



M Ű E G Y E T E M 1 7 8 2

**Budapesti Műszaki és Gazdaságtudományi Egyetem**

Villamosmérnöki és Informatikai Kar

Méréstechnika és Információs Rendszerek Tanszék

# Megbízható kommunikációs kapcsolattal rendelkező földi irányító állomás fejlesztése UAV-hoz

SZAKDOLGOZAT

*Készítette*

Böjti Paszkál

*Konzulensek*

Vörös András, Dr. Bartha Tamás

2013. november 4.

# Tartalomjegyzék

<b>Kivonat</b>	<b>4</b>
<b>Abstract</b>	<b>5</b>
<b>Bevezető</b>	<b>6</b>
<b>1. A feladatkiírás pontosítása és részletes elemzése</b>	<b>7</b>
<b>2. Előzmények (irodalomkutatás, hasonló alkotások), az ezekből levonható következtetések</b>	<b>8</b>
2.1. Repülőgép felépítése . . . . .	8
2.2. Vezeték nélküli modem . . . . .	9
2.3. Földi állomások . . . . .	10
2.3.1. Hadászati . . . . .	10
2.3.2. Paparazzi . . . . .	10
2.3.3. ArduPilot . . . . .	11
<b>3. Tervezés részletes leírása, a döntési lehetőségek értékelése és a választott megoldások</b>	<b>12</b>
3.1. Kommunikáció . . . . .	12
3.2. Redundáns jelek feldolgozása . . . . .	12
3.3. Feltöltés . . . . .	12
3.3.1. Protokoll . . . . .	12
3.4. GUI kialakításának szempontjai . . . . .	12
3.4.1. . . . .	13
<b>4. Megtervezett műszaki alkotás értékelése, kritikai elemzése, továbbfejlesztési lehetőségek</b>	<b>14</b>
4.1. Főképernyő . . . . .	14
4.2. Diagnosztikai adatok . . . . .	14
4.3. Hibadiagnosztika . . . . .	14
4.4. RS232 . . . . .	14
4.5. Sebesség . . . . .	16
4.6. Irány . . . . .	17
4.7. GUI . . . . .	17

<b>5. Összefoglalás</b>	<b>19</b>
<b>Függelék</b>	<b>21</b>
F.1. Függelék1 . . . . .	21

## HALLGATÓI NYILATKOZAT

Alulírott *Böjti Paszkál*, szigorló hallgató kijelentem, hogy ezt a szakdolgozatot meg nem engedett segítség nélkül, saját magam készítettem, csak a megadott forrásokat (szakirodalom, eszközök stb.) használtam fel. Minden olyan részt, melyet szó szerint, vagy azonos értelemben, de átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

Hozzájárulok, hogy a jelen munkám alapadatait (Böjti Paszkál, Megbízható kommunikációs kapcsolattal rendelkező földi irányító állomás fejlesztése UAV-hoz, angol és magyar nyelvű tartalmi kivonat, 2013, Vörös András, Dr. Bartha Tamás) a BME VIK nyilvánosan hozzáférhető elektronikus formában, a munka teljes szövegét pedig az egyetem belső hálóján keresztül (vagy autentikált felhasználók számára) közzétegye. Kijelentem, hogy a benyújtott munka és annak elektronikus verziója megegyezik. Dékáni engedéllyel titkosított diplomatervek esetén a dolgozat szövege csak 3 év eltelte után válik hozzáférhetővé.

Budapest, 2013. november 4.

---

*Böjti Paszkál*  
hallgató

# Kivonat

Napjainkban egyre nagyobb teret hódít a pilóta nélküli légi járművek alkalmazása. Az 1960-as években a hadszíntéren jelentek meg először, ahol megfigyelésre, felderítésre és olyan feladatokra használták, ahol kockázatos lett volna emberi életet veszélyeztetni. Az utóbbi években praktikussága, alacsony üzemeltetési költségei miatt más területeken is hasznosnak bizonyult ez a technológia, pl. geológia mintázatok kutatása, mely az emberi perspektívából nehezen észlelhető, tűzoltósági alakulatok koordinálása, otthoni hobby felhasználás.

Az MTA-SZTAKI Rendszer és Irányításelméleti Kutatólaboratóriumában kidolgozott szabályozó algoritmusok gyakorlatba való átültetésére egy pilóta nélküli járművet hoztak létre, mely a biztonságos üzemeltetés mellett, illetve az estelegesen előfordulható hibák ellen redundáns hardware elemekkel védekezik. Szakdolgozatom keretében ennek a repülőnek a földi állomását dolgoztam ki, mely a redundánsan küldött rádiójelek feldolgozására és megfelelő megjelenítésre használandó. A földi személyzet mozgó térkép alapú vizualizáción láthatja az aktív útvonalpontokat és a gép útvonalát. Az esteleges hibákat egy másik nézetben áttekinthetően láthatja.

# Abstract

Nowadays...

# Bevezető

a feladat értelmezése, a tervezés célja, a feladat indokoltsága, a diplomaterv felépítésének rövid összefoglalása

[1]A pilóta nélküli légi jármű gondolata egészen a XX. század elejére nyúlik vissza, mikor az I. világháborúban egy olyan távirányítású repülőt alkottak, mely robbanószerrel a fedélzeten a célpontba csapódva okozott kárt. Később a vietnámi háborúban több mint 3000 küldetésben vett részt ilyen repülő és mindössze 554 veszett oda. A technológiai korlátok miatt a fő funkcionalitása video felvétel készítése egy meghatározott útvonalon (általában egyenes vonal, körökkel kiegészítve) repülve, majd a bázisra való visszaérkezés. A rádiótechnika fejlődése miatt egyre összetettebb feladatok elvégzésére lettek képesek. A nagyobb átviteli sebességnek köszönhetően valós időben, monitoron keresztül kezelheti az operátor a távirányítású repülőt. A mai UAV-k több üzemmódot is támogatnak, egyik az előbb említett távirányítás, másik a fedélzeti intelligenciára hagyatkozó. Mind a hagyományos repülőiparban, mind ebben az érában, szükséges és célszerű az emberi terhelés csökkentése, utasszállító gépek esetében is a robotpilóta elvégez minden olyan korrekciót, melyet azelőtt a pilóta folyamatos figyelésével, koncentrációjával lehetett elérni. A modern integrációnak köszönhetően, olyan fejlett feldolgozóegységgel dolgozhatjuk fel az adatokat, melyeknek nem jelentős a fogyasztása, nem foglalnak sok helyet. A szenzorokból érkező információkra, az algoritmusoknak köszönhetően úgy képes reagálni, hogy az nem veszélyezteti a repülő levegőben maradását. Ám hiába a fejlett hardware, a valóban automatikus üzemeltetés még mindig távoli cél, emberi beavatkozás mindig is kelleni fog olyan helyzetekben melyre nincs előre felkészítve az intelligenciája. Ilyen esetekben létfontosságú, hogy az operátor lássa a gép aktuális pozícióját, diagnosztikai adatait.

Dolgozatomban áttekintem a legjelentősebb nem katonai fejlesztésű UAV-khez készült földi irányító állomások tulajdonságait, az ezekből levont következtetéseket tervezési fázisban felhasználok. Bemutatom a kommunikációhoz használt vezeték nélküli modemet, annak a rendszerbe illesztését. Kidolgozom a redundáns jelek feldolgozását, azok felhasználóbarát megjelenítését, továbbá a vissza-irányú kommunikáció protokollját, ennek tervezési módszerét. Végül a térkép alapú vizualizáció megoldását mutatom be.

## 1. fejezet

# A feladatkiírás pontosítása és részletes elemzése



## 2. fejezet

# Előzmények (irodalomkutatás, hasonló alkotások), az ezekből levonható következtetések

### 2.1. Repülőgép felépítése

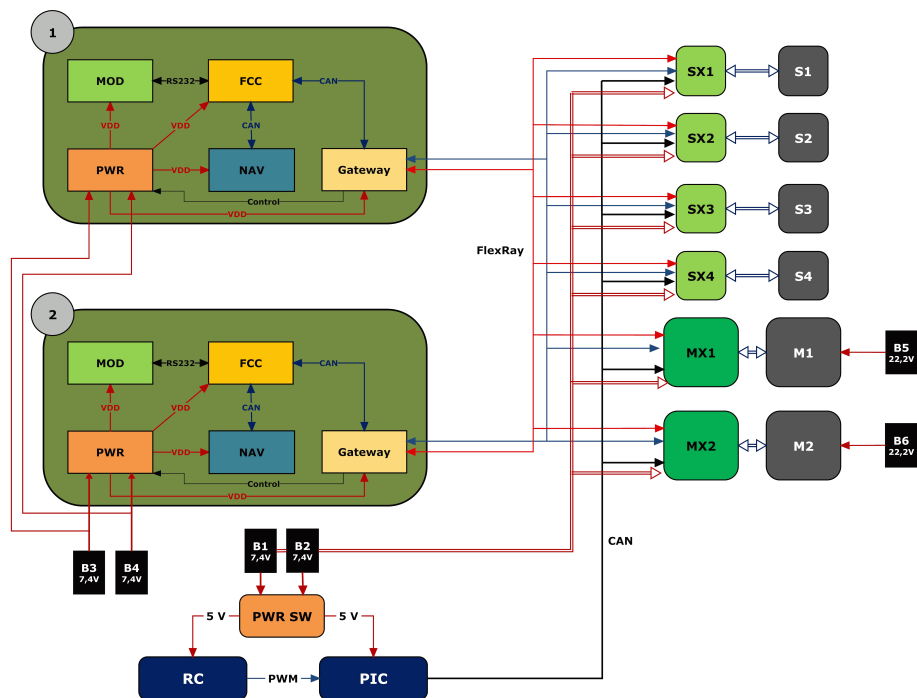
A repülőgép egy saját fejlesztésű modell, a tervek alapján a laborban készült el. A biztonságkritikusság miatt redundánsan szerepel:

- motor
- akkumulátor
- központi számítógép



2.1. ábra

2.2. ábrán látható, hogy központi számítógép kettőzött, párhuzamosan működnek, irányítják a servo motorokat. A szenzorokból érkező jelek egy Gateway-en keresztül az FCC-be jutnak, ahol a feldolgozás után a szabályozó algoritmusok kiadják a megfelelő utasítást a kormánysserveknek. Ilyen utasítás lehet pl, a csűrőlapok adott fokban történő elmozdítása,



2.2. ábra

motor fordulatszámának változtatása. A motorok, úgy mint a központi egység, külön tápellátást kapnak, ezzel is csökkentve a végzetes meghibásodás esélyét. Jelen esetben egyik legfontosabb elem a kommunikációért felelős modem, mely soros porton keresztül kapcsolódik az FCC-hez. A vevő oldalon hasonló modem, szintén soros porton küldi a földi állomásnak a vett jelet, a köztük lévő vezeték nélküli kommunikáció saját szabványa a cégnek. A választás [12]XBee-PRO 868 típusú modemre esett, mely alacsony fogyasztása és nagy hatótávolsága miatt ideális egy ilyen környezetbe.

## 2.2. Vezeték nélküli modem

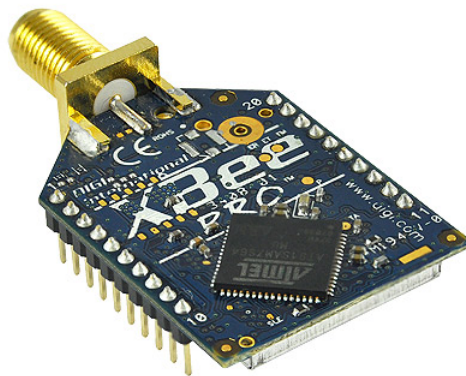
A kiválasztott modem kétféleképpen képes kommunikálni egy másik modemmel:

- API csomagküldés
- AT transzparens

**API** módban egy eszköz több eszköztől tud csomagokat venni, ha egy csomag megérkezett a küldőtől a fogadóig, egy ACK<sup>1</sup> üzenettel válaszol, ha ezt nem kapja meg, újraküldi. Egy csomag többek közt tartalmazza a küldő és a fogadó címét, lehetőség van broadcast üzenetek küldésére is, melyet minden eszköz megkap. Továbbá az adatintegritás céljából checksum mezőjében összesítve van egy csomag adata.

**AT** mód nem más mint egy vezeték nélküli kapcsolat 2 sorosport közt. Nem csinál mást, mint a soros portján bejövő adatokat egy XBee szabvány szerint rádiójelekké alakítja, melyet a virtuális kapcsolat végpontja fogad és visszaalakítja sorosportra. Ez pont-pont kommunikációra hivatott.

<sup>1</sup>Acknowledgement, nyugtázó üzenet



**2.3. ábra**

## **2.3. Földi állomások**

Egy UAV vezetéséhez elengedhetetlen egy bázis, ahonnan a földi személyzet irányítja, monitorozhatja a repülést. Általában több funkciót lát el:

- Küldetés tervezés: útvonal meghatározása, illetve a célpontok kijelölése
- Adatok megjelenítése: megfelelő módon kijelezni a repülőgép állapotát, esetleges hibát.

Számos megoldás született a földi állomás GUI<sup>2</sup>-jának kialakítására. Elsődleges követelmény, hogy az operátor mindig a legfontosabb információkat láthassa, ehhez szoftverergonomiailag kell megtervezni a műszerek, adatok elrendezését. Az alábbiakban összehasonlításra kerülnek a megjelenítési felületek.

### **2.3.1. Hadászati**

Általában közel a hadszíntérhez üzemeltetik a repülőket, így a személyzetet, illetve a felszereléseket páncélozott járművel szállítják, védik. Pl. az MQ-1 Predator GCS egy utánfutóban helyezkedik el, mely szünetmentes tápegységet biztosít a pilóta és segédszemélyzeti, adatelemző/rögzítő és radar munkállomásoknak. A kommunikációt UHF és VHF frekvencián bonyolítják közvetlen rálátás esetén, azon kívül műhold közbeiktatásával.

A 2.5. ábrán egy hordozható állomás látható, mely magában foglalja az akkumulátort, kijelzőt, kezelőszerveket. Kompaktsága miatt ez a legelterjedtebb a kézzel indítható UAV-k esetében.

### **2.3.2. Paparazzi**

Ez egy több platformos GCS, mely különböző projektekben létrehozott repülőkhöz használható valós-idejű monitorozásra. Könnyen testre szabható az igényeknek megfelelően.

? Simultaneous flying multi UAS support ? Multi-system support (multiple protocols, multiple autopilots/projects) by writing a IVY Plugin ? 2D Map capable of displaying

<sup>2</sup>Graphics User Interface, grafikus megjelenítés



**2.4. ábra.** *Utánfutós megoldás*



**2.5. ábra.** *Hordozható GC*

Google Satellite, OpenStreetMaps Images and Microsoft Satellite Maps ? Mission planning ? Realtime movable waypoints ? Realtime flightplan adjustments if needed ? System status overview ? Realtime Airframe in Air tuning and calibration ? Supports rotary and fixed-wing e.g. Airplanes, helicopters, coaxial and quadrotors ? Definable Hotkeys for quick simple in the field control ? Voice status output ? Full freely configurable GUI layout

### **2.3.3. ArduPilot**

TODO

## 3. fejezet

# Tervezés részletes leírása, a döntési lehetőségek értékelése és a választott megoldások

### 3.1. Kommunikáció

### 3.2. Redundáns jelek feldolgozása

### 3.3. Feltöltés

A repülőgép által lerepülendő feladat útvonalpontjait hasonlóképpen, mint az adatok fogadását, vezeték nélküli csatornán küldjük fel. A feltöltendő adat küldésének protokollja létfontosságú, mivel ha valamilyen hiba kerül a kommunikációba akkor az akár végzetes is lehet. Gondolok itt olyan hibára, hogy egy fordulópont koordinátája úgy kerül feltöltésre, hogy az kiesik a repülő hatósugarából és ezzel nem számolva, lemerül a tápellátást szolgáló akkumulátor. Az ilyen hibák ellen célszerű a feltöltés protokolljába hibadetektálást építeni, hogy ezek a feldolgozás( esetlegesen felszállás ) előtt kiderüljenek ki.

#### 3.3.1. Protokoll

A követendő fordulópontokat egy fejléccel és egy checksum-mal ellátott csomagban küldjük el a repülőnek. Mivel a modem is duplikált, így a 2 sorosportra csatlakoztatott modemmel redundánsan küldjük el a csomagot. Ha a csomag sértetlenül megérkezett, ACK jelzéssel válaszol.

### 3.4. GUI kialakításának szempontjai

Az előző fejezetben ismertetett grafikus felületekből levonva a következtetéseket, nyilvánvaló, hogy a GUI kialakításában fontos a repülőgép aktuális pozíciójának térképen való mutatása, az repülési állapot könnyen értelmezhető megjelenítése, illetve az esetlegesen előforduló problémák feltűnő jelzése.

**3.4.1.**

## 4. fejezet

# Megtervezett műszaki alkotás értékelése, kritikai elemzése, továbbfejlesztési lehetőségek

### 4.1. Főképernyő

### 4.2. Diagnosztikai adatok

### 4.3. Hibadiagnosztika

### 4.4. RS232

A modemből érkező adatokat soros porton keresztül fogadja a program, a tesztkörnyezet felállításához HIL adatok szolgáltak. A küldött log fájlokat egy programmal beolvasom és egy [11]null-modem segítségével sorosporton keresztül küldöm a megfelelő portra.

Beállítottam 2 párt, COM20-COM21 és COM10-COM11 közt, a pároson küldöm, páratlanon fogadom az üzeneteket.

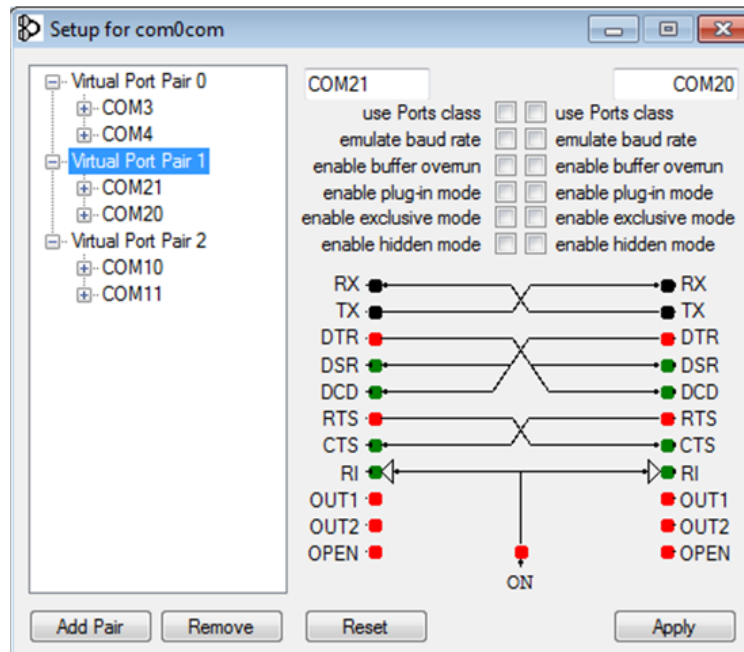
153 bájtos egy csomag, melyet egy UUT 3 bájtos fejléc és egy 2 bájtos checksum zár. A checksum a hasznos bájtok 16 bitre csonkolt összege. Minden fogadott csomagnál, a feldolgozás előtt kiszámolom az összeget és ellenőrzöm, az egyezést, a rossz csomagok egyelőre eldobásra kerülnek.

Fogadó oldalon a két sorosport aszinkron ír 1-1 byte tömböt, melyből egy dekódoló függvénnyel nyerjük ki a sebesség, pozíció, irány, stb. adatokat.

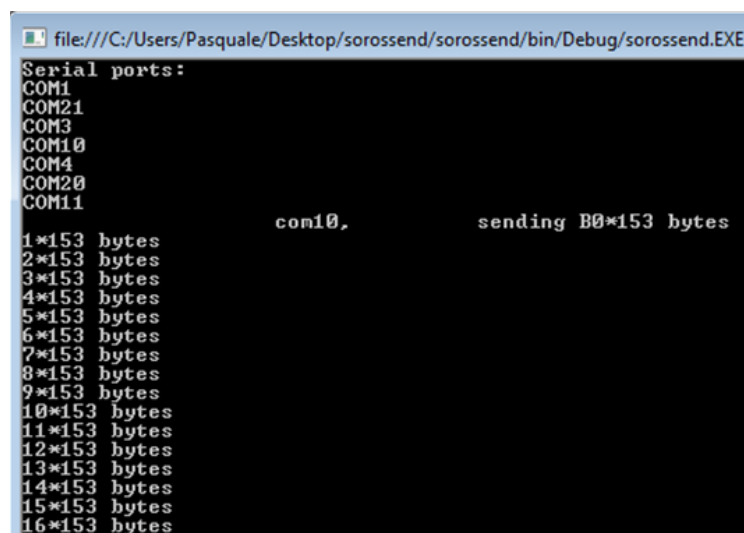
```
public double[] Decode(byte[] array)
```

Ebben a függvényben ellenőrzöm, a checksum-ot, illetve a kezdő UUT bájt hármast. Mivel bájtosával lehet feldolgozni az adatokat, így pl. a 4 bájtos időbélyeget 4 db egymás után jövő bájtból kell összerakni:

```
uint ido = (uint)array[3]<<24 | (uint)array[4]<<16 |  
(uint)array[5]<<8 | (uint)array[6];
```



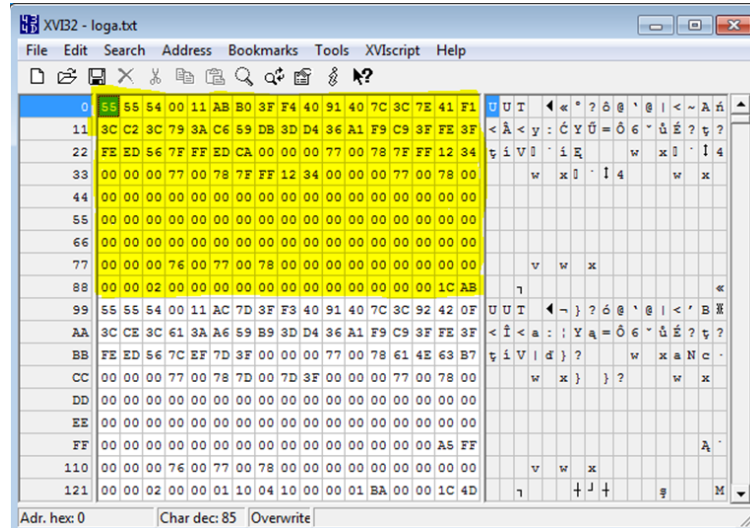
4.1. ábra



4.2. ábra

Ugyanígy folytatódik az adatok feldolgozása, az előre megadott protokoll szerint.





4.3. ábra

bájt index	leírás	típus	skálázás	offset
1	start	char(fix 'U')		
2	start	char(fix 'U')		
3	start	char(fix 'T')		
4	idő 1/4	unsigned int	10000	0
5	idő 2/4			
6	idő 3/4			
7	idő 4/4			
...				
26	északi irány 1/2	unsigned short	0x7FFF/400	200
28	keleti irány 1/2	unsigned short	0x7FFF/400	200
30	lefelé irány 1/2	unsigned short	0x7FFF/400	200
...				

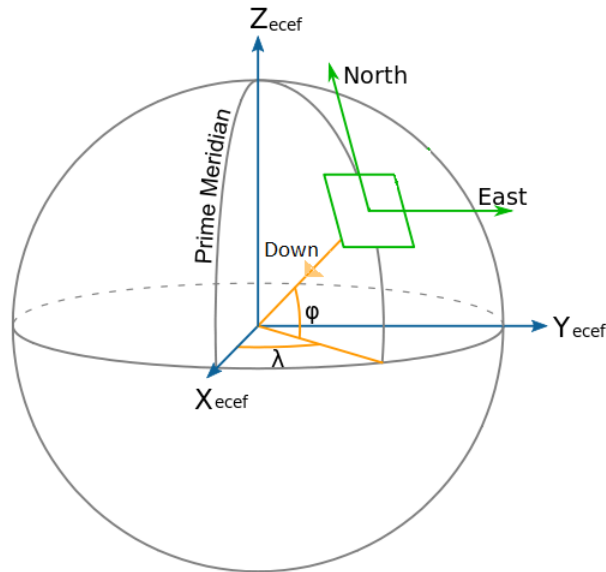
Mivel a változások Hamming-távolsága<sup>1</sup> kicsi lenne, az eredeti számábrázoláson, így skálázással és offset képzéssel megnöveljük. A sebesség adatoknál a visszakódolás:  $eredeti = (nyersadat/skalazas) - offset$  képlettel oldható meg.

#### 4.5. Sebesség

A sebesség NED<sup>2</sup> koordinátarendszerben van megadva, mely a repülő középpontjából indul. Mivel kis magasságban repül a repülő, így síknak közelíthetjük a Föld felületét, ez számítások szempontjából előnyös, mivel könnyebb vele dolgozni. A sebesség számítási módja:  $\sqrt{(V_E)^2 + (V_N)^2}$   
Emelkedés:  $-V_D$

<sup>1</sup>Bináris számok XOR képzésével kapott 1-esek száma

<sup>2</sup>Nort East Down, Local Tangent Plane, helyi koordináta rendszer



4.4. ábra. Koordináta rendszer

## 4.6. Irány

Az irány meghatározásához a 4 negyedsíkot külön kellett választani:

```

if (Ecomp > 0 && Ncomp > 0)
{
    heading = Math.Atan(Ecomp * Ncomp);
}
else if (Ecomp > 0 && Ncomp < 0)
{
    heading = Math.Atan(Ecomp * Math.Abs(Ncomp)) + 90;
}
else if (Ecomp < 0 && Ncomp < 0)
{
    heading = Math.Atan(Math.Abs(Ecomp) * Math.Abs(Ncomp)) + 180;
}
else if (Ecomp < 0 && Ncomp > 0)
{
    heading = Math.Atan(Math.Abs(Ecomp) * Ncomp) + 270;
}

```

## 4.7. GUI

A grafikus felület kialakítása során figyelembe kell venni, hogy első ránézésre a legfontosabb adatok látszódnak. Kettő fél közül az első oldalon a legfontosabb műszerek találhatók, jobb oldalon egy Google Maps térkép. A térkép egy lokális cache-ből tölti be az előre letöltött térképszelvényeket, így a terepen lehetőség van offline módon is használni ezt a

funkciót.

## 5. fejezet

# Összefoglalás

SZUMM

# Irodalomjegyzék

- [1] [http://en.wikipedia.org/wiki/Unmanned\\_aerial\\_vehicle](http://en.wikipedia.org/wiki/Unmanned_aerial_vehicle), 2013. május 4, 10:00
- [2] [http://en.wikipedia.org/wiki/Hewitt-Sperry\\_Automatic\\_Airplane](http://en.wikipedia.org/wiki/Hewitt-Sperry_Automatic_Airplane), 2013. május 4, 10:00
- [3] [http://en.wikipedia.org/wiki/Unmanned\\_combat\\_air\\_vehicle](http://en.wikipedia.org/wiki/Unmanned_combat_air_vehicle), 2013. május 4, 10:00
- [4] <http://www.defenseindustrydaily.com/uav-ground-control-solutions-06175/>, 2013. május 4, 10:00
- [5] [http://api.ning.com/files/ga5AVWy8xTu8cx9rHdzJ73Epc\\*dzb0uy\\*UPe0O8wFxogMF6WNRW8StK3xzkG2iBk16kNFpYTZe80NgM8kbXOiyrEwUXIt/GroundControlStation.pdf](http://api.ning.com/files/ga5AVWy8xTu8cx9rHdzJ73Epc*dzb0uy*UPe0O8wFxogMF6WNRW8StK3xzkG2iBk16kNFpYTZe80NgM8kbXOiyrEwUXIt/GroundControlStation.pdf), 2013. március 19., 15:00
- [6] <http://www.rodiangroup.com/uav-ground-control-station.html>, 2013. március 19., 15:00
- [7] <https://code.google.com/p/ardupilot-mega/wiki/HappyKillmore>, 2013. március 19., 16:00
- [8] [http://www.vikingaero.com/uav\\_ground\\_control\\_station.html](http://www.vikingaero.com/uav_ground_control_station.html), 2013. március 19., 16:00
- [9] <http://www.mono-project.com>, 2013. március 19., 17:00
- [10] <http://msdn.microsoft.com/en-us/library/system.io.ports.serialport.aspx>, 2013. március 20, 10:00
- [11] <http://com0com.sourceforge.net/>, 2013. május 4, 10:00
- [12] <http://www.digi.com/products/wireless-wired-embedded-solutions/zigbee-rf-modules/point-multipoint-rfmodules/xbee-pro-868>, 2013. október 29, 14:00

# Függelék

## F.1. Függelék1