás 1., A GNSS-ről általában" Nyugat-magyarországi Egyetem 2010.
- [2] Dr. Varga József „GPS alapismeretek kezdőknek” [Online] Elérhető: http://www.agt.bme.hu/staff_h/varga/gps/kezdoknek.html [Hozzáférés dátuma: 2017.10.16.]
- [3] [Online] Available: http://images.slideplayer.com/37/10727197/slides/slide_6.jpg. [Hozzáférés dátuma: 21. 11. 2017.]
- [4] I. Dr. Szalai, Mechatronikai rendszerek speciális érzékelői és aktuátorai, Pannon Egyetem, 2014.
- [5] NEO-6 series Versatile u-blox 6 GPS modules [Online] Elérhető: https://www.u-blox.com/sites/default/files/products/documents/NEO-6_ProductSummary_%28GPS.G6-HW-09003%29.pdf [Letöltve: 2017.10.15.]
- [6] NEO-6 u-blox 6 GPS Modules Data Sheet [Online] Elérhető: [https://www.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_\(GPS.G6-HW-09005\).pdf](https://www.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_(GPS.G6-HW-09005).pdf) [Letöltve: 2017.10.16]
- [7] u-blox 6 Receiver Description Including Protocol Specification [Online] Elérhető: https://www.u-blox.com/sites/default/files/products/documents/u-blox6_ReceiverDescrProtSpec_%28GPS.G6-SW-10018%29_Public.pdf [Letöltve: 2017.10.16]
- [8] NMEA 0183 [Online] Elérhető: http://www.nmea.org/content/nmea_standards/nmea_0183_v_410.asp [Hozzáférés dátuma: 2017.10.16]